Development an ontology based intelligence search system for goods for an online store on the CS-Cart platform

Gleb Guskov, Vladimir Zarayskiy, Aleksey Filippov, Anton Romanov
Information systems department
Ulyanovsk State Technical University
Ulyanovsk, Russia
guskovgleb@gmail.com, v-zar@list.ru, al.filippov@ulstu.ru, romanov73@gmail.com

***Abstract*—The article presents the implementation of a intelligence search system for an CS-Cart platform online store. The system is based on the formation of a domain products ontology. The system assumes the processing and expansion of the user request. This article describes the structure of an OWL ontology. The article describes the algorithm for processing custom queries, as well as the results of experiments.**

***Keywords*—knowledge base, ontology, smart search, e-commerce.**

I. Introduction

The term "information retrieval" was first introduced into scientific circulation by the American mathematician Calvin Moers in 1951. In state standard 7.73-96, information retrieval is actions, methods and procedures that provide the selection of certain information from a data array.

The problems of modern information retrieval are reflected in the work processes of the online store. It's that users can not quickly and accurately find the desired item and leave the online store.

Currently, the platform for the CS-Cart online store by default search by matches in the name or description of the product with the entered user request.

In this research, process of implementing a intelligence search system will be considered. Intellectual search system is part of an online store based on CS-Cart platform, which allows you to expand the capabilities of the standard functionality of the search engine.

Any Internet resource must have a search panel, as it is an integral part of the graphical user interface. Most systems today are based on the Boolean search model.

Boolean variables get the value "true" or "false" depending on the occurrence or not occurrence of the query terms in the corresponding document.

The search model is foundation for search technology in a particular system . The search model is a combination:

- the representations of documents form;
- an approach of forming representations of search queries;
- type of criterion of relevance of documents.

There are also systems that use external services to search through an array of data. Unfortunately, in this case, there is no way to control the behavior of the search engine, and the it's internal mechanisms are not available.

The growth of the World Wide Web has exposed the flaws classic search and indexing methods. Information retrieval can be based on a semantic structure. The Semantic Web allows connections to be created between terms, giving meaning to information and helping the machine integrate new hidden knowledge.Information retrieval can be based on a semantic structure, namely the relationship between information available on the network [1].This problem complicates the translation of user queries into an executable query for the information system, which leads to the loss of some of the relevant results when searching. Therefore, the identification of the semantics of search queries plays an important role in the development of search engines. This area of research is called semantic search. Semantic search aims to deliver contextually relevant results to the user by understanding the purpose and meaning of the query provided.

Most often, users know what they are looking for, but do not know how to formulate a search query. Users formulate a request based on the key characteristics or metadata of the sought object [2]. The online store provide to categorize products and set individual characteristics for each product. Presenting products, categories, characteristics and relationships between them as semantic network, and apply an inference engine to form the result of a search query, quality of the search engine will increase. The use of semantic search for an online store contains scientific interest.

On the basis of the ontological approach, there are platforms for creating information systems in which the advantages of semantic technologies are available [3], [4], [5].

Since with the help of ontology it is possible to formalize any subject area in the form of a set of statements and terms [6], that is, the possibility of using ontology for e-commerce. Ontological representation of knowledge will help customers can find the right products and make purchasing decisions. Ontologies can be applied in a search system to expand and adjust user queries.

II. Ontology based search system

The ontological approach to knowledge representation has many successfully solved problems in various industries. For example, the task of classifying documents in an electronic archive can also be solved using the ontological approach [7] or the task of finding similar program products [8].

The structure of the ontology classes is shown in “Fig. 1”. The concepts of characteristics and categories are stored in the "features" and "categories" classes, respectively. The first concept contains the names of the characteristics, and the second contains the names of the categories in the form in which these terms are presented in the online store. Product concepts are stored in the products class. The filling of the ontology with concepts occurs in an automated mode.

The intelligence search algorithm contains 3 stages:

- Query string preprocessing;
- Formation of logical conditions and definition of classes;
- Synchronization of the inference engine and getting a list of products.

A. Query string preprocessing

The input query string is split into tokens and stop words are removed from the resulting array of terms. Bigrams are formed from the received tokens. Then, using an approximate string comparison [4], terms for autocomplete are compared with a bigrams list. The algorithm for the approximate comparison of strings is based on the calculation of the Levenshtein distance [5]. Comparing two strings with each other, the comparison function produces a number from 0 to 100, which indicates the degree of similarity. From the beginning, a partial comparison takes place, which looks for a match with a bigram in the entire set of autocomplete terms. The resulting set of matches is re-checked against the bigram using a word-by-word comparison. Thus, a list of concepts that exist in the ontology is distinguished.

B. Forming logical conditions and determine classes

At the second stage of the formation of logical conditions, it is checked to which class the element of the list of found concepts belongs. For each characteristic from the general set of conditions, a new generated condition of the "hasFeature value currentFeature" format is added. Where "hasFeature" is an entity of type "ObjectProperty", which allows you to create a relationship

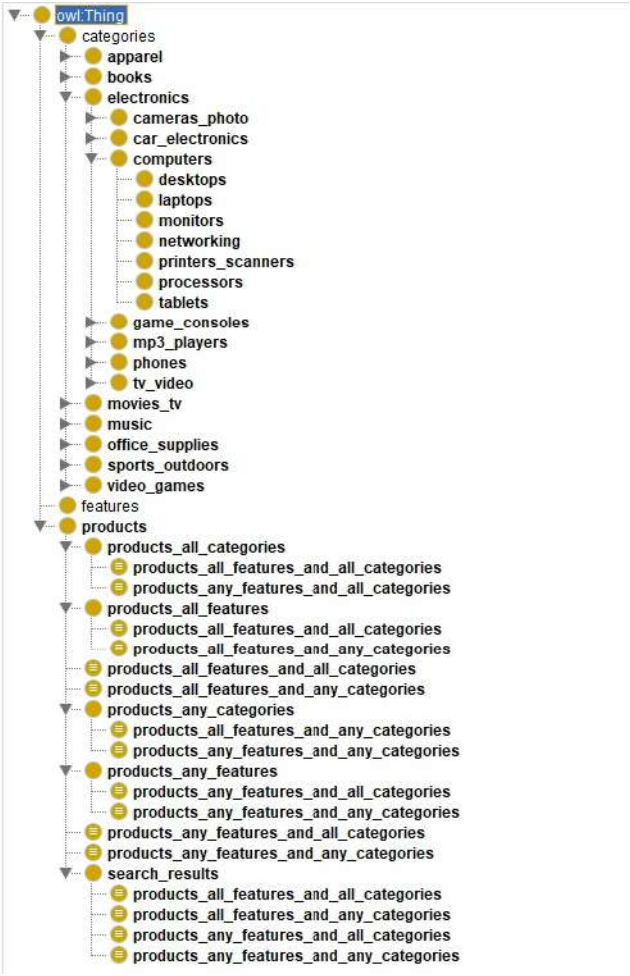

Figure 1. Ontology class structure.

between the concept of the product and the concept of the characteristic "currentFeature". A set of conditions for characteristics and categories form equivalence conditions for four classes: two for the logical "AND" condition and two for the logical condition "OR". The sets of conditions are also combined using the logical operators "AND" and "OR", and form the value of the equivalence of four classes:

- products_all_categories - products belonging to all categories from a set of conditions;
- products_all_features - products that have all the characteristics from a set of conditions;
- products_any_categories - products belonging to at least one category from a set of conditions;
- products_any_features - products that have at least one characteristic from a set of conditions.

These classes have subclasses that represent the intersections of the sets of found concepts:

- products_all_features_and_all_categories - products that have all characteristics and belong to all categories from a set of conditions;

- products_all_features_and_any_categories - products that have all characteristics and belong to at least one category from a set of conditions;
- products_any_features_and_all_categories - products that have at least one characteristic and belong to all categories from a set of conditions;
- products_any_features_and_any_categories - products that have at least one characteristic and belong to at least one category from the set of conditions.

C. Information system developed during the research

The sequence diagram shown in “Fig. 2” is the best way to describe how the system works. The diagram shows the process of searching for goods in the system being developed in general form. The diagram consists of the following elements: 1. User request; 2. Two repositories of information (database and knowledge base); 3. Two developed program modules (search controller and search service); 4. An auxiliary module required for preprocessing a request.

As part of the research, a web service in python was developed using the REST API of the flask framework and a module for an online store on the CS-Cart platform version "4.11.2 Ultimate" in php. The ontology is stored in an owl file on the web service server. Interaction with the ontology is carried out using the owlready2 [9] library for python. Autocomplete words are stored as a list in a json file.

III. Synchronizing the inference engine and forming list of products

The third stage starts the synchronization of the inference engine. Subclasses that allow you to identify the intersection of the sets of found concepts contain products that match the conditions. It is necessary to traverse all four classes in order from more precise to more complete list. More precise is a class that contains products that satisfy all the conditions, connected by the "AND" operator. And less accurate, but more complete, through the "OR" operator. The HermiT inference engine was used to synchronize the classes.

IV. Experimental results

Experiments were run on demo store data, which contains 83 categories, 300 products, and 35 types of characteristics. The time for filling the ontology with all concepts and relations between them ranged from 0.2 to 1 second, depending on the amount of data. To check the synchronization time of the inference engine on the same user request, larger data sets were formed based on the existing ones. The library for interaction with ontology allows to automatically launch two inference machines: HermiT and Pellet. The results of the execution time of queries are presented in ”Table. I”.

The results of the experiments, HermiT is faster at coping with inference almost twice as fast as Pellet.

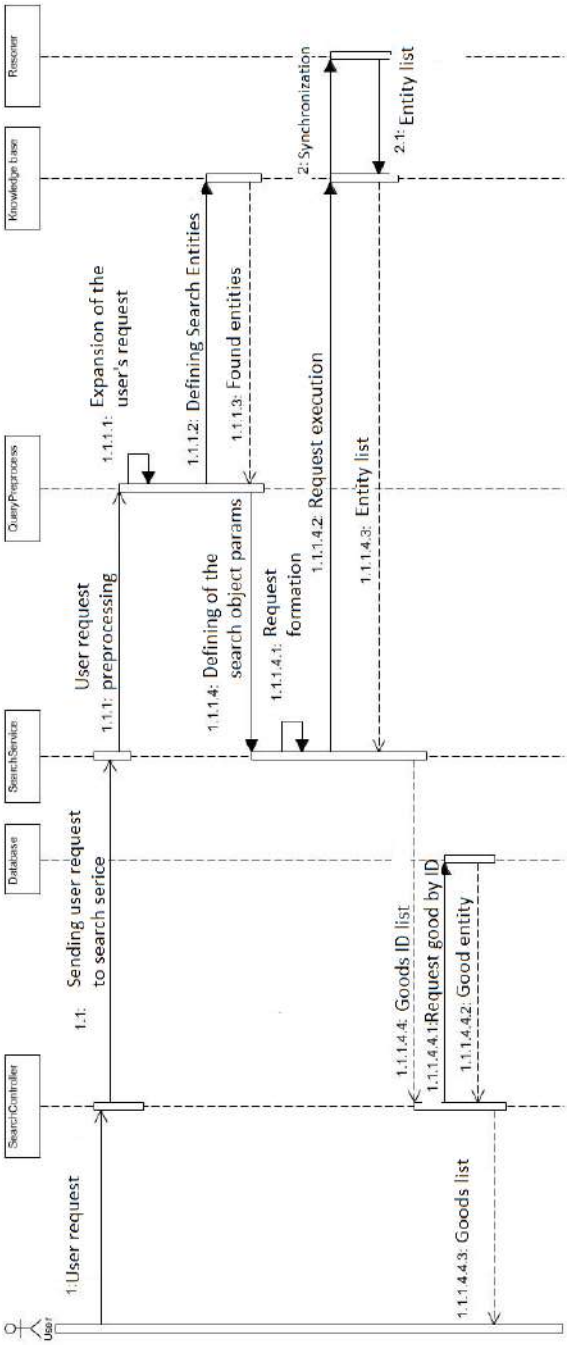

Figure 2. System sequence diagram.

Table I
THE RUNNING TIME OF THE ALGORITHM WITH DIFFERENT REASONERS

Total goods	***Founded goods***	***HermiT***	***Pellet***
300	10	4,5 sec	9 sec
600	20	15,8 sec	18,2 sec
1000	30	26,5 sec	46,3 sec
2000	60	72,1 sec	180 sec
5000	150	529 sec	1010 sec

The larger the total number of products included in the ontology, the longer the search takes. The results of executing queries by the intelligent search system(ISS) and the standard search algorithm(SSA) based on word coincidence are presented in "Table. II'.

Table II
COMPARISON OF QUERY RESULTS

Query	Goods count		Relevant goods count	
	ISS	*SSA*	*ISS*	*SSA*
blue medium t-shirt	1	0	1	0
apple iphone 128gb	10	0	2	0
windows 7 premium	9	0	8	0
monitor ips fullhd	29	0	7	0
huawei notebook	18	0	16	0
huawei	3	3	3	3
games	1	11	1	5

The results of the experiments, the standard search algorithm cannot find any product based on user queries consisting of several words. When searching for the word "huawei", the standard search system found all relevant products, since this word was present in the name of the products. When searching for the phrase "huawei notebook", the standard search search system did not find the products, as it was in the previous query, since this phrase is missing in the name or description of the goods. The intelligent search system was able to find the same products for the query "huawei" as the standard search algorithm, since in this case, among the characteristics of the products found, there was a brand characteristic with the value "huawei". When performing the query "huawei notebook" among the products that have the brand "huawei" and the type of notebook "notebook", there was no combination of such characteristics. However, there are products that have either the first or the second characteristic. Therefore, the system strives to offer the user products that may be of interest to him, and not leave him with an empty search result. The average accuracy of the smart search system according to the results of the experiments from Table 2 was 70 percents.

V. CONCLUSION

As a result of the work, an intelligent search system was developed for an online store based on the CS-Cart platform using ontology. The experiments carried out show that the developed system gives out products with characteristics and categories by which the user searches, in contrast to the standard search. The standard search algorithm does not find products for the query, for which smart search returns products. The running time of the algorithm with the HermiT inference engine is less than 10 seconds for a small number of products, which can be used for online stores with a specific subject area. To increase the accuracy of the algorithm, you can add additional information about the product to the ontology and modernize the query preprocessing process. To reduce the query time, you need to add an inference engine written in python to the library.

ACKNOWLEDGMENT

This work is supported by the Russian Foundation for Basic Research (project No. 19-47-730006).

REFERENCES

[1] Saravanaguru K., Shekhar M. Semantic Web Search based on Ontology Modeling using Protégé Reasoner. School of Computing Science and Engineering, 2013, pp. 10

[2] Kravchenko Yu.A., Markov V.V., Novikov A.A. Cemanticheskii poisk v semantic web [Semantic search on the semantic web]. Izvestiya YuFU. Tekhnicheskie nauki [Izvestia Yufu. Technical science], 2016, 11p. (In Russ)

[3] Shunkevich D.V., Taberko V.V., Ivaniuk D.S., Pupena O.N. Ontological Approach for Standards Development Within. Otkrytye semanticheskie tekhnologii proektirovaniya intellektual'nykh system [Open semantic technologies for intelligent systems], 2013, pp. 31-42.

[4] Yarushkina N.G., Filippov A.A., Moshkin V.S., Guskov G.Yu. Development of a System Based on a Fuzzy Knowledge Base Filled from Various Formats Documents. Recent Developments and the New Direction in Soft-Computing Foundations and Applications. Studies in Fuzziness and Soft Computing, vol 393. Springer, Cham. https://doi.org/10.1007/978-3-030-47124-8_13

[5] Filippov A.A., Yarushkina N.G., Moshkin V.S. Development of a Technological Platform for Knowledge Discovery. Computational Science and Its Applications – ICCSA 2019. ICCSA 2019. Lecture Notes in Computer Science, vol 11619. Springer, Cham. https://doi.org/10.1007/978-3-030-24289-3_39.

[6] Lukashevich N.V. Tezaurusy v zadachakh informatsionnogo poiska [Thesauri in information retrieval problems], Moskow Izd-vo Moskovskogo universiteta [Moscow Publishing house of Moscow University], 2011, 512 p. (In Russ)

[7] Koval A.R., Zarubin A.A., Moshkin V.S. Classification of Text Documents of an Electronic Archive Based on an Ontological Model. Intelligent Data Engineering and Analytics, 2021. https://doi.org/10.1007/978-981-15-5679-1_56.

[8] Dudarin P.V., Yarushkina N.G., Guskov G.Yu. Two Phase Approach to Detection of Software Projects with Similar Architecture Based on Clustering and Ontological Methods. Recent Developments and the New Direction in Soft-Computing Foundations and Applications. Studies in Fuzziness and Soft Computing, vol 393. Springer, Cham. https://doi.org/10.1007/978-3-030-47124-8_12.

[9] LAMY Jean-Baptiste Owlready2 Documentation Owlready2. at: https://owlready2.readthedocs.io//en//v0.31// (accessed 2021, May)

Разработка системы интеллектуального поиска товаров на основе онтологий для интернет-магазина на платформе CS-Cart

Гуськов Г. Ю., Зарайский В.И.
Филиппов А.А., Романов А.А.

В статье представлены особенности реализации системы интеллектуального поиска для интернет-магазина на платформе CS-Cart. Предлагаемая система основана на формировании онтологии предметной области к которой относятся товары. В рамках исследования рассматриваются процессы системы по обработке и расширению запроса пользователя. В статье представлена структура онтологии в формате OWL. В работе описан алгоритм обработки пользовательских запросов, а также представлены результаты экспериментов.

Received 31.05.2021