Design and Development of an Intelligent Semantic Recommendation System for Websites

Zhiqiang Zhang[1(✉)], Heping Yang[1], Di Yang[1], Xiaowei Jiang[1], Nan Chen[1], Mingnong Feng[1], and Ming Yang[2]

[1] National Meteorological Information Center, Beijing 100081, China
zhangzhiqiang@cma.gov.cn
[2] Zhejiang Meteorological Information Network Center, Hangzhou 310001, China

Abstract. When searching for the interesting content within a specific website, how to describe the initial need by selecting proper keywords is a critical problem. The character-matching search functions of website can hardly meet users' requirements. Furthermore, building the content of webpages of a specific website and the associated rules is uneconomical. This paper, based on the framework of the Lucene engine, applied a semantic ontology, the calculation of the relevance of word entries, and the semantics of keywords to design an intelligent semantic recommendation system with the Jena secondary semantic analysis technique. Subsequently, the expanded keywords were semantically ranked based on the term frequency analysis technique. Meanwhile, the ontology algorithm and their relevance were introduced as the dynamic weight values. Finally, in the text content retrieval process, the search results were ranked based on the previous relevance weights. The experimental results show that the system designed in this paper is not only easy to develop but also capable of expanding users queries and recommending relevant content. Further, the system can improve the precision and recall for website search results.

Keywords: Ontology · Vertical search engine · Semantic expansion · System design

1 Introduction

With the development of computer and network, the conventional character-matching search technology need to be improved because appropriate keywords can not be confirmed to meet users' requirements. To improve the search accuracy and semantic relevance, VSEs (vertical search engines) with a relatively deep background in domain knowledge have been gradually applied in various industries [1]. VSEs are characterized by providing professional search results and carry the marks of the industries, such as Google Scholar, China National Knowledge Infrastructure Search Engine, Wan-Fang Data, Book Search Engines [2], Education Resource Search Engines [3] and Geographic Information Search Engines [4]. A pure VSE filter can screen, reindex and store web crawler data based on a certain domain knowledge format for the users' retrieval [5].

X. Zhang et al. (Eds.): CloudComp 2019/SmartGift 2019, LNICST 322, pp. 200–209, 2020.
https://doi.org/10.1007/978-3-030-48513-9_16

Semantics reasoning based on VSEs was introduced to expand the implicit information from users' queries with the logical relationships in an ontology [6]. The previous ontologies, such as Cyc (http://www.cyc.com/) [7] and WordNet (a lexical database; http://wordnet.princeton.edu/) [8], can augment the content of different characters based on the semantics. Therefore, ontologies introduced into VSEs had significantly improved the precision and recall of the search results. Ontology-based VSEs are extensively used in various fields, including agriculture [9], industry [10] and text mining [11]. This technology has become an effective method and development trend for knowledge summarization and query sharing in various fields.

Currently, building an ontology requires a generalized or specialized thesaurus and clear logical relationships between the descriptors. Furthermore, the query is expanded through reasoning based on the semantic logical relationships in the ontology [12]. To ensure high precision and recall in the retrievals, it is necessary to set up higher standards for both the integrity of the thesaurus and the logical relationships between the objects in the ontology (the terms in the thesaurus).

Focused on the above analyses, the paper presented a simple ontology for a data website based on the structure of its navigation directory, which could apply vertical searching with the development framework of the search engine. Indexes and semantics expanded the word segmentation results after query segmentation, and then the relevance of each expanded keyword was applied to the query as the dynamic weight in the subsequent calculation of the relevance ranking from the search results. For realizing relevance ranking, it is critical to improve the precision of search results and permit users to rapidly and accurately search for the target data and relevant information from the website. This intelligent semantic recommendation simplified the process of building an ontology-based VSE for a website, which is suitable for building ontology-based vertical search systems for the specific websites of varying sizes (large, medium and small).

2 Structural Framework and Flow of the System

The technical contents of the system were divided into three modules: analyze the relevant contents of the website and build an index database in the background; establish an ontology and an ontology index database based on the map and navigation directory of the website; segment, expand and index the content based on the users' queries and rank of the recommended results. Figure 1 shows the main design flow.

2.1 Webpage Index Database

An index database for previous webpage contents can be built by the following steps:

Step 1: store the relevant links (e.g., the homepage) in the URL (Uniform Resource Locator) list in a web crawler;
Step 2: traverse the URL list to obtain the webpage contents;
Step 3: analyze webpage contents, store the text, extract the relevant URL links, store them in the URL list, and then go to Step 2;

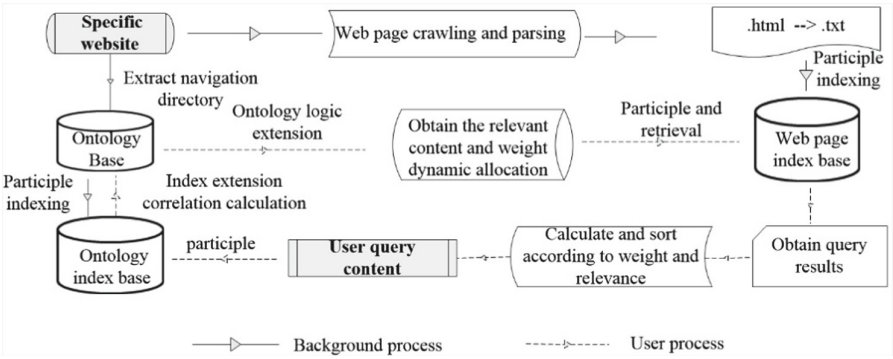

Fig. 1. Design framework for an ontology-based vertical search system.

Step 4: divide text contents into various fields, segment the words in each field, and build a webpage index database.

For information added to the website at later stage, Step 4 can be directly executed to improve the webpage index database.

2.2 Ontology and Ontology Index Database

From a prototype development point of view, this system established a simple ontology based on the navigation directory of website.

Step 1: list the column headings as ontology objects;
Step 2: define the relationship attributes between the ontology objects based on the column levels;
Step 3: build an ontology for the website;
Step 4: build indexes for objects name in an ontology and build an ontology index database.

2.3 Webpage Index Database

Step 1: segment the words in a query input by the user (original query);
Step 2: perform an index search in the ontology index database (Step 4 in Sect. 2.2), rank the search results based on the relevance, and take the relevance scores as the weight of the keywords for later-stage analysis:

$$\text{RESULT}\{(\text{K1},\ \text{Score\_1}),\ (\text{K2},\ \text{Score\_2}),\ \ldots\ldots,\ (\text{Kn},\ \text{Score\_n}),$$

where K represents a record in the ontology index database related to the query, and Score represents the corresponding relevance score;

Step 3: perform reasoning on each object in RESULT within the ontology obtained in same type objects:
RESULT{{(K1, Score_1), (K1_k1, Score_1), (K1_k2, Score_1) , ontology to obtain n Sect. 2.2), rank the search results based on the relevance a website.si, where, Kn_km represents the mth object of the same type determined by reasoning based on the nth content, and Score represents the weight inherited from Step 2;
Step 4: segment and deduplicate the words in each object in RESULT (the segmented keywords inherit the previous weight (for duplicate keywords, the highest weight is inherited)) and perform an index search;
Step 5: set the weight of each field in the webpage index database, rank the search results based on the weights dynamically assigned to the keywords, and list the results.

3 Key Technologies

Focused on the intelligent semantic recommendation system framework, the related technologies and their implementation were introduced in this section.

3.1 Web Crawler

Heritrix, an open-source web crawler written with Java language, can be customized, modified and encapsulated as flexible web crawling tool [13]. Heritrix was applied to crawl and import a data website into the Html parser package to analyze and design webpage contents, and then a patterned data extraction framework was developed. Currently, two types of webpage data website are dataset information webpage and dynamic news information webpages, respectively. The relative webpage analysis and extraction framework are developed for these two types of webpage, which were used to screen the effective information.

3.2 Index Building and Retrieval

Webpage information was processed and stored as a new text file, and Lucene was used to build a search engine. Lucene is an open-source full-text search library in the Java development environment. For creating indexes between files, Lucene can also create indexes within a file to improve the retrieval efficiency. Moreover, Lucene can also perform Boolean operations and fuzzy and group searches [14]. Meteorological data website was selected in the research, and the text files were introduced to creating indexes. Meanwhile, intra-text file indexes were created with 10 components of the data composition, including data name, keyword and spatial range. During the search process, Lucene was used to perform a multi-field search for the keywords (Multi-Field Query Parser) and allocate the corresponding weight to facilitate the calculation and rank the relevance of the results.

3.3 Chinese Word Segmentation

Building a search engine (including creating indexes and realizing user retrieval) with Lucene, it is necessary to perform word segmentation on the metadata and to index for users' queries. The IKAnalyzer, fine-grained segmentation algorithm for forward iteration, was designed to segment word in the system with processing capacity of 600,000 words per second. Chinese word segmentation toolkit was developed with Java language and supported industry-specific and users' customized dictionaries [15]. The system applied IKAnalyzer 5.0 jar toolkit and configures industry-specific and user's dictionaries through the designment.

3.4 Ontology Establishment

Ontology was built based on Protégé [16] for the CMDSC (China Meteorological Data Service Center) website. The main concepts in this ontology include data technology and service processes. The data technology include ground, upper-air and maritime data. The ground data include basic meteorological element data observed at Chinese ground meteorological stations, climate data collected at Chinese ground international exchange stations and standard value of ground climate in China. The main relationships between the concepts are part-of and instance-of relationships (the system considers only part-of relationships).

3.5 Query Expansion

The system expanded a query and dynamically assigns a weight to the relevant contents by the method described in Sect. 2.3. This section focuses mainly on the sematic expansion of ontology. Jena is primarily used to expand the ontology for queries with logical reasoning and a parser for files in the RDF (Resource Description Framework)/Extensible Markup Language and OWL formats. The SPARQL Protocol and RDF Query Language in Jena can be used to retrieve relevant semantics in the ontology. The Jena application programming interface can be used to determine the position of an object in the ontology as well as the class/subclass above and below and at the same the position of the object [17].

3.6 Result Ranking

Result ranking is a key part for users' experience. The higher relevance of the content keywords had in users' queries, the higher rank the content located. Using the TF-IDF (Term Frequency–Inverse Document Frequency) algorithm [18], the system calculates the relevance of the search results by setting the field weight and dynamically assigning weight to the segmented words based on Eq. (1):

$$SCORES = TF \times IDF \times BOOST \times fieldNorm \tag{1}$$

where *TF* is the square root of the number of appearances of the searched word in the file, *IDF* is the inverse document frequency, which is the number of files appeared by

retrieved content, *BOOST* is the boost factor, whose value can be set using both field and doc (the values set using field and doc will take effect at the same time), *fieldNorm* is calculated in advance (when the TF and IDF remain unchanged, the less content a file contains, the higher value the fieldNorm has):

$$IDF = log(numDocs/(docFreq + 1)) + 1 \tag{2}$$

where *numDocs* is total number of files, *docFreq* is the number of files contained the word.

$$fieldNorm = 1/\sqrt{(wordsNum - 1)} \tag{3}$$

where *wordsNum* is file length.

4 Experiment and Analysis

4.1 Data Preparation

Total 1,640 dataset webpages and 159 pieces of published dynamic and popular science information were crawled from a meteorological data website from CMDSC with the Heritrix web crawler. Fields were designed and indexed for the relevant file information (Table 1). TextField showed that indexes were created for segmentic words and supports users' queries. StringField showed that indexes were not created for the segmented words. Weight showed the BOOST (boost factor) of results ranking after users' searches in each field.

Table 1. Building index information for dataset file fields.

Field name	Explanation	Field type	Weight
Title	Data name	TextField	1.0f
Description	Data description	TextField	1.2f
Keyword	Keyword	TextField	1.2f
s_date	Start time of the data	StringField	–
e_date	End time of the data	StringField	–
Range	Spatial range	TextField	1.0f
Source	Data source	TextField	1.0f
q_description	Data quality description	TextField	1.1f
m_time	Time of creation	StringField	–
Freq	Update frequency	TextField	1.0f
Url	Website	StringField	–

4.2 Chinese Word Segmentation

The accuracy of segmenting the words in a file and a user's query directly affects the precision of the search results. To improve the word segmentation accuracy, 15,456 specialized meteorological terms were imported into IKAnalyzer as a customized lexicon. Table 2 summarizes the comparison of the segmentation of 10 common meteorological terms selected from the main types of data on the CMDSC website. Apparently, the intelligent word segmentation accuracy of IKAnalyzer increased significantly after importing the specialized terms. Further, specialized terms need to segment the words in a file at a finer granularity level during the indexing process. The system uses the fine-granularity mode of IKAnalyzer to segment sentences to ensure the comprehensiveness of the index building.

Table 2. Search results obtained based on keyword matching and ontology matching.

Number	Word content	Search results obtained based on character matching	Search results obtained based on ontology matching
1	Vertical cumulative liquid water content	86(15)	1084(105)
2	Multi-year ten-day upper-air level value	1092(219)	1416(151)
3	Dew point temperature of air	486(32)	486(32)
4	Single-station Doppler radar base data	287(23)	287(23)
5	Chinese FengYun polar-orbiting meteorological satellite	992(69)	1414(148)
6	Historical atmospheric dust fall	433(46)	1092(131)
7	Agricultural meteorological data	1176(131)	1362(141)
8	Tropical cyclone data	1355(129)	1360 (138)
9	Numerical weather forecast model	571(53)	902(135)
10	Integrated grid precipitation data	1347(142)	1351(149)

5 Result and Discussion

The precision (hit rate) and the recall of the retrieval results are the standard measures for determining the quality of a search engine. Searches were performed using the terms

listed in Table 2. Table 2 shows the search results obtained by pure character-matching searching and by searching after query expansion in the ontology. Clearly, the recall was significantly higher than that performed based on ontology-matching. The two approaches yielded the same numbers of search results for "dew point temperature of air" and "single-station Doppler radar base data". For "dew point temperature of air", and the object names in the ontology did not contain the information without expanded words. For "single-station Doppler radar base data" within the ontology, "Doppler radar base data" was expanded, and the expanded keywords were included after segmenting the original word. Therefore, the two approaches yielded the same number of search results.

The accuracy of the first N number of search results (TopN) (i.e., the proportion of the first relevant N records) was used to examine the precision (hit rate). Research data obtained showed that the user views of the first page of the search results account for 47% of the total user views [19]. Searches were performed using the terms listed in Table 2. The accuracy of the top 20 search results was calculated and found to be at 100%. This finding occurs mainly because there was a hit on the names of some of the datasets, the datasets of 31 provinces of the same type could be obtained (i.e., the dataset headings have the same characters except for only the name of the province), which affected the determination of the overall accuracy. During the statistical analysis, datasets for 31 provinces of the same type were treated as one record. For example, the Standard value of 31 provinces were treated as only one record. The numbers in the brackets in Table 2 are the statistical values obtained after this treatment.

Figure 2 shows the search accuracy determined after the abovementioned treatment. As shown in Fig. 2, the accuracy of the search results obtained after the expansion of the ontology was significantly higher than that of the search results obtained by character matching. This finding occurs mainly because weights were assigned to the keywords obtained by semantic expansion when scoring and ranking the relevance of the search results, which improved the hit rate (precision) for the query. There were relatively few search results for "dew point temperature of air" and "single-station Doppler radar base data", which basically appeared in the first 20 records. So, the accuracy was same. The accuracy of the search results for "tropical cyclone data" was low in both cases, which is mainly due to the relatively small amount of relevant information on the website.

A comparison between the pure character-matching searching approach and ontology-matching shows that this system has higher recall and precision. This system is suitable for rapidly locating requisite information within the enormous volumes of meteorological data on the CMDSC website and then it was proved to be an effective means for realizing meteorological data sharing.

(1) Indexing and query segmentation. In the process of building the index database for the system and segmenting a user's query, a specialized lexicon and a stop word library were introduced to segment the indexes at a coarse granularity level and intelligently segment the query. The aim to segment and filter the specialized terms and improve the relevance of the search results to the special field had been achieved.
(2) Ontology building and expansion. An ontology was built for a website based on the functional requirement of the system and the structure of the navigation directory of

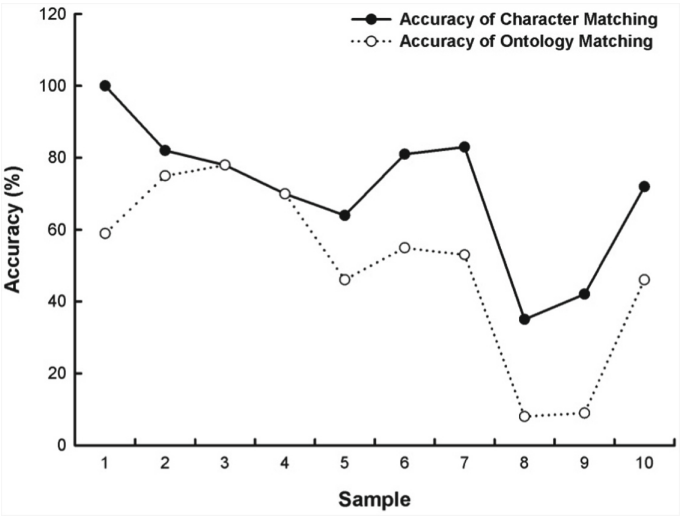

Fig. 2. Accuracy of the top 20 search results for the 10 query contents.

the website. In addition, the ontology index searching and analysis techniques were applied to expand a user's query, which increases the number of records relevant to the queries retrieved with higher recall.

(3) Dynamic assignment of weight to terms. The system not only assigned weights to the indexed contents of a webpage in various fields but also dynamically assigns weight to the keywords in a query. In other words, the relevance of a user's query for expanding the ontology object index database was used as the weight of each keyword when calculating the relevance rank of the search results. This approach optimizes the ranking results and significantly improves the precision (hit rate) of the search engine.

6 Conclusions

The intelligent semantic recommendation system established an ontology based on the navigation directory of a website. In the query expansion process, the ontology object index database and semantic reasoning had been combined. A comprehensive investigation and experiment were performed on an ontology-matching search engine. The design process of this method can serve as a reference for building simple ontology-based VSEs for other websites. The ontology-based semantic vertical searching function can be further improved later by expanding the ontology of the relevant specialized terms. Further, some other objects could be introduced into the ontology and the better searching results could be obtained.

Acknowledgments. This work was supported partially from Chinese National Natural Science Foundation "Development of Data Sharing Platform of Tibetan Plateau's Multi-Source Land-Atmosphere System Information" under grant number 91637313; the Special Scientific Research

Fund (Major Special Project) for Public Welfare Professions (Meteorology) under the grant number GYHY(QX) 20150600-7.

References

1. Aizawa, A.: An information-theoretic perspective of TF-IDF measures. Inf. Process. Manag. **39**, 45–65 (2003)
2. Chang, P.C., Galley, M., Manning, C.D.: Optimizing Chinese word segmentation for machine translation performance. Presented at The Workshop on Statistical Machine Translation, pp. 224–232. Association for Computational Linguistics (2008)
3. Cong, Y., Chan, Y., Ragan, M.A.: A novel alignment-free method for detection of lateral genetic transfer based on TF-IDF. Sci. Rep. **6**, 1–13 (2016)
4. Corby, O., Dieng-Kuntz, R., Faron-Zucker, C.: Querying the semantic web with corese search engine. Presented at European Conference on Artificial Intelligence, pp. 705–709 (2017)
5. Fu, Q.: Lucene research and implementation on the vertical search engine application to university library books. J. Taiyuan Normal Univ. **10**, 104–107 (2011)
6. Hendler, J., Lenat, D., Lenat, D., et al.: Very large knowledge bases-architecture vs engineering. Presented at International Joint Conference on Artificial Intelligence, pp. 2033–2036. Morgan Kaufmann Publishers Inc. (1995)
7. Hsu, Y.Y., Chen, H.Y., Kao, H.Y.: Using a search engine-based mutually reinforcing approach to assess the semantic relatedness of biomedical terms. Plos One **8** (2013). https://doi.org/10.1371/journal.pone.0077868
8. Kara, S., Alan, O., Sabuncu, O., et al.: An ontology-based retrieval system using semantic indexing. Presented at the IEEE International Conference on Data Engineering Workshops, pp. 197–202 (2012)
9. Liu, D.F., Fan, X.S.: Study and application of web crawler algorithm based on heritrix. In: Advanced Materials Research, vol. 220, pp. 1069–1072 (2011)
10. Lombardo, V., Piana, F., Mimmo, D.: Semantics–informed geological maps: conceptual modeling and knowledge encoding. Comput. Geosci. **116**, 12–22 (2018)
11. Mcbride, B.: A semantic web toolkit. IEEE Internet Comput. **6**, 55–59 (2002)
12. Noy, N.F., Sintek, M., Decker, S., et al.: Creating semantic web contents with Protégé-2000. IEEE Intell. Syst. **16**, 60–71 (2005)
13. Yao, Y.: Library resource vertical search engine based on ontology. Presented at International Conference on Smart Grid and Electrical Automation, pp. 672–675. IEEE Computer Society (2017)
14. Huntley, R., Dimmer, E., Barrell, D., et al.: The gene ontology annotation (GOA) database. Nat. Preceding **10**, 429–438 (2009)
15. Pirro, G., Talia, D.: An approach to ontology mapping based on the Lucene search engine library. Presented at IEEE International Workshop on Database and Expert Systems Applications, pp. 407–411 (2007)
16. Wang, C., Li, S., Xiao, H.: Research on Ontology-based arid areas agriculture search engine. J. Agric. Mech. Res. **8**, 184–191 (2013)
17. Reviewer-Lin, D.: Review of WordNet: An Electronic Lexical Database. MIT Press, Ch. 25, pp. 292–296 (1999)
18. Sun, J., Li, Y., Wan, J.: Design and implementation of the search engine for earthquake based on Heritrix and Lucene. Seismol. Geomagnetic Obs. Res. **37**(5), 172–178 (2016)
19. Ding, Y.H., Yi, K., Xiang, R.H.: Design of paper duplicate detection system based on Lucene. Presented at IEEE Wearable Computing Systems, pp. 36–39 (2010)