Yu *EURASIP Journal on Wireless Communications and Networking* (2019) 2019:30
https://doi.org/10.1186/s13638-019-1354-z

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

Research on information retrieval model based on ontology

Binbin Yu[1,2]

Abstract

An information retrieval system not only occupies an important position in the network information platform, but also plays an important role in information acquisition, query processing, and wireless sensor networks. It is a procedure to help researchers extract documents from data sets as document retrieval tools. The classic keyword-based information retrieval models neglect the semantic information which is not able to represent the user's needs. Therefore, how to efficiently acquire personalized information that users need is of concern. The ontology-based systems lack an expert list to obtain accurate index term frequency. In this paper, a domain ontology model with document processing and document retrieval is proposed, and the feasibility and superiority of the domain ontology model are proved by the method of experiment.

Keywords: Ontology, Information retrieval, Genetic algorithm, Sensor networks

1 Introduction

Information retrieval is the process to extract relevant document from large data sets. Along with the increasing accumulation of data and the rising demand of high-quality retrieval results, traditional information-retrieval techniques are unable to meet the task of high-quality search results. As a newly emerged knowledge organization system, ontology is vitally important in promoting the function of information retrieval in knowledge management.

The existing information retrieval model, such as the vector space model (VSM) [1], is based on certain rules to model text in pattern recognition and other fields. The VSM splits, filters, and classifies the text that looks very abstract, and carries on the statistics to the word frequency data of the text. The computer carries out the text according to certain rules and carries on the statistics to the word frequency information of the text.

Probability model [2] mainly relies on probabilistic operation and Bayes rule to match data information, in which the weight values of feature words are all multivalued. The probabilistic model uses the index word to represent the user's interest, that is, the personalized query request submitted by the user. Meanwhile, there is no vocabulary set with a standard semantic feature and document label. Traditional weighted strategy lacks semantic information of the document, which is not representative for the document description. On the basis of semantic annotation results, weighted item frequency [3] and domain ontology of the semantic relation are used to express the semantics of the document [4].

The VSM and probability model can simplify the text processing into vector space or probability set. It involves the term frequency property to describe the number of occurrences of query words in the paper. Considering the particularity of document segmentation, the word in different sections has a different weight of summarization for the paper, which simply calculates that word appearance is not sufficient. Meanwhile, there is no vocabulary set with standard semantic feature and document label.

The introduction of ontology into the information retrieval system can query users' semantic information based on ontology, and better satisfy users' personalized retrieval needs [5]. Short of the vocabulary set with semantic description, user information demand logic view is miscellaneous and incorrect to express the semantic of the user's requirement. In such an information retrieval model, even if we choose the appropriate sort function R (R is the reciprocal of the distance between

Correspondence: yubinbin80@sina.com
[1]College of Computer Science and Technology, Jilin University, Changchun, China
[2]College of Information Technology and Media, Beihua University, Jilin, China

points), the logical view cannot represent the requirements of the document and the user, and the retrieval results cannot convince the user.

In order to improve the accuracy and efficiency of user retrieval, we build a model based on information retrieval and domain ontology knowledge base. The combination of ontology-based information retrieval system provides semantic retrieval, and a keyword-based information retrieval system calculates a better factor set in document processing, with better recall and precision results.

The contributions of this paper are as follows:

Genetic algorithm is designed and implemented. Genetic algorithm is a kind of search method that refers to the evolution rule of the biological world. It mainly includes coding mechanisms and control parameters. The genetic algorithm is a heuristic method which simulates the population evolution by searching through solution space in selection, crossover, and mutation to select an optimal factor set by combinations of factors. The option weighted factor tuned by a training set using genetic algorithms will apply to a practical retrieval system [6].

Domain ontology is applied as the base of semantic representation to effectually represent user requirement and document semantics. Domain ontology is the detailed description of domain conceptualization which expresses the abstract object, relation, and class in one vocabulary set [7].

Designing and implementing the information retrieval system is composed of two parts: document processor and document retrieval. In the information retrieval model, an ontology server is added to tags and indexes the retrieval sources based on ontology; the query conversion module implements semantic processing in users' needs and expanses the initial query on its synonym, hypernym, and its senses. The retrieval agent module uses the conversion of queries for retrieving the information source.

The full text is divided into five parts: the first part is an overview of ontology-based information retrieval system. The second part introduces the relevant work of this study. The third part is the design of information retrieval model based on domain ontology. The fourth part carries on the experimental study and analyses of the result. The fifth part summarizes the full text and puts forward the issues that need further study.

2 Methods

Faced with the huge amount of data in the network, it is an important problem for users to acquire the information accurately and efficiently. So far, retrieval methods develop various mathematical models. The classical information retrieval models include the Boolean model [8], probability model [9], vector model [10], binary independent retrieval model, and BM25 model. The following are the solutions of these models.

Suppose k_i is the index term, d_j is the document, $w_{i,j} \geq 0$ is the weight of tuples (k_i, d_j), which is the significance of k_i to d_j semantic contents. Let t refer to the number of index term. $K = \{k_1, \ldots, k_t\}$ is index term set. If an index term does not appear in the document, then $w_{i,j} = 0$. So the document d_j is represented by an index term vector $\overrightarrow{d}_j$:

$$\overrightarrow{d}_j = \left(w_{1j}, w_{2j}, w_{3j}, \ldots, w_{tj}\right) \tag{1}$$

The Boolean model is a classical information retrieval (IR) model based on set theory and Boolean algebra. Boolean retrieval can be effective if a query requires unambiguous selection [11]. But it can only result in whether the document is related or not related. The Boolean model lacks the ability to describe the situation that query words partially match a paper. The similarity result of document d_j and query q is binary, either 0 or 1. The binary value has limitations and the Boolean queries are hard to construct.

The VSM, which is proposed earlier by Salton, is based on the vector space model theory and vector linear algebra operation, which abstract the query conditions and text into vectors in the multidimensional vector space. The multi-keyword matching here can express the meaning of the text more [1]. Compared with the Boolean model, the VSM calculates relevant document ranking by comparing the angle relating similarity between the vector of each document and the original query vector in the spatial representation.

The probabilistic model [2] mainly relies on probabilistic operation and Bayes rule to match data information. The probabilistic model not only considers the internal relations between keywords and documents, but also retrieves texts based on probability dependency. The model usually based on a group of parameterized probability distributions, consumes the internal relation between keywords and documents and retrieves according to probabilistic dependency. The model requires strong independent assumptions for tractability.

The binary independence retrieval model [12] is evolved from the probabilistic model with better performance. Assuming that document D and index term q is described in two-valued vector $(x_1, x_2, \ldots x_n)$, if index term $k_i \in D$, then $x_i = 1$; otherwise, $x_i = 0$. The correlation function of index term and document are shown below.

$$\mathrm{Sim}(D, q) = \sum \log \frac{p_i(1-q_i)}{q_i(1-p_i)} \tag{2}$$

Here, $p_i = r_i/r$, $q_i = (f_i - r_i)/(f - r)$, f refers to amount number of document in the training document set. r is the number of document related to user query in the training document set. f_i represents a number of document, including index term k_i in the training document

set. R_i is the number of document, including k_i in r relation documents.

Okapi BM25 model is called BM25, which is an algorithm based on probabilistic retrieval model. Okapi BM25 model [13, 14] is a model developed from probabilistic model incorporates with term frequency and length normalization. The local weights are computed parameterized frequencies including term frequency, document frequency, and global weights as RSJ weights. Local weights are based on a 2D Poisson model while the global weights are based on the Robertson-Spärck-Jones Probabilistic Model. By reducing the number of parameters to be learned and approximated, based on these heuristic techniques, BM25 often achieves better performance compared to TF-IDF (term frequency - inverse document frequency).

3 Based on the domain ontology information retrieval model

The concept in domain ontology has a relation to other concepts simultaneously. The interrelation between concepts of the semantic relative network implements synonym expansion retrieval, semantic entailment expansion, semantic correlation expansion. We introduce a domain ontology information retrieval model to apply ontology into the traditional information retrieval model by query expansion to improve efficiency.

An illustration of structure for the information retrieval model is shown in Fig. 1.

The system consists of two parts: ontology document processing (including domain ontology servers, data source, document process unit and information database) and ontology document retrieval (including domain ontology server, query transition, custom process, and retrieval agent).

3.1 Ontology document processing

Document processing extracts useful information from an unstructured text message and establishes mapping relations between document terms and concepts based on domain ontology [15]. The document processing is shown in Fig. 2.

In the preprocessing procedure, each document in the document set implements vocabulary, analyzes words, and filters numbers, hyphens, and punctuations. Using a stop word list removes function words to leave useful words such as noun and verb [16]. Extracting stem words and removing the prefix and postfix improve the accuracy of retrieval. Finally, determining certain words as an index element expresses literature content conception.

Annotating semantic on a retrieved object by analyzing characteristic vocabulary builds the mapping relation between words and concepts. First, characteristic words are extracted and the weight of each word is calculated by counting word frequency to distinguish the importance of words. In this paper, the genetic algorithm is used to calculate the best weighting factor. In the end, it is applied to the actual retrieval system.

The system automatically learns weighted factor by genetic algorithm. It is a heuristic method which simulates biological evolution processes and through factor mutation eliminates the non-ideal factor sets and leaves the optimal factor set. The algorithm tries to maximize the fitness function as a parameter estimation to search a population consisting of the fittest individual; in our case, those are the parameters of weighted term in retrieving. In Fig. 3, the pseudo-code of genetic algorithm for weighted term frequency is described.

This algorithm simulates the evolution process by gradually adjusting weight factor and eliminating factor combination with a low fitness value. If the fitness result for one combination is lower than the other one, this group will be likely excluded in the next generation. To avoid the local optimization, we select many original generations and decrease the unqualified group time by time. In each iteration, the factor interval lies in $[w_i - 0.2, w_i + 0.4]$ to lower the negative factors. Fitness function $P(t)$ determines how fit an individual is with new weighted combination $(w'_{\text{tit}}, w'_{\text{key}}, w'_{\text{abs}})$. The traditional

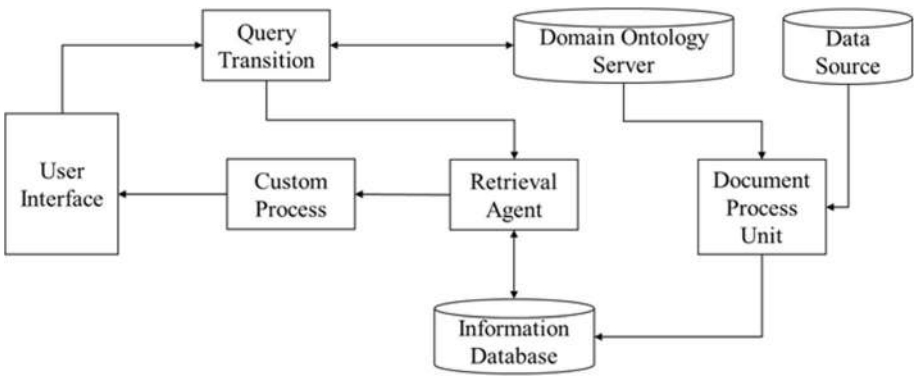

Fig. 1 An illustration of structure for information retrieval model

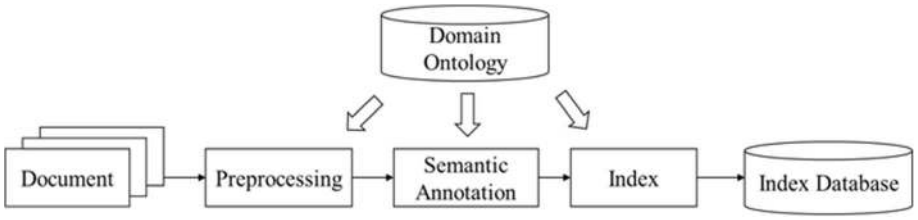

Fig. 2 Ontology document processing

```
(1) Input: paper list, expert rank list
(2) Output: weight of factor: wtit, wkey, wabs
(3) Initialize P(0)
(4) for t=0:#max_iteration_times do
(5)     // Evaluate Fitness
(6)     P(t)=f(wtit, wkey, wabs) – f(wtit', wkey', wabs')
(7)     //Select operation to P(i)
(8)     Slice=random()* ∑ P(t)
(9)     for i=1:#PopulationSize do
(10)        FitnessSoFar += P(i)
(11)if FitnessSoFar>slice
(12)            TheChosenOne = P(i)
(13)    end for
(14)    //Mutation operation to P(t)
(15)    for i=0:#ChromosomeSize do
(16)if random()<matitionRate
(17)        Chromo[i]+=(random()-0.5)*maxPerturbation
(18)if chrmo[i] > left
(19)            Chrmo[i]=right
(20)    if chrmo[i]>right
(21)            Chromo[i]=left
(22)    end for
(23)end for
```

Fig. 3 Pseudo-code for select weight factor by genetic algorithm

factor set is replaced by *P(t)* with higher fitness, then calculated with a query word for similarity of each papers and generated the rank list. The penalty function f is used to get the distance of the expert list.

Then, for each semantic meaning of ontology term, whether it exists in extracting characteristic vocabulary is checked. If the semantic exists, the document and weight with semantic term is calculated to manifest the text with semantic information.

After document feature extraction, document index based on the concept to reflect the internal relation between text index terms is established and ambiguity during annotation is excluded. An index based on the concept consists of feature words with their relation given by semantic parsing. Feature words connect through ontology instance and documents. The structure of the ontology concept index is shown in Fig. 4.

3.2 Ontology document retrieval

The procedure of document retrieval is listed below:

1) The user inputs search words or phrases in the search interface, then the system removes function words and reserves noun and verb. Term extraction from words is implement to get semantic conceptual words and phrases. The result is passed to the query transition module.

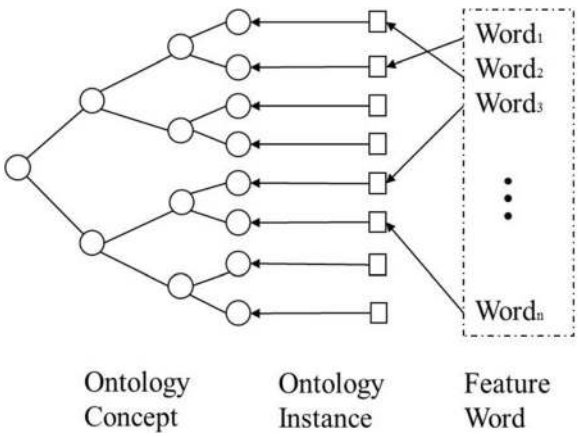

Fig. 4 Index structure based on the ontology concept

2) The query transition module sends consequence to the ontology server to search for a corresponding semantic concept, including hypernym, hyponym, synonym, and conceptual meanings [17]. If the word is not found in the ontology database, it returns to the user to help in adjusting the retrieval strategy.
3) For the matching concept in domain ontology, the query transition module implements search, semantic judgment, and query extension to add semantic information to query. The module submits query to a retrieval agent for searching. For words with an uncertain semantic message, execute a keyword matching method to search.
4) Handled by the custom process module, the user interface list query results according to exact word, synonym, hypernym, and hyponym words.

Before the retrieval process, the system executes semantic analysis for the user query request. Keyword is extracted from stop words and whether keyword belongs to ontology database is checked. Through combining concepts in ontology library, more semantic information is obtained by semantic reasoning. The pseudo-code of query semantic analysis algorithm is shown in Fig. 5.

After applying semantic analysis on user request, semantic information is able to be used in the retrieval strategy. The pseudo-code of information retrieval algorithm is shown in Fig. 6.

4 Experiment and results

4.1 The experimental design of the information retrieval model based on ontology

In order to evaluate the performance of the information retrieval model based on ontology, it is necessary to use ontology tools for modelling, such as Protégé [18] as an ontology modeling tool, ICTCLAS [19] as word segmentation tool, Jena [20] as semantic parsing tool, and Lucene as semantic indexing tool.

The data set contains 1000 scientific papers and papers from the IEEE digital library, which are used to extract the core concepts in the domain ontology. Then the final conceptualization system is established. The literature is divided into 10 groups. Each group contains 100 papers related to a query subject or key words (e.g., computer architecture and operating system). Therefore, 10 experts rank lists are available for retrieval.

The evaluation criterion considers the similarity of each paper towards every query word. For example, the mistaken sort term distance of the top neighboring

```
(1)  Input: query expression
(2)  Output: semantic concept set TermsWithOntologyConcept
(3)  Stem the query expression.
(4)  Get terms (q1, q2, ..., qn) using stop list.
(5)  Count terms weights (w1, w2, ..., wn) according to word frequency
(6)  if (terms q match the concepts in ontology database)
(7)      TermsWithOntologyConcept.add(q)
(8)  else
(9)      TermsWithoutOntologyConcept.add(q)
(10) if (TermsWithOntologyConcept.IsEmpty != true)
(11)     Decide the relation between terms.
(12) if (TermsWithoutOntologyConcept.IsEmpty != true)
(13)     foreach (term t in TermsWithoutOntologyConcept)
(14)         Nconceptmatching1.add(t)
(15) if (TermsWithOntologyConcept.IsEmpty == true &&
     TermsWithoutOntologyConcept.IsEmpty != true)
(16)     foreach (term t in TermsWithoutOntologyConcept)
(17)         Nconceptmatching2.add(t)
```

Fig. 5 Pseudo-code for the query semantic analysis algorithm

```
(1)  Input: query expression
(2)  Output: semantic concept set TermsWithOntologyConcept
(3)  if (TermsWithOntologyConcept.IsEmpty != true)
(4)     QueryExpansion(q1, q2, …, qi)
(5)     // expanse concepts on synonym, semantic implying and extension
(6)     Get retrieval results according to mapping in concepts and docs
(7)  returnNconceptmatching1
(8)     // returnNconceptmatching1 to users to adjust retrieval strategy
(9)  If (Nconceptmatching2.IsEmpty != true)
(10)    Calculate the similarity between (q1, q2, …, qi) and (w1, w2, …, wi)
(11) Get retrieval results according to similarity and docs
(12)    return Nconceptmatching2
(13)    // returnNconceptmatching2 to users to adjust retrieval strategy
```

Fig. 6 Pseudo-code for the information retrieval algorithm

papers is higher than the ones of lowest papers. The formula below is for how to collect the distance within rank list R and R':

$$P(t) = \frac{\sum_{i=1}^{n}[(n-i) \times \mathrm{dis}(i)]}{\sum_{i=1}^{\lfloor \frac{n}{2} \rfloor}[(n-i) \times i] + \sum_{\lfloor \frac{n}{2} \rfloor}^{n}[(n-i)^2]} \tag{3}$$

Here, n represents the paper numbers in the rank list. The dis(i) represents the position distance for paper i in the rank list and expert rank list. $P(t)$ represents the distance between the two rank lists of the denominator specification.

4.2 Analysis of experimental results

The genetic algorithm with simulated annealing method is compared in relation to iteration numbers and average distance of the rank list. The result is shown in Fig. 7. The X-axis time is the number of iterations in two algorithms, and the Y-axis average distance is calculated by formula (3) which shows the difference of the ranking list with expert list. After iteration for 200 times, the average distance is close to overall optimal. The

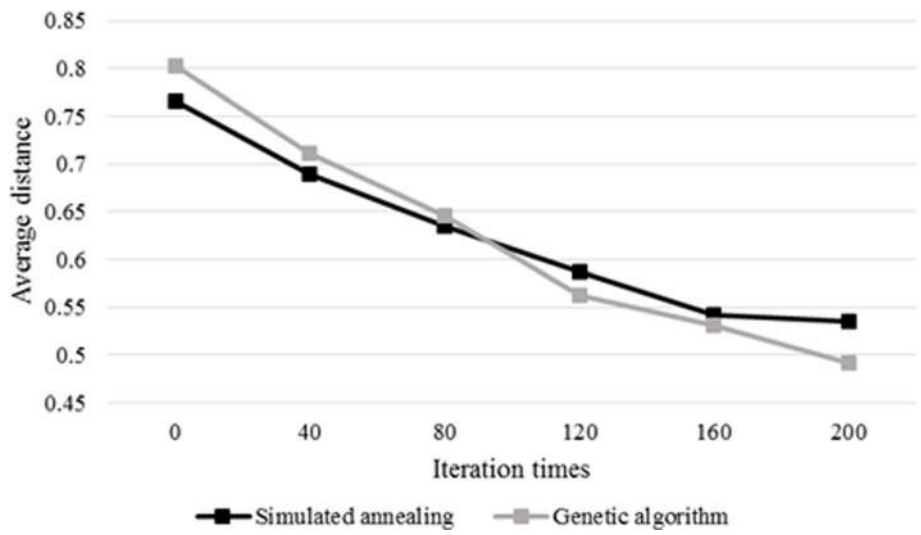

Fig. 7 Comparison of the simulated annealing and genetic algorithm in average distance and iteration times

Table 1 Precision and recall rate of ontology retrieval

Threshold	$\zeta = 0.5$		$\zeta = 0.55$		$\zeta = 0.6$	
Group num.	Precision	Recall	Precision	Recall	Precision	Recall
1	84.50%	83.36%	100.00%	82.45%	100.00%	81.85%
2	38.92%	100.00%	93.12%	100.00%	100.00%	51.00%
3	74.35%	100.00%	94.65%	94.43%	99.12%	94.65%
4	83.23%	100.00%	93.68%	100.00%	96.34%	45.74%
5	51.36%	100.00%	95.44%	100.00%	100.00%	100.00%
Average	66.47%	96.67%	95.38%	95.38%	99.09%	74.65%

algorithm deduces the optimized weight combination of factors which are $w_{tit} = 3$, $w_{abs} = 2$, $w_{key} = 0.6$.

The different threshold similarity value ζ is taken, in which $\zeta = 0.5$ means $\mathrm{sim}(S_q, S_j) \geq 0.55$. Every experiment counts retrieval documents set results $|A|$, ontology relevant documents $|B|$, and user query relevant document in the retrieval set $|A \cap B|$ to calculate the precision and recall rate. The result is shown in Table 1.

The precision rate improves with the threshold increasing. The precision rate reaches more than 99% when $\zeta = 0.6$. However, the recall rate only reaches 74%, which means the query result lost the critical information.

When $\zeta = 0.5$, the recall rate maintains a higher rate while precision remains low. Because of the system search, all the documents have ontology which relates with a query. The $\zeta = 0.55$ balance both the precision rate and recall rate.

5 Conclusion

In order to better satisfy users' retrieval needs and optimize the performance of information retrieval, domain ontology is introduced into the information retrieval system. In this paper, the information retrieval model based on domain ontology is proposed. The system includes document processing and ontology document retrieval with the ontology server, information database, and query transition and retrieval agent modules. We present a genetic algorithm to calculate the optimum combination of weighted factors of word frequency. Base on the evaluation criterion, we apply the system to query documents and compare with expert lists. The genetic algorithm shortens the distance compared with simulated annealing, and the ontology retrieval model exhibits a better precision and recall rate to understand the users' requirements.

Our future work is to further implement an automatic or semi-automatic method such as data mining to an establish ontology database to prevent the high difficulty in ontology establishment. And we may further implement modeling personalized query preference and return retrieval results according to different user query demands.

Abbreviation

IR: Information retrieval; VSM: Vector space model

Acknowledgements

This work is supported by the Science and Technology Research Project of Department of Education of Jilin Province (Grant 201657).

Funding

The Science and Technology Research Project of Department of Education of Jilin Province (Grant 201657).

Availability of data and materials

The data are included in this published article.

Author's contributions

The manuscript was written through contributions of the author. The author read and approved the final manuscript.

Author's information

Binbin Yu: Ph.D. candidate, College of Computer Science and Technology, Jilin University. Lecturer, College of Information Technology and Media, Beihua University. His research interests include Network security and so on.

Competing interests

The author declares that he has no competing interests.

Publisher's Note

Received: 21 November 2018 Accepted: 23 January 2019
Published online: 01 February 2019

References

1. M. Tang, Y. Bian, F. Tao, The research of document retrieval system based on the semantic vector space model. J Intelligence. **5**(29), 167–177 (2010)
2. C. Ma, W. Liang, M. Zheng, H. Sharif, A connectivity-aware approximation algorithm for relay node placement in wireless sensor networks[J]. IEEE Sensors J. **16**(2), 515–528 (2016)
3. Yang, X., Yang, D., Yuan, M. Scientific literature retrieval model based on weighted term frequency. Intelligent information hiding and multimedia signal processing (IIH-MSP), 2014 tenth international conference on. IEEE, 2014: 427–430
4. M. Xu, Q. Yang, K.S. Kwak, Distributed topology control with lifetime extension based on non-cooperative game for wireless sensor networks[J]. IEEE Sensors J. **16**(9), 3332–3342 (2016)
5. Y. Yan, J. Du, P. Yuan, Ontology-based intelligent information retrieval system. J, Software **26**(7), 1675–1687 (2015)
6. Y. Lu, M. Liang, Improvement of text feature extraction with genetic algorithm. New Technol. Library Inf. Serv. **4**(245), 48–57 (2014)
7. D. Vallet, M. Fernández, P. Castells, *An ontology-based information retrieval model. The Semantic Web: Research and Applications* (Springer, Berlin Heidelberg, 2005), pp. 455–470
8. C.D. Manning, P. Raghavan, H. Schütze, *Introduction to Information Retrieval (Vol. 1, p. 496)* (Cambridge University Press, Cambridge, 2008)
9. K.S. Jones, S. Walker, S.E. Robertson, A probabilistic model of information retrieval: Development and comparative experiments: Part 1. Inf. Process. Manag. **36**(6), 779–808 (2000)
10. Wong, S. M., Ziarko, W., & Wong, P. C. (1985). Generalized vector spaces model in information retrieval. In Proceedings of the 8th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 18–25). ACM
11. R. Baeza-Yates, B. Ribeiro-Neto, *Modern Information Retrieval*, vol 463 (ACM press, New York, 1999)
12. Premalatha, R., & Srinivasan, S. Text processing in information retrieval system using vector space model. In Information Communication and Embedded Systems (ICICES), 2014 International Conference on (pp. 1-6). IEEE
13. E. M. Voorhees, D. K. Harman (eds.), *TREC: Experiment and Evaluation in Information Retrieval*, vol 1 (MIT press, Cambridge, 2005)

14. R.M. Pereira, A. Molinari, G. Pasi, Contextual weighted representations and indexing models for the retrieval of HTML documents. Soft. Comput. **9**(7), 481–492 (2005)
15. Y. Zhang, K. Nan, Y. Ma, Research on ontology-based information retrieval system models. Appl Res. computer **8**(25), 2241–2249 (2008)
16. H. Kim, S.-w. Han, An efficient sensor deployment scheme for large-scale wireless sensor networks[J]. IEEE Commun. Lett. **19**(1), 98–101 (2015) [D] (Doctoral dissertation, Central South University)
17. J.J. Messerly, G.E. Heidorn, S.D. Richardson, W.B. Dolan, K. Jensen, *U.S. Patent No. 6,161,084* (U.S. Patent and Trademark Office, Washington, DC, 2000)
18. C. Keßler, M. Raubal, C. Wosniok, in *Smart Sensing and Context*. Semantic Rules for Context-Aware Geographical Information Retrieval (Springer, Berlin Heidelberg, 2009), pp. 77–92
19. Y.G. Cao, Y.Z. CAO, M.Z. JIN, C. Liu, Information retrieval oriented adaptive Chinese word segmentation system. J Software **3**, 17 (2006)
20. P. Castells, M. Fernández, D. Vallet, P. Mylonas, Y. Avrithis, in *On the Move to Meaningful Internet Systems 2005: OTM 2005 Workshops*. Self-tuning personalized information retrieval in an ontology-based framework (Springer, Berlin Heidelberg, 2005), pp. 977–986