

Ontology-based Semantic Similarity Transfer Algorithm

Ying Wang

School of Computer Science, Wuhan University, Wuhan, China
School of Computer Science and Technology, Wuhan University of Technology, Wuhan, China
Email: Vickyi_wy@163.com

Shihong Chen

School of Computer Science, Wuhan University, Wuhan, China
National Engineering Research Center for Multimedia Software, Wuhan University, Wuhan, China

Yimin Qiu

School of Computer Science, Wuhan University, Wuhan, China
School of Computer Science, Wuhan University of Science and Technology, Wuhan, China

Abstract—Most current network resources recommendation systems are used in a decision-making environment with low user participation, which can not effectively meet user expectations. The main reason is that the collection of user preferences is extremely difficult, which leads to the lack of information acquisition, and the lack of effective semantic similarity metric. Accordingly, this paper uses ontology for resource description to build a multiple inheritance hierarchical ontology model, and uses user preference model to generate attribute nodes, and construct a multiple inheritance graph model on account of the user personalization. Simultaneously, it construct preference transfer vector by user preference model, and then uses single-step transfer and multi-step transfer of SSTA to effectively extend evaluation from evaluated resources to the unevaluated resources. Experiment proves that SSTA has excellent recommendation effect compared with mature recommendation systems.

Index Terms—SSTA, multiple-inheritance-based graph model, preference transfer vector.

I. INTRODUCTION

User's online applications are mainly for two activities, search and browse. The user, who clearly knows the target to search, can accurately use the search operation to complete the acquisition of resources. When a user does not explicitly have a search target, he will browse all available resources. However, most of current networks are still based on a "one size fit all" approach to provide service, which rarely consider the differences between users, and provide all users with the same topology and manifestations. In fact, users usually have different interests and browsing habits, "one size fit all" obviously can not reflect and meet the needs of different users. Therefore, no matter from the aspect of the access to information or providing service, the resources

recommendation system that provides service with proactive and personalized information according to personal hobbies and interests has become the focus of the attention.

With the development of science and technology, some websites have tried to use resources recommendation system. The system is mainly based on the current user and / or other user preference requirements to assist users to browse the useful resources. For some reasons, this system is facing many problems. Resources recommendation system is usually used in a decision-making environment with low user participation. In this environment, users do not spend too much time to describe user preferences to the system. In addition, the system can not effectively deduce preference score of all the resources in system according to the known user preference model. Accordingly, the current resources recommendation system can not effectively meet user expectations is mainly because of the difficulties of user preferences acquisition and the lack of effective semantic similarity metric.

Based on different technical means, resources recommendation system can be divided into a rule-based, content-based recommendation system, collaborative filtering and hybrid mode. Content-based recommendation system is based on the calculated system resources and user preference similarity to recommend resources to the user, but its recommendation is always based on previous collection from the user's interests that were unable to match the user's latest interests. Moreover, this system does not take into account the information of other users with similar interests. The recommendation based on content is the mainstream technology of the personalized recommendation system. There are several successful cases, including a Web browser (Letizia is [1]), the news filtering (WebMate [2]), email filtering.

Collaborative filtering is to build a user model according to user evaluation of the resources, and compare the similarity between users for auxiliary resources recommendation. This method will divide the users with similar interests into a user group. The resources to a certain user's interest will be recommended to others in the group. Cold start problem has always been a major drawback of collaborative filtering technology. For new resources in the system, no users give the evaluation, it can not be recommended. Collaborative filtering is still the most successful recommendation system; typical successful examples are GroupLens [3], PHOAKS [4], Ringo [5], SiteSeer [6], etc. Hybrid recommendation system, which combines strengths of variety recommendation techniques, can achieve better recommendation accuracy. However, only few successful recommendation systems use hybrid resources recommendation technology, Stanford University's Fab [7] system is the most successful instance.

II. CONSTRUCTION OF ONTOLOGY STRUCTURE

This article uses ontology to describe resources to indicate the semantic relationships between the resources. The ontology has a semi-balanced multi-level structure, and in which each node represents a prototype concept.

Multiple inheritance hierarchical structure means that a concept can have more than one parent concept. When all the concepts have at most only one parent concept, and consider only the case of inheritance relationship, the ontology structure can be seen as a tree. However, in practical applications, a concept may have more than one parent concept. In this case, the multiple inheritance hierarchical structure becomes a directed acyclic graph (DAG). Figure 1 shows two ontology structures, (a) contains a simple tree structure, (b) contains a structure of DAG.

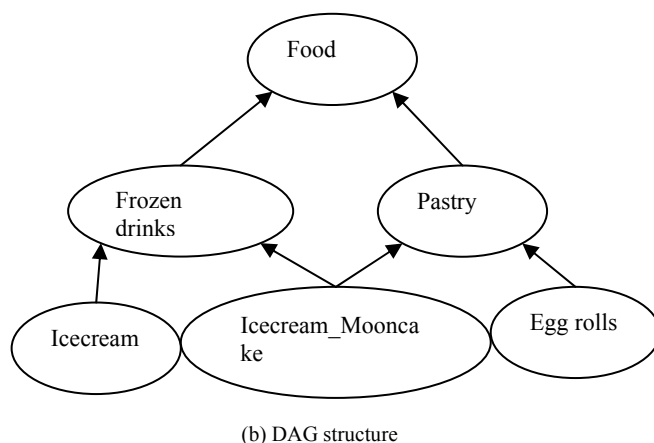
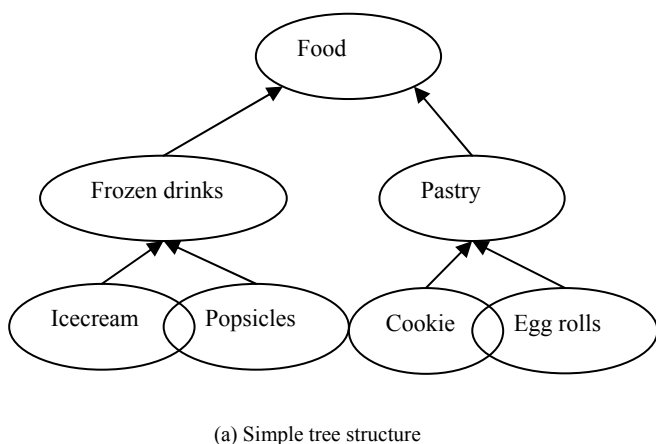


Fig. 1 Ontology instances with tree structure and graph structure.

In this kind of ontology, a resource may be instance of one or more concepts, and sides of ontology indicate the implicit or explicit character. Each concept can has a set of sub-concepts that can be called descendants, but all instances of a concept do not necessarily belong to the sub-concepts of it. For example, the ontology of the red wine, wine a, b, c respectively represent the instances of different concept that are the airless wine, the flavoring wine and special wine. In this case, some features are implicit. Because the wine a, b, c can be distinguished between a series of characteristics contain color description and certain taste.

A. Clustering Algorithm

In recent year's research, scholars in the field of data mining have proposed many unsupervised learning clustering algorithms. These algorithms are divided into six categories: fuzzy clustering algorithm, nearest neighbor clustering, hierarchical clustering, artificial neural network clustering, statistical clustering, and density-based clustering algorithm [8]. These algorithms are widely used in various fields, for example, the Reference [9] propose a new text clustering algorithm based on k-means and self-organizing model, the algorithm has higher accuracy and better stability. Reference [10], by means of weighted semantic similarity metric of is-a relationship, make ontology metric method originally used in hierarchical structure can be extended to the ontology with more general structure. Reference [11] analyzes the lack of fuzzy C-means (FCM) algorithm and genetic clustering algorithm and proposes a hybrid clustering algorithm based on immune single genetic and fuzzy C-means.

According to the ontology instances feature of the above analysis, it is necessary to study the hierarchical clustering. There are two different kinds of hierarchical clustering, agglomerative hierarchical clustering and divisive hierarchical clustering.

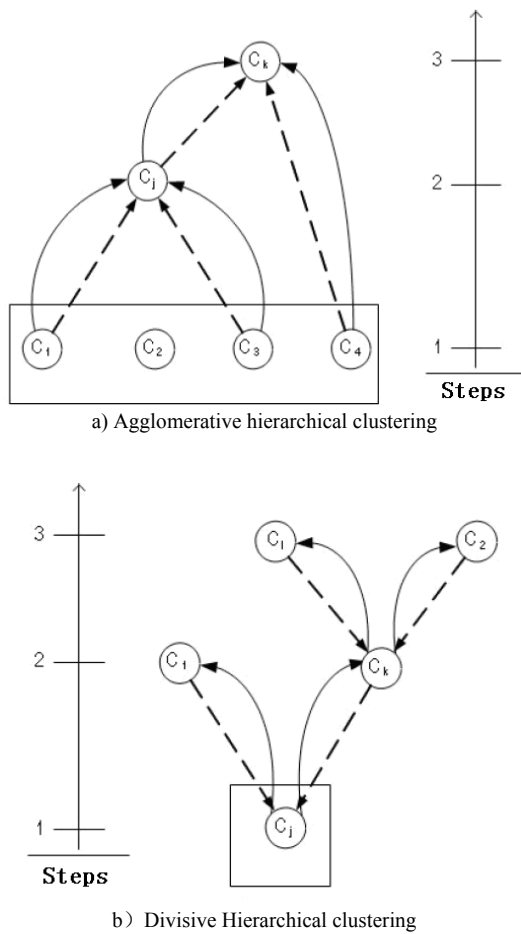


Fig. 2 Example of the first three steps in Figure 5.1 clustering algorithm

Main difference of the two methods is the order of hierarchical decomposition. Agglomerative hierarchical clustering is a bottom-up text clustering and its fundamental idea is to take each object as a cluster, these cluster by then merge to form a larger cluster repeatedly and finally stop merging until a termination condition. On the contrary, divisive hierarchical clustering is to see all objects as in the same cluster, then split the cluster, and stop dividing when a termination condition is reached. The biggest advantage of divisive hierarchical clustering is that it can guarantee low computational complexity, and low systems cost and in turn guarantee the practical completion of cluster operation of hundreds of thousands of objects. But the problems of this algorithm include local minimum problem and issues of serious dependence on the order of input resources.

B. Construction of Hierarchical Classification Glossary

Key step of the ontology construction is to construct a tree-structured hierarchy glossary. In the ontology structure, the nodes represent concepts, and the edges represent the inheritance relationships between concepts. Ontology concept instances are system resources. From the research on the relationship between users and resources in the process of resources recommendation, the conclusions are as follows:

- 1) A group of resources has become an instance of a concept because of some implicit or explicit common feature;
- 2) The concept is the abstraction of a set of characteristics, and characteristics exist between different concepts can be distinguished from each other;
- 3) User evaluation reflects the preferences of the users for some or all of the features of the resources.

According to the above facts, the similarity of characteristics between the resources can be extracted by comparing the resource evaluation of users. Based on this similarity, by clustering algorithm, all resources can be allocated to different cluster. The principle of the clustering algorithm is that the degree of difference between elements in a same cluster is small, and the degree of difference between elements in the different cluster is large. It hereby can be used to describe the fact that the same concept instances have common features, and different concepts are to be distinguished from each other by different characteristics. The process of clustering algorithms can be used to construct a tree-structured classification glossary.

Measurement of the distance between the two classes is an important part of the hierarchical aggregation algorithm; it mainly consists of two important parameters, similarity measurement method and connection rules. Euclidean distance is used as a similarity measurement method. Connection rules include single connection rules, full join rules, and average connection rules between classes, within-class average connection rules and Ward method. This connection rules can be defined as follows

[12] (wherein, containing $\|x - y\|$ is the Euclidean norm, n_i and n_k is the number of samples in class o_i and o_k , $C(n_i + n_k, 2)$ represents the extraction method of total number of different combinations of the two elements from the $n_i + n_k$ elements):

Single connection aggregation rules (slink) :

$$d(o_i, o_k) = \min_{x \in o_i, y \in o_k} \|x - y\| \tag{1}$$

Fully connected aggregation rules (clink) :

$$d(o_i, o_k) = \max_{x \in o_i, y \in o_k} \|x - y\| \tag{2}$$

Class average join aggregation rules:

$$d(o_i, o_k) = \frac{1}{n_i n_k} \sum_{x \in o_i} (\sum_{y \in o_k} \|x - y\|) \tag{3}$$

The Class Average joins aggregation rules:

$$d(o_i, o_k) = (1 + C(n_i + n_k, 2)) \sum_{x, y \in (o_i, o_k)} \|x - y\| \tag{4}$$

Ward method:

$$d(o_i, o_k) = \frac{1}{n_i + n_k} \sum_{x \in (o_i, o_k)} \|x - n\|^2 \quad (5)$$

Wherein, n is the center of the fusion clustering.

C. Multiple Inheritance Hierarchical Ontology Model

In the process of constructing ontology with the algorithm, the selection to merge or split is based on a single winner which enables the optimal solution of a certain condition function of the connection rules. A condition function neglects other possible effective suboptimal solutions. However, considering the suboptimal solutions can generate multiple inheritance hierarchical ontology models that resemble directed acyclic graph (DAG). Ontology with this structure can more accurately describe the connection of characteristics between concepts.

Based on this consideration, the existing algorithms were improved to generate ontology with multiple inheritance hierarchical structure and describe more abundant information and effectively assist in the ontology-based reasoning process. Research and experiments showed that using clink condition functions of the polymerization clustering algorithm can generate multiple inheritance hierarchical ontology model most effectively. The core idea of the algorithm is to improve the optimal solution to meet the condition function requirements, give an optimal clustering window, and then makes conditional function values fall clusters in the window, all the clusters are taken as suboptimal or optimal clusters, and merged with each other. Formal definition of the optimal clustering window is as follows:

Definition 1: given clustering sets $A, \forall A_i, A_j \in A$, calculate $dMax = \max(\min_{i \in A_i, j \in A_j} \|i - j\|)$, then the optimal window value $WindowValue = (1 - \lambda)dMax$, wherein $\lambda \in [0,1]$ is an adjustable parameter.

Algorithm 1: construct hierarchical structure ontology

Input: all the users' preference model, the leaves clustered number, window size factor λ , the number of clustered threshold θ

Output: hierarchical structure ontology

- 1 Resource – resource similarity matrix generated from the model of user preferences
- 2 Assigning to each resource i a separate cluster A_i , and initializes $A = A \cup A_i$
- 3 While $|A| > \theta$ do
- 4 $D \leftarrow \phi$
- 5 for all $\forall A_i, A_j \in A$ do
- 6 $D \leftarrow D \cup (d_{A_i, A_j} = \min_{i \in A_i, j \in A_j} \|i - j\|)$

- 7 end
- 8 $dMax \leftarrow \max(D)$
- 9 Taken min (D) corresponding to the two clustered A_i, A_j
- 10 $A_k \leftarrow merge(A_i, A_j)$
- 11 $A \leftarrow A / (A_i \cup A_j)$
- 12 $WindowVal = (1 - \lambda)dMax$
- 13 $D \leftarrow \phi$
- 14 for all $A_l \subset A_k, A_m \subset A$ do
- 15 $d_{A_l, A_m} = \min_{l \in A_l, m \in A_m} \|l - m\|$
- 16 if $d_{A_l, A_m} < WindowValue$ then
- 17 $A_n \leftarrow merge(A_l, A_m)$
- 18 $A \leftarrow A / A_l$
- 19 end
- 20 end
- 21 $A = A \cup A_n$
- 22end
- 23return A

III. SEMANTIC SIMILARITY MEASURE

Semantic similarity metric is very hot in the field of ontology research. The universal ontology have a level layer structure associated by is-a relationship, such as WordNet[13, 14], with about 82% of this relationship. The proportion of this relationship in GeneOntology[15] reached 87%. The Refrence [10] is using the weighted semantic similarity measure of is-a relationship. Experience shows that this algorithm is excessively dependent on the topology of the main ontology, and largely ignored by the node with the heterogeneity of the attribute information. Consequently, there exist great differences between the similarities calculated from the current algorithm and human understanding.

A. Formal Description

Given formalized representation of multiple attributes dynamic model, $G=(V, E, A, IC)$, wherein $V=\{1, 2, \dots, n\}$ is set of n nodes, $E = \{(i, j) | i, j \in V\}$ is set of m undirected edges; $A = \{A_1, A_2, \dots, A_n\}$ is set of attributes of n nodes; Wherein any one node $v_i \in V$ corresponds to an attribute vector $A_i = [v(a_1), v(a_2), \dots, v(a_l)]$; A_i is the l attributes set of node i ($A_i \in A$); $v(a_j)$ is value of node v_i 's attribute; $IC = \{IC_1, IC_2, \dots, IC_k\}$ is set of implicit characteristic attributes, and continuously updated by growing of user preference model. Implicit

characteristic attributes and dominant characteristic are used cooperatively with user preferences.

Characteristic attributes of user can be extracted via user preference model. These characteristic attributes added into the ontology graph structure as nodes are called attribute nodes. The t strip sides increased to connect attribute nodes and original resource nodes are called attribute sides. For example, extract feature attributes W1, W2 from preference model of user W, and add them to the structure of the original graph. Shown in Figure 3, connect nodes with this attribute in original graph; the dashed side is newly added WEi side.

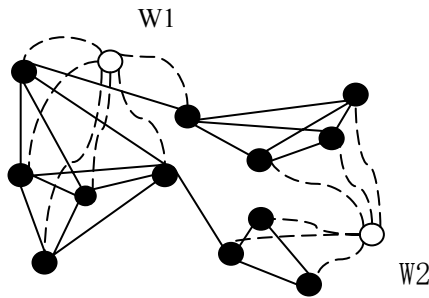


Fig. 3 Ontology graph structure with attribute node and attribute side

B. Semantic Similarity Measure

The expression of graph structure constituted by the original node is $G = (V, E, A, IC)$; V is the set of original nodes. Structure expression of $A = \phi, IC = \phi$ with attribute nodes is $G' = (V', E', A, IC)$, where W is the set of attribute nodes; $V' = V \cup W$, $E' = E \cup WE$, A, IC are dominant and implicit attribute nodes extracted from the user's preference model.

Definition 2: set of neighboring node of original nodes. $\forall v \in V$, its neighbor node set is $N(v)$.

$$N(v) = \{n \in V' | (v, u) \in E'\} \cup \{v\} \tag{6}$$

Definition 3: similarity measure of original nodes. $\forall va, vb \in V$. The structurized similarity of side $\langle va, vb \rangle$ is defined as

$$\varpi(va, vb) = \frac{|N(va) \cap N(vb)|}{\sqrt{|N(va)| |N(vb)|}} \tag{7}$$

The definition of the structurized similarity of side $\varpi(va, vb)$ shows that: the more common nodes share in two nodes, the greater association degree these two nodes will have. Public node includes the original concept node and the attribute node of the concept. Obviously, the structurized similarity of the side is symmetrical, ie $\varpi(va, vb) = \varpi(vb, va)$.

Definition 4: similarity measure of attribute node and original node. $\forall v \in V, \forall w \in W, V \cup W = V'$. The structurized similarity of side $\langle v, w \rangle$ is defined as

$$\omega(v, w) = \frac{1}{n(v)} \tag{8}$$

Wherein $n(v)$ is the attribute number of original node v , i.e. for each attribute of the original node, the correlation degree of attribute and original node is $1 /$ the total number of attributes.

C. Semantic Similarity Transfer Algorithm (SSTA)

The correlation between attributes extracted from user preference and original node is the key factor in obtaining a valid resource. In order to effectively extend the evaluated attributes and preference model between the original nodes to the unevaluated original node, and extract preference degree transfer vector, a single-step and multi-step transition can be used to deduce particular user preference value of all original nodes.

Definition 5: Single-step transition. The user u preference model: preference transfer vector $M = (m_1, m_2, \dots, m_n)$, n preference attributes value in the preference

mode. $w_i \in W, v_j \in V$ and $\langle w_i, v_j \rangle \in WE$. The similarity measure between attribute node and the original node becomes:

$$\omega(w_i, v_j) = \frac{1}{n(v)} * M_i \tag{9}$$

Definition 6: Multi-step transition between the original nodes without direct connection with attribute nodes. The user u preferences model: preference transfer vector $M = (m_1, m_2, \dots, m_n)$, N preference attributes value in preference model $w_i \in W, v_j, v_k \in V$,

and $\langle w_i, v_j \rangle \in WE, \langle v_j, v_{i1} \rangle \in E, \dots, \langle v_{in}, v_k \rangle \in E$, and no connection between sides w_i, v_k . Similarity measure of preference evaluation for extension:

$$\sigma(w_i, v_k) = \omega(w_i, v_j) * \prod \varpi(v_j, v_k) \tag{10}$$

For the unevaluated concept, in order to avoid the deviation of evaluation been constantly enlarged in the derivation of transfer process, it is necessary to shorten the transmission path. Therefore, there is a need to find the concepts with the most adjacent semanteme.

Go from v_j through the side with maximum value $\varpi(v_j, x)$, $\prod \varpi(v_j, v_k)$ is the product of similarity metric of the side through v_j to v_k .

IV. EXPERIMENTAL ANALYSES

Currently, there are many filtering algorithms [16-18]. The performance comparison between different collaborative filtering algorithms can be based on the Jester data set. The Jester Joke data set contains 1.7 million ratings of over 50, 000 users on 150 jokes. The score range is continuous real number from -10 to 10. These data are published by the University of California, Berkeley, Ken Goldberg. 10 and 40 resources were selected to be used to build the model, the x-axis represents the set value of the neighbors number used in the model, y-axis represents the resources recommendation accuracy. Select SSTA (Using “▼”) and traditional collaborative filtering algorithm based on similarity [19] (Using “▲”), collaborative filtering algorithm based on item rating prediction [19](Using “●”) as well as content-based collaborative filtering algorithm[20] (Using “◆”) for comparison.

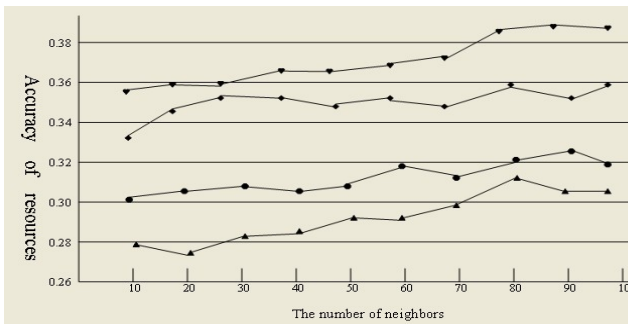


Fig. 4 Resource recommendation accuracy of 10 evaluation resources under different algorithms

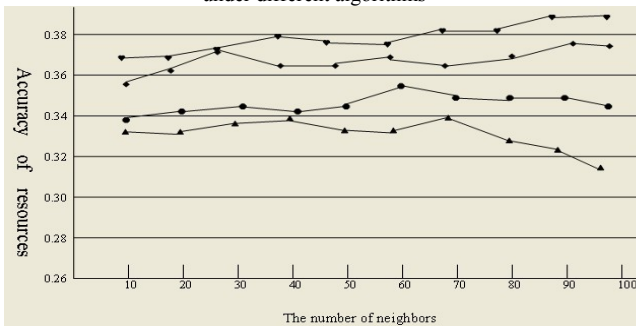


Fig. 5 Resource recommendation accuracy of 40 evaluation resources under different algorithms

From figure 4, figure 5 we can see that, the SSTA algorithm have a better recommendation effect than several common collaborative filtering algorithms. When the number of evaluation resources is small (10 in this experiment), the SSTA algorithm has excellent recommendation effect compared with other algorithms. With 40 user evaluation resources to build the ontology recommendation model, the SSTA algorithm still achieved very good recommendation effect.

The SSTA algorithm effectively avoids the problem-cold start problem in collaborative filtering technology. Firstly, add new resources to multiple inheritance hierarchy ontology model with algorithm 1; Secondly, the connected edge of the new resources and related attributes can be generated according to user preference

attribute model, then with single-step transfer and multi-step transfer algorithm, the degree of preference between the user's existing attributes and resources can be extended to the new resources.

V. CONCLUSION

This article uses ontology to describe resources, and construct personalized topology diagram with attribute nodes. These nodes are generated by dominant and implicit attributes extracted from the user's preference model. Simultaneously, it defines the algorithm of descriptive model and semantic similarity measure, and uses user preference transfer vector to effectively extend the evaluation from evaluated resources to the unevaluated resources. The experiment based on Jester data set and the comparisons with the mature recommendation models prove that this model can effectively improve the accuracy of the resource recommendation in practical use.

ACKNOWLEDGMENT

This work is supported by Natural Science Foundation of Hubei Province of China (Grant No. 2011CDB449)

REFERENCES

- [1]. Henry Lieberman.Letizia: "An Agent That Assists Web Browsing".In Proceedings of the International Joint Conference on Artificial Intelligence, Menlo Park, CA:AAAI Press, 1995.Pages 924~929.
- [2]. Chen, L., Sycara, K. "WebMate:A Personal Agent for Browsing and Searching". In Proceedings of the 2nd International Conference on Autonomous Agents and Multi Agent Systems, AGENTS'98(pp.132-129).ACM Press, 1998.
- [3]. Gordon, L.R., J.L.Herlocker, J.A.Konstan, D.Maltz, B.N.Miller, J. Riedl. "GroupLens: Applying Collaborative Filtering to Usenet News". Communications of the ACM, Vol.40(3): pp.77-87, 1997.
- [4]. Terveen, L.Hill, W.Amento, B.McDonald, D.Crester, "J.PHOAKS:A System for Sharing Recommendations", Communications of the ACM 40(3), March 1997, pp.59-62.
- [5]. Shardanand, U., Maes, P.(1995). "Social Information Filtering: Algorithms for Automating "Word of Mouth"". In Proceedings of CHI'95.Denver, CO.
- [6]. J.Rucker and M.J.Polano.Siteseer: "Personalized Navigation for the Web". Communications of the ACM, Vol.40(3): pp.73--75, March 1997.
- [7]. Balabanovic, M., Y.Shoham. "Fab: Content-Based, Collaborative Recommendation". Communications of the ACM, Vol.40(3): pp.66-72, 1997.
- [8]. Lin, C.-R., Chen, M.-S. "Combining partitional and hierarchical algorithms for robust and efficient data clustering with cohesion self-merging". IEEE Transactions on Knowledge and Data Engineering, 17(2):pp.145-159, 2005.
- [9]. Li Xinwu, "A New Text Clustering Algorithm Based on Improved K_means". JOURNAL OF SOFTWARE, VOL. 7, NO. 1, JANUARY 2012: 95-101
- [10]. Maguitman, A., M. F. R. H. V. A. "Algorithmic Detection of Semantic Similarity". In Proceedings of the International World Wide Web Conference, WWW'05, pp.107 - 116, 2005.

- [11]. Hongfen Jiang, Yijun Liu, Feiyue Ye, Haixu Xi, Mingfang Zhu, Junfeng Gu. JOURNAL OF SOFTWARE, VOL. 8, NO. 1, JANUARY 2013:134~141
- [12]. Marques JP, forward; Wu Yifei, translated pattern recognition - principles, methods and applications, Beijing: Tsinghua University Press, 2002.51-74 (in Chinese).
- [13]. Miller, G., Beckwith, R., Fellbaum, C., Gross, D., Miller, K. "Introduction to WordNet: An On-line Lexical Database". Technical report, Cognitive Science Laboratory, Princeton University, 1993.
- [14]. Leacock, C., C. M. "Combining local context and WordNet similarity for word sense identification". In Fellbaum, pp.265 – 283, 1997.
- [15]. GO, T. G. O. C. "Gene ontology: tool for the unification of biology". In America, N., editor, Nature Genetic, volume 25, pp. 25–29, 2000.
- [16]. Chickering D, Hecherman D. Efficient approximations for the marginal likelihood of Bayesian networks with hidden variables. Machine Learning, 1997, 29(2/3):181~212.
- [17]. Sarwar B, Karypis G, Konstan J, Riedl J. Analysis of recommendation algorithms for E-commerce. In: ACM Conference on Electronic Commerce. 2000. 158~167.
- [18]. Wolf J, Aggarwal C, Wu K-L, Yu P. Horting hatches an egg: A new graph-theoretic approach to collaborative filtering. In: Proceedings of the ACM SIGMOD International Conference on Knowledge Discovery and Data Mining. San Diego, 1999.201~212.
- [19]. DengAilin, Zhu Yangyong, ShiBole. A collaborative filtering recommendation algorithm based on item rating[J]. Journal of Software, 2003, 14(9): 1621-1628. (in Chinese).
- [20]. Kim BM, LiQ, Park C S, et al. A new approach for combining content-based and collaborative filters [J]. Journal of Intelligent Information Systems, 2006, 27: 79-91.

Ying Wang Hubei Province, China. She was born in 1979. She is a PhD student at wuhan university. Her research interests include knowledge engineering, artificial intelligence and decision-making analysis.

She is Lecturer in Wuhan University of Technology, has been involved in national and provincial scientific research, and the development of horizontal issues. Published a number of papers and college textbook.

Shihong Chen Hubei Province, China. Professor and doctoral supervisor of Whhan University, deputy director of the National Engineering Research Center for Multimedia Software, deputy director of Open Research Laboratory of the Ministry of Education multimedia software. 5 won the provincial and ministerial awards, hosted and participated in 14 projects. His research interests include Software Engineering and Multimedia technology.

Yimin Qiu She is a PhD student at wuhan university. Her research interests include knowledge engineering, and Multimedia technology. She works in wuhan university of science and technology.

Call for Papers and Special Issues

Aims and Scope.

Journal of Software (JSW, ISSN 1796-217X) is a scholarly peer-reviewed international scientific journal focusing on theories, methods, and applications in software. It provide a high profile, leading edge forum for academic researchers, industrial professionals, engineers, consultants, managers, educators and policy makers working in the field to contribute and disseminate innovative new work on software.

We are interested in well-defined theoretical results and empirical studies that have potential impact on the construction, analysis, or management of software. The scope of this Journal ranges from the mechanisms through the development of principles to the application of those principles to specific environments. JSW invites original, previously unpublished, research, survey and tutorial papers, plus case studies and short research notes, on both applied and theoretical aspects of software. Topics of interest include, but are not restricted to:

- Software Requirements Engineering, Architectures and Design, Development and Maintenance, Project Management,
- Software Testing, Diagnosis, and Validation, Software Analysis, Assessment, and Evaluation, Theory and Formal Methods
- Design and Analysis of Algorithms, Human-Computer Interaction, Software Processes and Workflows
- Reverse Engineering and Software Maintenance, Aspect-Oriented and Feature Interaction, Object-Oriented Technology
- Component-Based Software Engineering, Computer-Supported Cooperative Work, Agent-Based Software Systems, Middleware Techniques
- AI and Knowledge Based Software Engineering, Empirical Software Engineering and Metrics
- Software Security, Safety and Reliability, Distribution and Parallelism, Databases
- Software Economics, Policy and Ethics, Tools and Development Environments, Programming Languages and Software Engineering
- Mobile and Ubiquitous Computing, Embedded and Real-time Software, Database, Data Mining, and Data Warehousing
- Internet and Information Systems Development, Web-Based Tools, Systems, and Environments, State-Of-The-Art Survey

Special Issue Guidelines

Special issues feature specifically aimed and targeted topics of interest contributed by authors responding to a particular Call for Papers or by invitation, edited by guest editor(s). We encourage you to submit proposals for creating special issues in areas that are of interest to the Journal. Preference will be given to proposals that cover some unique aspect of the technology and ones that include subjects that are timely and useful to the readers of the Journal. A Special Issue is typically made of 10 to 15 papers, with each paper 8 to 12 pages of length.

The following information should be included as part of the proposal:

- Proposed title for the Special Issue
- Description of the topic area to be focused upon and justification
- Review process for the selection and rejection of papers.
- Name, contact, position, affiliation, and biography of the Guest Editor(s)
- List of potential reviewers
- Potential authors to the issue
- Tentative time-table for the call for papers and reviews

If a proposal is accepted, the guest editor will be responsible for:

- Preparing the “Call for Papers” to be included on the Journal’s Web site.
- Distribution of the Call for Papers broadly to various mailing lists and sites.
- Getting submissions, arranging review process, making decisions, and carrying out all correspondence with the authors. Authors should be informed the Instructions for Authors.
- Providing us the completed and approved final versions of the papers formatted in the Journal’s style, together with all authors’ contact information.
- Writing a one- or two-page introductory editorial to be published in the Special Issue.

Special Issue for a Conference/Workshop

A special issue for a Conference/Workshop is usually released in association with the committee members of the Conference/Workshop like general chairs and/or program chairs who are appointed as the Guest Editors of the Special Issue. Special Issue for a Conference/Workshop is typically made of 10 to 15 papers, with each paper 8 to 12 pages of length.

Guest Editors are involved in the following steps in guest-editing a Special Issue based on a Conference/Workshop:

- Selecting a Title for the Special Issue, e.g. “Special Issue: Selected Best Papers of XYZ Conference”.
- Sending us a formal “Letter of Intent” for the Special Issue.
- Creating a “Call for Papers” for the Special Issue, posting it on the conference web site, and publicizing it to the conference attendees. Information about the Journal and Academy Publisher can be included in the Call for Papers.
- Establishing criteria for paper selection/rejections. The papers can be nominated based on multiple criteria, e.g. rank in review process plus the evaluation from the Session Chairs and the feedback from the Conference attendees.
- Selecting and inviting submissions, arranging review process, making decisions, and carrying out all correspondence with the authors. Authors should be informed the Author Instructions. Usually, the Proceedings manuscripts should be expanded and enhanced.
- Providing us the completed and approved final versions of the papers formatted in the Journal’s style, together with all authors’ contact information.
- Writing a one- or two-page introductory editorial to be published in the Special Issue.

More information is available on the web site at <http://www.academypublisher.com/jsw/>.

Declaration and Translation of Spatial Access Control Policy <i>Aijuan Zhang, Jingxiang Gao, Jiuyun Sun, and Cheng Ji</i>	1132
Target Tracking Based on Optimized Particle Filter Algorithm <i>Junying Meng, Jiaomin Liu, Juan Wang, and Ming Han</i>	1140
Using Finite Element Software to Simulation Fracture Behavior of Three-point Bending Beam with Initial Crack <i>Chao Zhang, Peng Cao, Yixiang Cao, and Jianbo Li</i>	1145
Research and Design of Water Pump Diagnosis System Based on Wavelet Transform and DSP <i>Yongxian Song, Kangde Zhao, and Yuan Feng</i>	1151
FS-DS based Multi-sensor Data Fusion <i>Shunyuan Sun, Jing Gao, Minfang Chen, Baoguo Xu, and Zhiguo Ding</i>	1157
Fast Restoration of Warped Document Image based on Text Rectangle Area Segmentation <i>Kuo-Hsien Hsia, Shao-Fan Lien, and Juhng-Perng Su</i>	1162
A Modified Segmentation Approach for Synthetic Aperture Radar Images on Level Set <i>Qiangqiang Peng and Long Zhao</i>	1168
A Three-Dimensional Virtual Simulation System of Spinning Production Line <i>Yicun Hua, Yongsheng Ding, Kuangrong Hao, and Yanling Jin</i>	1174
Dimensionality Reduction of Multi-spectral Images for Color Reproduction <i>Ying Wang, Sheping Zhai, and Jianyuan Liu</i>	1180
The Development of a Mobile Terminal Middleware Platform Based on HTML5 <i>Xiaolin Yi and Yishuang Ning</i>	1186
On the System Dynamics of the Logistic Risk of Marketing Channels <i>Ying Ma and Fei Feng</i>	1194
Quay Crane Allocation of Container Terminal Based on Cluster Analysis <i>Yu Wang, Youfang Huang, Huiqiang Zheng, and Daofang Chang</i>	1201
Knowledge-based Links for Automatic Interaction with Programming Online Judges <i>Guojin Zhu and Yefeng Chen</i>	1209
Chinese Word Segmentation for Agriculture <i>Kui Fang, Weiqiong Bu, Wu Lou, and Luming Shen</i>	1219
Computer Aided Design and New Manufacturing Process of Spherical Hob <i>Xiaoqin Zhang, Zhanqi Hu, and Zhe Hao</i>	1227
Adaptability Evaluation of Enterprise Information Systems Based on Object-based Knowledge Mesh <i>Chaogai Xue</i>	1236
CCA Secure Threshold KEM Scheme Against Adaptive Corruption Attacks in Standard Model <i>Yuanju Gan, Licheng Wang, Jianhua Yan, and Yixian Yang</i>	1245
Applying Model Checking to Destructive Testing and Analysis of Software System <i>Hiroki Kumamoto, Takahisa Mizuno, Kensuke Narita, and Shin-ya Nishizaki</i>	1254
A P2P Network Model Based on Hierarchical Interest Clustering Algorithm <i>Fanbin Meng, Lei Ding, Sheng Peng, and Guangxue Yue</i>	1262
Ontology-based Semantic Similarity Transfer Algorithm <i>Ying Wang, Shihong Chen, and Yimin Qiu</i>	1268
