Automatic Discovery of Semantic Relations Based on Association Rule

Xiangfeng Luo, Kai Yan and Xue Chen

Joint Lab of Next-Generation Internet Interactive Computing, Shanghai University, Shanghai, China
Email: luoxiangfeng@163.com, kaikai20013@sina.com and xuechen@shu.edu.cn

***Abstract*—Automatic discovery of semantic relations between resources is a key issue in Web-based intelligent applications such as document understanding and Web services. This paper explores how to automatically discover the latent semantic relations and their properties based on the existing association rules. Through building semantic matrix by the association rules, four semantic relations can be extracted using union and intersection in set theory. By building a cyclic graph model, the transitive path of association relation is discovered. Document-level keywords and domain-level keywords as well as their parameters are analyzed to improve the discovery accuracy. Rules can be gained from the experiments to optimize the discovery processes for relations and properties. Further experiments validate the effectiveness and efficiency of the relation discovery algorithms, which can be applied in Web search, intelligent browsing and Web service composition.**

***Index Terms*—Algorithm, Association Rule, Semantic Relation, Transitivity**

1. Introduction

The advent of the Web significantly improves the quality of living life. Web services and Web mobile computing provides great convenience for daily life. There are large volumes of information in the Web and Web mobile systems, among which a rather part is abundant and out-of-order. So users have to take a lot of trouble and time to find the right answers from mass mistake information [1-3].

How to solve the above problem is a fundamental issue in Web services and Web mobile systems. One resolution aims to discover the semantic relations among information. For example, Semantic Web technologies propose the semantic relations of "is-a", "isPartOf" and "including", etc [4][5], and Semantic Link Networks (*SLN*) introduces the relations of "isPartOf", "subtype", "similar", "cause-effect", "sequence", "implication" and "instance" to facilitate Web services [6-7]. But it is hard for these technologies to automatically extract semantic relations from mass and out-of-order Web information [1] [3]. The other resolution is data mining that proposes a lot of algorithms to automatically extract association rules from databases such as Apriori and concept lattice [8-9], and recently Web Mining technology occurs to extract association rule from Web information [10]. The association rule is a general concept with the form of "if *A* then *B*", meaning that the occurrence of *A* will lead to the occurrence of *B*, i.e., $A \rightarrow B$. But the association rule is too general to provide clear and specific semantics, which determines that it cannot be widely used in intelligent Web services and Mobile systems.

For effectively and efficiently support intelligent Web services and Mobile systems, this paper explore how to obtain clearer and more specific semantic relations from the existing association relations. By analyzing and deducing the association rules, a series of algorithms are developed to automatically discover latent semantic relations and their properties. As an automatic discovery tool, these algorithms can be deployed in Web browser, Web services and Mobile computing.

The organization of the paper is as follows. Section 2 introduces the discovery process for the latent semantic relations. Section 3 presents the discovery process for the properties of semantic relations. Conclusions are drawn in the last section.

2. Discovery of Semantic Relations

2.1 Semantic Relations

In a set of association rules, assuming that keyword *A* is an antecedent who has association relations with a set of descendant keywords $U1=\{C, D, E, F, G, H\}$, and keyword *B* is an antecedent who has association relations with a set of descendant keywords $U2=\{C, D, E, F, G, F\}$ (see Figure 1).

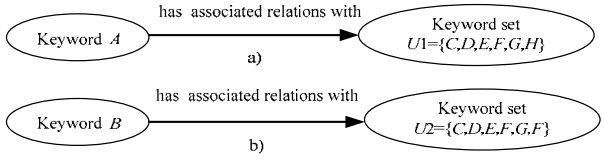


Figure 1. Discovery of Semantic Relations

Through analyzing $U1$ and $U2$, we obtain that $U1 \cap U2 = \{C, D, E, F, G, H\}$, $N(U1 \cap U2) = 6$ and $U1 \cup U2 = \{C, D, E, F, G, H, F\}$, $N(U1 \cup U2) = 7$, where $N(U1 \cap U2)$ is the

number of keywords in the set $U1 \cap U2$; $N(U1 \cup U2)$ is the number of keywords in the set $U1 \cup U2$.

We define the semantic degree α as the ratio of the number of N(U1∩U2) to N(U1∪U2), i.e., $\alpha = N(U1 \cap U2) / N(U1 \cup U2)$. So, the value of α between *A* and *B* is 6/7=0.86.

Different kinds of semantic relations can be discovered under the different values of α. If α is 0, then *A* and *B* has the empty relation, meaning that there is no relation between them. If α is 1, two situations occur. One is that $A \rightarrow B$ and $B \rightarrow A$ simultaneously in an association rule set. Thus *A* and *B* has the equivalence relation, i.e., $A \Leftrightarrow B$, meaning that *A* and *B* have the same ability to express semantics. If α is 1, but $A \rightarrow B$ or $B \rightarrow A$ are not satisfied, then *A* and *B* have the similar relation, meaning that *A* and *B* have the similar ability to express semantics. If α is less than 1, the similarity between *A* and *B* is decrease. The less the value is, the weaker the similarity. We define such a weak similarity as a new relation, cross relation, meaning that the semantics of *A* and *B* have a certain semantic intercross, but they are not totally similar. The discovery of cross relation has great significance in reality since it can provide users intercross information other than similar information for intelligent applications such as semantic searching and browsing. Thus, four kinds of semantic relations can be deduced from α, which we summarize as follows.

$$\alpha = \frac{N(U1 \cap U2)}{N(U1 \cap U2)} \begin{cases} = 0 & \text{then } A \text{ and } B \text{ have empty relation} \\ = 1 \begin{cases} A \rightarrow B \text{ and } B \rightarrow A, & \text{then } A \text{ and } B \text{ have equivalent relation} \\ \text{else}, & \text{then } A \text{ and } B \text{ have similar relation} \end{cases} \\ < 1 & \text{then } A \text{ and } B \text{ have cross relation} \end{cases}$$

We extract keywords from the documents in the proceeding of *WWW*2005, and then discover semantic relations between them (see in Table I).

TABLE I.
THE SEMANTIC RELATIONS BETWEEN KEYWORDS -READY PAPERS

"ontology" and "topic" have cross relation
"P2P" and "Peer-to-Peer" have equivalent relation
"Web" and "service" have cross relation
"presentation" and "document" have cross relation
……

2.2 Discovery Algorithm of Semantic Relations

This section studies how to automatically extract four kinds of semantic relations between keywords. These relations are deduced based on the association relations. To facilitate deduction, we only study the association rules with one antecedent and one descendant. From association relations, a latent matrix called semantic matrix can be extracted with more specific semantics, from which the four kinds of semantics relations can be automatically extracted.

1) Pre-Process

(1) Extracting keywords from a fixed set of documents;
(2) Using keywords to extract association rules from that set of documents; For example, 8 domain-level keywords are extracted from 20 documents published in *WWW*2005. Then association rules are extracted based on these keywords from the 20 documents. 8 keywords are "*page*", "*service*", "*data*", "*Web*", "*user*", "*search*", "*time*" and "*information*"**.** Several extracted association rules are shown in Table II.

TABLE II.
SEVERAL ASSOCIATION RULES EXTRACTED FROM 20 DOCUMENTS

If page then web	if search then information
If search then data	if search then page
If service then data	if time then search
…	…

2) Building semantic matrix

(3) Building *n*×*n* original semantic matrix based on *n* extracted keywords and the extracted association rules. *n* keywords are lined in the column and row respectively. If there has an association relation between two keywords, i.e., the antecedents and the descendants of association rules correspond to the keywords in the row and column respectively, and then the corresponding location in the semantic matrix is filled with 1. Otherwise, the location should be filled with 0. Table III illustrates the semantic matrix based on the keywords and extracted association rules in Table II.

TABLE III.
THE ORIGINAL SEMANTIC MATRIS

	page	service	data	web	user	search	time	information
page	1	0	0	1	0	1	0	0
service	0	1	1	1	0	0	0	0
data	1	1	1	1	1	0	1	0
web	0	0	1	1	0	0	1	0
user	1	1	1	1	1	1	0	0
search	1	0	1	0	1	1	1	1
time	0	0	0	0	0	1	1	1
information	0	0	0	0	1	0	0	1

(4) Using $\alpha = N(U1 \cap U2) / N(U1 \cup U2)$ to compute the semantic degree.
For example, the following computes the semantic degree between "*service*" and "*data*".
a) Along the row, we locate the keyword "*service*" and obtain its row vector {01110000} that corresponds to the "*service*" keyword set *U*1;
b) Along the row, we locate the keyword "*data*" and obtain its row vector {11111010} that corresponds to the "*data*" keyword set *U*2;
c) Use "*and*" logical operation to compute $N(U1 \cap U2)$;

*U*1	0	1	1	1	0	0	0	0
*U*2	1	1	1	1	1	0	1	0
*U*1∩*U*2	0	1	1	1	0	0	0	→ $N(U1 \cap U2) = 3$

d) Use "*or*" logical operations to compute $N(U1 \cup U2)$;

$$\begin{array}{lcccccccc} U1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ U2 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ \hline U1 \cup U2 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \end{array} \longrightarrow N(U1 \cup U2) = 6$$

e) Obtain

$$N(U1 \cap U2) \Big/ N(U1 \cup U2) = 3/6 = 0.5 .$$

Through the above analysis, we obtain the semantic degree between "*service*" and "*data*" is 0.5. Thus, the relation between "*service*" and "*data*" is cross relation.

According steps a) to e), we know that the cross relations hidden in the original semantic matrix are shown in Table IV.

TABLE IV.
CROSS RELATIONS EXTRACTED FROM THE ORIGINAL SEMANTIC MATRIX

"SERVICE" AND "DATA" HAVE CROSS RELATION
"SERVICE" AND "USER" HAVE CROSS RELATION
"SERVICE" AND "WEB" HAVE CROSS RELATION
"DATA" AND "USER" HAVE CROSS RELATION
"DATA" AND "WEB" HAVE CROSS RELATION
"SEARCH" AND "USER" HAVE CROSS RELATION

Since all the candidate documents come from the proceeding of *WWW*2005, through the analysis of Table IV, we can conclude that the extracted cross relations are obviously correct.

(5) To get more semantic relations, the first extracted semantic relations can be regarded as association rules to fill the original semantic matrix.

Table V shows the rebuilt semantic matrix by the above extracted semantic relations.

TABLE V.
RE-BUILDING SEMANTIC MATRIX

	page	service	data	web	user	search	time	information
page	1	0	0	1	1	1	0	0
service	0	1	1	1	1	0	0	0
data	1	1	1	1	1	0	1	0
web	1	1	1	1	1	0	1	0
user	1	1	1	1	1	1	0	0
search	1	1	1	1	1	0	1	1
time	0	0	0	0	1	1	0	1
information	0	0	0	0	1	0	0	1

We restrict the re-building times for the semantic matrix to be 2. On the one hand, rebuilding matrix will help discover more semantic relations. However, on the other hand, increase rebuilding times will increase the number of noise relations that will decrease the precision of the extracted dependent relations.

(6) Repeat steps (4) and (5), more semantic relations are obtained.

There are several important parameters that can influence the discovery process for semantic relations such as the number of the keywords extracted from the documents and the semantic degree of the semantic relation.

Assuming that the number of the keywords is *n*, when we compute $N(U1 \cap U2)/N(U1 \cup U2)$ by the "*and*" and "*or*" logical operations, the proposed algorithm complexity is O (n^2).

2.3 The impact of keyword number on performance

Keywords have two different types, document-level keywords and domain-level keywords. For example, in Grid domain, a document applies "*Grid*" technology to solve a problem of "*SARS*". So "*SARS*" belongs to document-level keywords and "*Grid*" belongs to domain-level keywords. Domain keywords have high occurrence frequency in documents. But domain keywords cannot replace document keywords. This section studies the impact of the two types of keywords on the discovery performance.

2.3.1. The impact of document-level keywords on performance

For the convenience of discussion, we set the re-building times of the semantic matrix is 1. We extract cross relations with their degrees larger than 0.6 as an example to study the impact of document-level keywords on performance.

a) Correlation between the number of document-level keywords and the precision of discovered cross relations

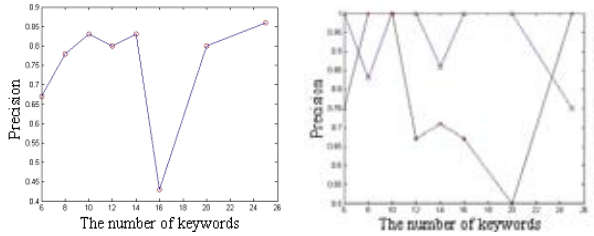
We use 6, 8, 10, 12, 14, 16, 20 and 25 document keywords to verify the discovery performance of semantic relations. Experimental results are shown in Figure 2(a). Several extracted cross relations are shown in Table VI, where the cross relations with bold font are noise relations.

TABLE VI.
SEVERAL EXTRACTED CROSS RELATIONS
(THE CROSS RELATION WITH BOLD FONT IS THE NOISE RELATION)

The number of keywords	The cross relations
…….	……..
10	specificity and property have cross relation stream and window have cross relation user and metric have cross relation client and proxy have cross relation **mapping and problem** do not have cross relation time and algorithm have cross relation
…….	……..

Figure 2(b) shows two repeated experiments. The blue line is the results with 10 randomly chosen documents from the proceeding of *WWW*2005 (*p12.pdf, p22.pdf, p33.pdf, p43.pdf, p54.pdf, p66.pdf, p76.pdf, p86.pdf, p97.pdf, p107.pdf*, this collection is denoted as ***D*1**). The black line is the results of another 10 documents (*p117.pdf, p128.pdf, p138.pdf, p148.pdf, p160.pdf, p170.pdf, p180.pdf, p190.pdf, p207.pdf, p199.pdf*, this collection is denoted as ***D*2**).

Comparing with Figure 2(a) and 2(b), the curve of the results is very high. No clear evidences show that the number of document-level keywords has a tight correlation with the experimental conditions. Generally speaking, the lower quantity of the document-level keywords is, the lower precision of extracted cross relations is.



(a) (b)

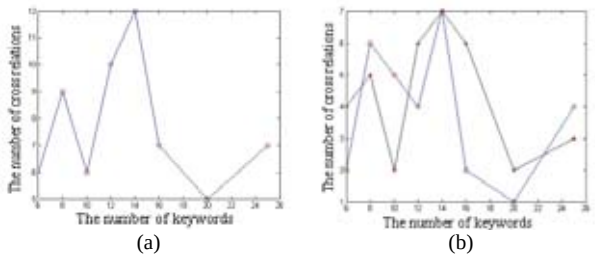
Figure 2. Correlation between the number of document keywords and the precision of extracted cross relations

What makes this result? Through analysis of document-level keywords, we summarize three reasons. First, the low quantity of document-level keywords cannot represent its own content, so the precision is low. Second, the high quantity of document-level keywords can not gain high precision since abundant document-level keywords are possibly the noise ones. Third, different documents have different appropriate quantity of document keywords. We believe that the three reasons can explain the high waves of the extracted precision based on document-level keywords.

b) Correlation between the number of document-level keywords and the number of discovered cross relations

We use 20 documents *D*1 and *D*2 as the experimetal data. Correlation between the number of document-level keywords and the number of the extracted cross relations are shown in Figure3(a). Figure 3(b) shows two repeated experiments. The blue line is the results by *D*1 and the black line is the results by *D*2.

Comparing Figure 3(a) and Figure 3(b), we observe that when the keywords around 14, the quantity of the extracted cross relations is near optimal. The extracted number of cross relations is tightly correlated with the document-level keywords under the expermental conditions.



(a) (b)

Figure 3. Correlation between the number of document-level keywords and the number of extracted cross relations

2.3.2. The impact of domain-level keywords on performance

If there are only a few number of domain-level keywords, some documents may have possibly a few keywords from which only a few semantic relations are extracted. To obtain more semantic relations, we set the re-building number to be 2, and only extract cross relations with degrees larger than 0.6.

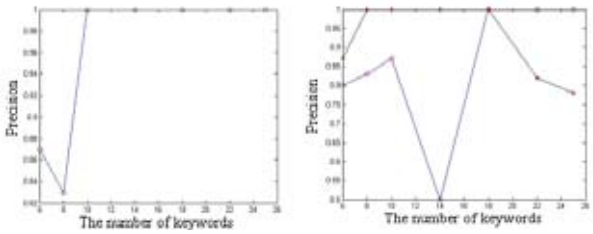
a) Correlation between the number of domain-level keywords and the precision of the discovered cross relations

We choose 6, 8, 10, 14, 18 and 25 domain-level keywords to verify the correlation between the number of keywords and the precision of semantic relations. Experimential results are shown in Figure 4(a). The repeated expermients with *D*1 and *D*2 are shown in Figure 4(b), in which the blue line is the results of *D*1 and the black line is the results of *D*2. Several extracted semantic relations are shown in Table VII.

TABLE VII.
SEVERAL EXTRACTED CROSS RELATIONS

The number of keywords	The cross relations
…….	……..
10	“user” and “page” have cross relation “user” and “search” have cross relation “page” and “data” have cross relation “algorithm” and “data” have cross relation
…….	……..

From Figure 4(a) and Figure 4(b), we conclude that the precision increases with the number of the domain keywords. Two reasons may lead to this tendency. First, the semantic relations between domain-level keywords are stronger than that of document-level keywords. Second, to achieve the same precision, more domain-level keywords are needed than document-level keywords. Therefore it is necessary for domain-level keywords to achieve better precision than document keywords.



(a) (b)

Figure 4. Correlation between the number of domain keywords and the precision

b) Correlation between the number of domain-level keywords and the number of discovered cross relations

To verify the correlation between the number of domain-level keywords and the number of extracted cross relations, three repeated experiments are shown in Figure 5, where the blue line are the results by *D*1 and the black line are the results by *D*2.

Comparing Figure 5(a) with 5(b), we conclude that the number of the extracted semantic relations decreases with the number of the domain-level keywords. Usually, the extracted relations increase with the numner of keywords. The main reason is that if we want to achive the same semantic degree with more keywords, then one antecedent keyword needs to induce more descendant keywords in *U* which need more stronger relations between keywords.

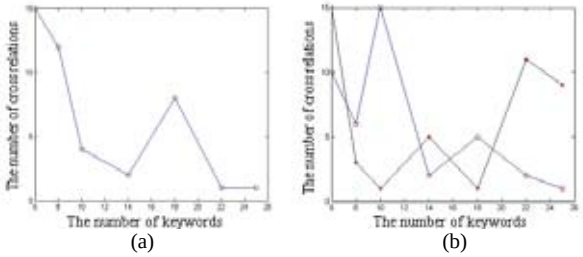


Figure 5. Correlation between the number of domain-level keywords and the quantity of extracted cross relations

This reason explains why the high number of domain keywords have higher precision and few semantic relations under the experiment condition of the same semantic degree. If we want more semantic relations in the high quantity of domain keywords, then we need to decrease the threshold of the semantic degree. Although the quantity of the extracted semantic relations decrease with the number of the domain keywords, the extracted semantic relations are closer to the practice than the low number of the domain keywords.

2.4 The impact of semantic degree on performance

2.4.1 Document-level keywords based correlation between semantic degree and precision of discovered cross relations

According to the analysis of 2.3.1, we choose 17 document keywords, the set of the semantic degree α are {0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1} and the re-bulidings times of semantic matrix is 1 as the experimental condtions to verify the document keywords based correlation between the semantic degree and the precision of extracted cross relations. Experimental results are shown in Figure 6.

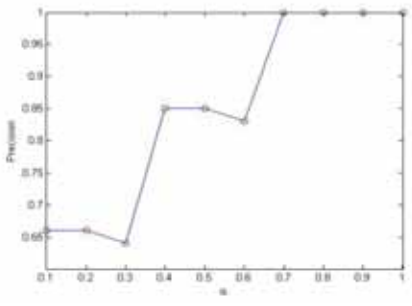


Figure 6. Document keyword based correlation between the semantic degree and the precision

Two conclusions are gained from Figure 6 and Figure 7.
(1) Extracting precision of cross relations increases with the increase of α.
(2) The number of the cross relations decreases with the increase of α. Larger α will lead to higher precision and fewer cross relations.

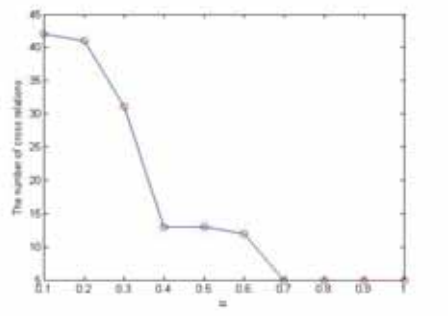


Figure 7. Document keywords based correlation between the semantic degree and the number of cross relations

From Figure 6 and Figure 7, we know that when α is 0.7,0.8, 0.9 and 1.0, the same number and the same precision of cross relations can be discovered. Through checking the middle experimental results, we find that there are two types of association rules, i.e., $A \rightarrow B$ and $B \rightarrow A$. But in the real situation, two rows with the same values rarely occur.

2.4.2. Domain-level keywords based correlation between the semantic degree and precision of discovered cross relations

According to the analysis of 2.3.2, we choose 22 domain-level keywords, the set of the semantic degree α are {0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1} and the re-buliding time of semantic matrix is 1 as the experiment conditions to verify the domain-level keywords based correlation between the semantic degree and the precision of extracted cross relations. Experimental results are shown in Figure 8 and Figure 9 respectively. The following conclusions are gained.

(1) The higher the semantic degree α will lead to higher precision and fewer number of the discovered cross relations;
(2) The total precision is higher than the document keywords because the keywords belong to domain keywords;

Since higher α leads to lower precision, it is necessary to make balance between the precision and the number of the cross relations. In our experimental conditions, α in [0.4, 0.5] is near optimal.

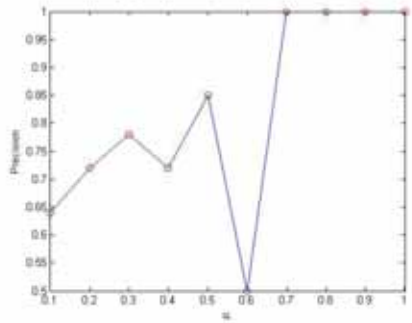


Figure 8. Document keywords based correlation between the semantic degree and the precision

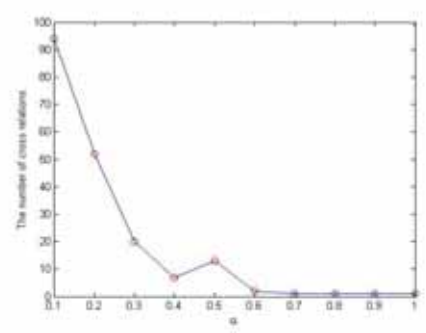


Figure 9. Document keywords based correlation between the cross degree and the number of cross relations

3. Properties of Semantic Relations

Four latent semantic relations (empty relation, equivalent relation, similar relation and cross relation) can be deduced based on the association rules. Together with association relation, there are totally five semantic relations. This section studies the properties of these semantic relations. Obviously, except for association relation, the other four relations are all symmetric. And equivalent relation and association relation are transitive. Compared with symmetry, transitivity is a more important property since in real applications it can help deduce latent semantic relation between unknown data, whereas symmetry cannot help deduction. Both association relation and equivalent relation have transitivity. Due to space constraints, we here introduce the discovery process of transfer path for association relation, and equivalent relation has similar discovery process.

3.1 Transitivity of Association Relation

The transitive path is in the form $A \rightarrow C \rightarrow G \rightarrow D \rightarrow \ldots \rightarrow E$ in a set of keywords $\{A, B, C, D, E, F, G, H \ldots\}$. Thus, $A \rightarrow E$ can be deduced, i.e., *A* and *E* have associated relation. All those transitive path link together in keyword-level or document-level, and gradually evolve into a Semantic Link Networks [6], which can used into intelligent applications such as Web services composition and semantic searching or mobile services, etc. For example, the algorithm to be presented in section 3.2 can extract transitivity from the documents that are randomly chosen from the proceedings of *WWW*2005 (see in Table VIII).

We use the confidence and the support of association rule [8] as the criterion to validate the correctness of the extracted transitive path. Taking Figure 10 as an example, the values above the arrow denote the confidence, and the values below the arrow denote the support. If the value is too small, then end the transitive path. For example in Figure 10, the confidence value and the support value from "*service*" to "*user*" are too small, thus this path ends in "*user*". High confidence and high support can ensure the correctness of the extracted transitive path.

Figure 10. Extraction of transitive path

TABLE VIII.
SEVERAL EXTRACTED TRANSITIVE PATH

"tree" and "ontology" have association relation	tree->page->user->search->server ->query->data->domain->semantic ->service->web->ontology
"semantic" and "domain" have association relation	semantic->data->user->page->tree ->ontology->domain
"search" and "user" have association relation	search->ontology->web->process->data ->source->paper->server->page->user

3.2 Extraction of Transitive Path

In this section, we use cyclic graph to extract transitive path from documents. For the convenience of discussion, we only extract the association rules with one antecedent and one descendant.

Assuming that antecedent of the association rule is denoted by *ifkey* and the descendant is denoted by *thenkey*. The extracting process is as following.

(1) 8 domain keywords are extracted from 20 documents in the proceeding of *WWW*2005. They are "*page*", "*service*" ,"*data*", "*Web*", "*user*", "*search*" ,"*time*" and "*information*"**;**

(2) Use 8 keywords to extract association rules from 20 documents. Several extracted association rules are shown in Table1;

(3) Cyclic graph is generated according to the extracted association rules (see Figure 11);

(4) Obtain one step association rule *x* in the cyclic graph;

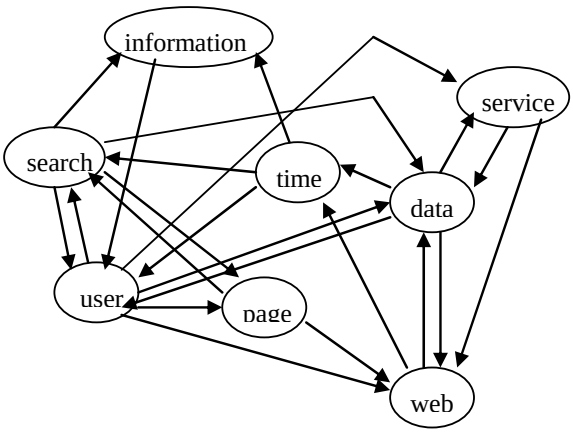


Figure 11. the generation of the cyclic graph according to the association rules

(5) Obtain next association rule *y* whose antecedent is as *x.ifkey*, that is *x.ifkey = y.if key* and *x.thenkey*! = *y.thenkey*. If *y* is empty, then there is no transitive path, then the process ends.

(6) Analyze association rule *y'* whose antecedent is as *y.ifkey*. And set *y= y'*. If *y* is empty, then there is no association relation. Jump out.

(7) Take *y.thenkey* as antecedent, take out *y'* in turn, set *y= y'*.
If *y.thenkey = x.thenkey*, then we find a transitive path. Otherwise, find the next rule *y'* by the antecedent *y.ifkey*, i.e., *y.ifkey = y'.ifkey* and *y.thenkey! = y'.thenkey*; in which if *y* is empty, then return to step (6), otherwise, set *y= y'* and call step (7);
(8) Add the confident and the support of association rules into the corresponding transitive path. If the confident and the support are less than a predefined threshold, then this transitive path ends.

3.3 The number of keyword impact on the length of transitive path

3.3.1 The impact of document-level keywords on the length of transitive path

We choose 6, 8, 10, 14, 16, 20 and 25 document keywords from 20 documents to verify the correlation between the number of document keywords and the quantity of transitive path. Several results are shown in Table IX and Figure 12 respectively. Figure 13 shows that the length of transitive path from a single document is short or even zero under the condition of a large number of document keywords. The reason is that in a finite collection of the association rules, if each document chooses 6 document keywords, then the whole document keywords maybe more than 100. Large numbers of document keywords will result in short transitive path in the finite set of the association rules.

TABLE IX
Several extracted transitive path based on document-level keywords

"tree" and "ontology" have association relation	tree->page->user->search->server->query ->data->domain->semantic->service->web ->ontology
"semantic" and "domain" have association relation	semantic->data->user->page->tree ->ontology->domain
"search" and "user" have association relation	search->ontology->web->process->data ->source->paper->server->page->user

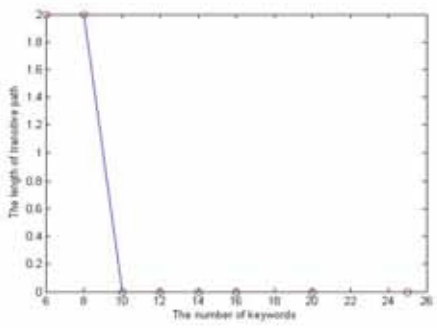


Figure 12. Correlation between the number of keywords and the length of the transitive path

3.3.2 The impact of domain-level keywords on the length of transitive path

We choose 6, 8, 10, 14, 18 and 25 domain keywords to verify the correlation between the number of domain keywords and the length of transitive path respectively. Several results with 20 documents are shown in Table X and Figure13 with blue line. Repeated experiments with the data of *D*1 are shown in Figure 13 with black line.

Through analyzing Figure 13, some conclusions are draw.
(1) Document number influence the length of the transitive path;
(2) There should be a moderate number of documents that can achieve the longest path length of transitivity.

TABLE X
Several extracted transitive path based on domain-level keywords

The number of keywords	Latent association relation
…….	……..
10	"user" and "page" have association relation user->search->algorithm->service->time->server-> data->web->ontology->page "data" and "user" have association relation data->web->service->time->user ……
…….	……..

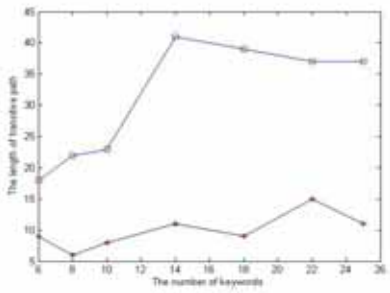


Figure 13. Correlation between the number of keywords and the length of transitive path (blue line uses the 20 documents and the black line uses 10 documents in D1)

4 Conclusion

Based on the general association rules, this paper built a semantic matrix to deduce four latent semantic relations, empty relation, equivalent relation, similar relation and cross relation, then analyze their properties. By building a cyclic graph model, the transitive path of association relation is discovered. A semantic link network (*SLN*) can be easily and automatically built with these semantic relations and their properties in both keyword level and document level. Such an *SLN* can guide search and browse among Web resources, and can help automatic composition of Web services, even can find the right cooperation partners in mobile systems. Experimental results validate the effectiveness and efficiency of the relation discovery algorithms, which have promising application prospect in Web services and mobile system.

To further optimize the discovery processes, we obtain the following rules from the experiments.
1) Document-level keywords are not tightly related with

the accuracy of the discovered semantic relation;
2) There should be a moderate number that can achieve the biggest number of semantic relations.
3) Using domain-level keywords to discover latent semantic relation is more accurate than document-level keywords.
4) The quantity of discovered semantic relations decreases with the increase of the number of domain-level keywords;
5) The threshold of the cross degree directly affects the accuracy and the quantity of relation discovery. The higher the threshold is, the higher the accuracy will be and the less the extraction quantity is.

We can get the following summaries from the experiments to guide the discovery of the transitivity of the association relation.

1) It is hard to extract a long length of transitive path between documents by document-level discovery algorithm;
2) The number of the documents influence the length of transitive path;
3) There should be a moderate number that can achieve the longest path length.

ACKNOWLEDGMENT

Research work is supported by the National Science Foundation of China (grants 90612010 and 60402016), the National Basic Research Program of China (973 project no. 2003CB317008).

References

[1] Zhuge H. The Knowledge Grid, World Scientific Publishing Co., Singapore, 2004.
[2] Zhuge H, Luo X. Automatic generation of document semantics for the e-Science Knowledge Grid. Journal of Systems and Software 2006; 79:969–983.
[3] Luo X, Hu Q, et al. Discovery of textual knowledge flow based on the management of knowledge maps. Concurrency and computation: practice and experience. 20:1791–1806, 2008.
[4] Hepp, M. Semantic Web and semantic Web services: father and son or indivisible twins? IEEE Internet Computing, 10(2), 85 – 88, 2006.
[5] Li D., Huan L. The Ontology Relation Extraction for Semantic Web Annotation. The 8th IEEE International Symposium on Cluster Computing and the Grid. 534-541. 19-22 May 2008.
[6] Zhuge H, Communities and Emerging Semantics in Semantic Link Network: Discovery and Learning. IEEE Transactions on Knowledge and Data Engineering. Manuscript ID: TKDE-2007-07-0321.R1 1.
[7] Zhuge H, Jia R. et al. Semantic Link Network Builder and Intelligent Semantic Browser. Concurrency and computation: practice and experience Concurrency. 16, 1453–1476, 2004.
[8] Agrawal R., Imielinski T., et al. Mining association rules between sets of items in large databases. In Proc. of SIGMOD, 207–216, 1993.
[9] Luo X, Fang N., et al. Semantic representation of scientific documents for the e-science Knowledge Grid. Concurrency and Computation: Practice and Experience. 20(7), 839-862, 2008.
[10] Kao H., Lin S., et al. Mining Web informative structures and contents based on entropy analysis. IEEE Transactions on Knowledge and Data Engineering. 16(1), 41-55, 2004.

Xiangfeng Luo received the master degree in Hefei University of Technology in 2000 and the Ph.D. degree in the same school in 2003. He was a post doctor at the China Knowledge Grid Research Group of Institute of Computing Technology (ICT) in Chinese Academy of Sciences (CAS) from 2003 to 2005. He is in charge of several projects, such as National Science Foundation of China (NSFC), the Great Research Project of NSFC, the Innovation Foundation of ICT, etc.

Professor Luo is currently an associated professor at School of Computers in Shanghai University, and his main research interests include the Web content analysis, Semantic Networks, Web knowledge flow, Semantic Grid and Knowledge Grid. His publications appeared in Concurrency and Computation: Practice and Experience, Journal of Systems and Software and Journal of Computer Science &Technology.

Kai Yan received the BS degree at Shanghai University. He is a member of the Joint Lab of Next-Generation Internet Interactive Computing of Shanghai University.

Xue Chen received the MS and PhD degree at the Institute of Computing Technology, Chinese Academy of Science. She is now an assistant professor in the Department of Computer Science at Shanghai University. Her interests are in the areas of Peer-to-Peer systems and Semantic Networks. Her publications appeared in IEEE Transactions on Knowledge and Data Engineering and IEEE Transactions on Parallel and Distributed Systems.