

Communities and Emerging Semantics in Semantic Link Network: Discovery and Learning

Hai Zhuge, *Senior Member, IEEE*

Abstract—The World Wide Web provides plentiful contents for Web-based learning, but its hyperlink-based architecture connects Web resources for browsing freely rather than for effective learning. To support effective learning, an e-learning system should be able to discover and make use of the semantic communities and the emerging semantic relations in dynamic complex network of learning resources. Previous graph-based community discovery approaches are limited in ability to discover semantic communities. This paper firstly suggests the Semantic Link Network SLN, a loosely coupled semantic data model that can semantically link resources and derive out implicit semantic links according to a set of relational reasoning rules. By studying the intrinsic relationship between semantic communities and the semantic space of SLN, approaches to discovering reasoning-constraint, rule-constraint and classification-constraint semantic communities are proposed. Further, the approaches, principles and strategies for discovering emerging semantics in dynamic SLN are studied. An e-learning environment incorporating the proposed approaches, principles and strategies to support effective discovery and learning is suggested.

Index Terms—Community discovery, e-learning, emerging semantics, semantic community, Semantic Link Network.



1 INTRODUCTION

The World Wide Web provides not only a worldwide information sharing platform but also plentiful contents for Web-based learning. However, the Web's hyperlink architecture interconnects Web resources for browsing freely rather than for learning effectively. So how to effectively organize learning resources of various types to support e-learning in a semantic context becomes a challenge.

The following three issues are critical for a Web-based e-learning system to organize resources for effective learning:

1. A self-organized semantic data model that can effectively organize resources and loosely couple query and the structure of organizing resources.
2. Automatically discovering various semantic communities in the network of semantically linked resources so that operations on resources can be efficiently executed.
3. Automatically discovering emerging semantic relations in a dynamic network of resources so that queries on various relations can be answered effectively.

To resolve the first issue, we propose the Semantic Link Network SLN, a self-organized semantic data model for semantically organizing resources, which can be abstract concepts or specific entities such as texts, images, videos and audios. For example, learning resources can be linked to their classes by *instanceOf* link, and a class

can be linked to its super-class by *subtype* link. The SLN has the following features: (a) It reflects various semantic relations between classes, between relations and between entity resources. (b) It is a semantics-rich self-organized network. Any resource can be semantically linked to any other resources. There is no strict structure like relational databases. (c) It can derive out implicit semantic links based on a set of reasoning rules. (d) The semantics of the network keeps evolving with various operations on the network.

To resolve the second issue, this paper investigates the approaches to discovering semantic communities in SLN according to the features of SLN. Previous graph-based community discovery approaches have the following three major limitations when applied to SLN: (a) The effect is unsatisfied because ordinary graphs cannot reflect semantic relations and it is incapable of supporting relation reasoning and relation query, which is often required in real applications. (b) The meaning of the discovered communities is unknown unless we assign semantics on edges and nodes. (c) Their costs are too high to be used in large-scale network.

To resolve the third issue, this paper proposes two approaches to discovering the emerging semantics in dynamic SLN. It is very useful for e-learning systems to know the emerging semantics of a community and the emerging semantic relations between resources in the SLN evolving with interaction between users and e-learning system.

Incorporating above solutions, an e-learning environment can support discovery and learning in a semantic context. The final part of this paper introduces the feature of such an e-learning environment.

• H.Zhuge is with the Knowledge Grid Research Group, Key Lab of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences. E-mail: zhuge@ict.ac.cn.

Manuscript received 2 July 2007.

2 RELATED WORK

2.1 On Web-based Learning

The study of Web-based learning mainly concerns the applications of computing technologies especially the Web technologies to support effective learning on the World Wide Web. An e-learning system that can select the sequence of Web resources and link them into a coherent, focused organization for instruction is introduced in [10]. It can automatically generate individual learning path from a repository of XML-based Web resources. A knowledge tree is suggested as an architecture for adaptive e-learning based on distributed reusable intelligent learning activities [8]. Many new technologies for e-learning over the Internet are introduced in [19]. The interested content and the basic semantic relation in large-scale network of contents is a basic issue of realizing effective Web-based learning.

2.2. On Community Discovery and Relation Query

Particular structure often exists in the networks such as the World Wide Web, citation networks, email networks, food webs, social networks and biochemical networks: nodes are often clustered into tightly-knit groups, and edges are dense within groups and loose between groups. Such a structure reflects the characteristic of human group behavior of sharing information. Research on discovering network community has been done as graph partitioning in graph theory, computer science, hierarchical clustering in sociology and geographical partition [4, 11, 14, 16, 17, 28, 36, 40]. One type of algorithms operates on the whole graph and iteratively cuts appropriate edges. They divide the network progressively into smaller disconnected communities. The key step of the divisive algorithms is the selection and removal of appropriate edges connecting communities. The idea of betweenness centrality is early proposed by Freeman [11]. Girvan and Newman proposed a divisive algorithm (called GN algorithm) to select the edges to be removed according to their 'edge betweenness' [7, 12, 22, 23, 24, 25], a generalization of the centrality betweenness [11]. Considering the shortest paths between all node pairs in a network, the betweenness of an edge is the number of the shortest paths through it. It is clear that when a graph is made of tightly bounded and loosely interconnected clusters, all of the shortest paths between nodes in different clusters need to go through these few inter-cluster connections, which therefore own large betweenness values. The GN algorithm consists in the computation of the edge betweenness for all edges in the graph and in the removal of those with the highest betweenness value. The iteration of this procedure leads to the split of the network into disconnected groups, which in turn undergo the same procedure until the whole graph is divided into a set of isolated nodes or a predefined condition (e.g., the number of expected communities) is satisfied. The communities are differentiated in a strong sense and in a weak sense in [26]. It worths to notice that many clustering algorithms can be used to discover communities under different conditions if we regard community discovery as a kind of

clustering process [32] [34].

In real-world networks, nodes and links contain some information, and may have certain semantics. If nodes are assigned with semantics, heuristic methods can be used to reduce the cost of operating the whole graph. The similarity or dissimilarity between nodes can be measured if nodes are represented by a set of features. Based on the dissimilarity between nodes, the minimum spanning trees generated from the original network can be used to efficiently regionalize socio-economic geographical units [4]. A physics-based approach is proposed to find communities efficiently by using the notions of voltage drops across network [36].

Semantic relation discovery is an important issue for network application. Aleman-Meza et al propose an approach to discovering various semantic associations between reviewers and authors in a populated ontology to determine a conflict degree of interest [3]. This ontology was created by integrating entities and relationships from two social networks: friend-of-a-friend and co-author social networks. Matsuo et al propose a social network extraction system POLYPHONET employing several techniques to extract relations of persons, detect groups of persons, and obtain keywords for person [21]. Approaches to learning social network from incomplete relationship data is proposed [20]. It assumes that only a small subset of relations between individuals is known, therefore, the social network extraction is translated into a text classification problem. Cai et al discuss the approach for mining hidden communities in heterogeneous social networks [9].

The topological structures of three real online social networking services, each with more than ten million users, are compared and analyzed in [2]. Kumar et al present a series of measurements of two such networks with over five million people and ten million friendship links, annotated with metadata capturing the time of every event in the life of the network [18].

Pujol et al discuss the issue of calculating the degree of reputation for agents acting as assistants to the members of an electronic community and give a solution. Usual reputation mechanisms rely on the feedback after interaction between agents [31]. An alternative way to establish reputation is related with the position of each member of a community within the corresponding social network. The group formation issue in large social network is also discussed in [5].

The growth of social networking on the Web and the properties of those networks have created a great potential for producing intelligent software that integrates users' social network and preferences. Golbeck and Hendler assign trust in Web-based social networks and investigate how trust information can be mined and integrated into applications [13]. Hossain et al draw on network centrality concepts and coordination theory to understand how project's team members interact when working towards a common goal [15]. A text-mining application based on the constructs of coordination theory was developed to measure the coordinative activity of each employee. Results show that high network centrality is correlated to the ability of an actor to

the ability of an actor to coordinate actions of others in a project team. Yang et al studied the approach to community mining in a signed social network [37]. An approach to discovering global network communities based on local centralities is proposed [38].

2.3. On Semantic Web and Semantic Link Network

Tim Berners-Lee et al proposed the notion of Semantic Web [6], which has become a research area. The URI (www.w3.org/Addressing/URL/URI_Overview.html) is used to uniquely identify resources. XML (www.w3.org/XML/), RDF (www.w3.org/RDF/) and OWL (www.w3.org/TR/owl-features/) describe the semantics of resources at different levels. SPARQL (www.w3.org/TR/rdf-sparql-query/) serves as the query language of the Semantic Web. The Semantic Link Network SLN was proposed as a semantic data model for organizing various Web resources by extending the Web's hyperlink to semantic link.

SLN is a directed network consisting of semantic nodes and semantic links. A semantic node can be a concept, an instance of concept, a schema of data set, a URL, any form of resources or even a SLN [40]. A semantic link reflects a kind of relational knowledge represented as a pointer with a tag describing such semantic relations as *causeEffect*, *implication*, *subtype*, *similar*, *instance*, *sequence*, *reference* and *equal*. The semantics of tags are usually commonsense and can be regulated by its category, relevant reasoning rules, and instances of usage. A set of general semantic relation reasoning rules was suggested in [40] and [42]. If a semantic link exists between nodes, a link of reverse relation may exist, e.g., $A \text{---}isSouthOf\text{---}B$ is the reverse link of $B \text{---}isNorthOf\text{---}A$, where *isSouthOf* and *isNorthOf* are commonsense. A relation could have reverse relation. Relations and their corresponding reverse relations are knowledge for supporting semantic relation reasoning. SLN is a self-organized network since any node can link to any other node used as semantic link. SLN has been used as a major mechanism of organizing resources for the Knowledge Grid [40]. Pons has successfully applied the SLN to object prefetching and achieved a better result than other approaches [30].

In the following, we extend previous SLN model to a self-organized semantic data model. For simplicity, SLN denotes both the Semantic Link Network as a model and the network of semantic links in this paper.

3 THE SEMANTIC LINK NETWORK — A SELF-ORGANIZED SEMANTIC DATA MODEL

3.1 The Basic Semantic Link Network Model

Various explicit and implicit semantic relations in the world constitute various semantic link networks, which can be formalized into a loosely coupled semantic data model for managing various Web resources. It consists of semantic nodes, semantic links between nodes, and a set of relational reasoning rules like $\alpha\beta\Rightarrow\gamma$ (i.e., the

connection of semantic relation α and semantic relation β implies semantic relation γ).

As a data model, SLN consists of the following parts as shown in Fig. 1:

1. **Primitive Semantic Space.** It specifies the semantics of semantic nodes and semantic links. It consists of the classification trees on concepts, which can also represent relations, reasoning rules and basic data types. In a classification tree, the root concept is classified by its subconcepts, which can be further classified by finer subconcepts. The semantic distance between two concepts in a classification tree is the sum of their distances to the nearest common ancestor. Usually, the first-level of the classification trees regulates commonsense, and the second-level regulates domain commonsense like ACM CCS. Users can use their own keywords to tag semantic nodes and semantic links by extending the classification. The frequently used user-defined tags can be regarded as commonsense by linking them to existing classes (concepts), but other user-defined tags should be given detailed explanation. A semantic node in SLN can be represented as *name: field*, or a schema of data sets [*name: field*, ..., *name: field*], where *name* and *field* respectively represents attribute and its data type in this case. A *field* can also refer to the path from a root to the class in the classification tree. The *field* can be default if it is a commonsense. A semantic link is denoted as *name: (SemanticNode, SemanticNode)*. The primitive semantic space is shared by all participants and evolves with their use of the space in managing the
2. **Metric Space.** It evaluates the semantic nodes and semantic links. The value of a semantic link is in positive proportion to the following three factors: (a) the values of its two ending nodes; (b) the times of its occurrence in SLN; and, (c) the times it participates in reasoning. The value of a semantic node is in positive proportion to the values of its neighbor nodes. The metric space also determines

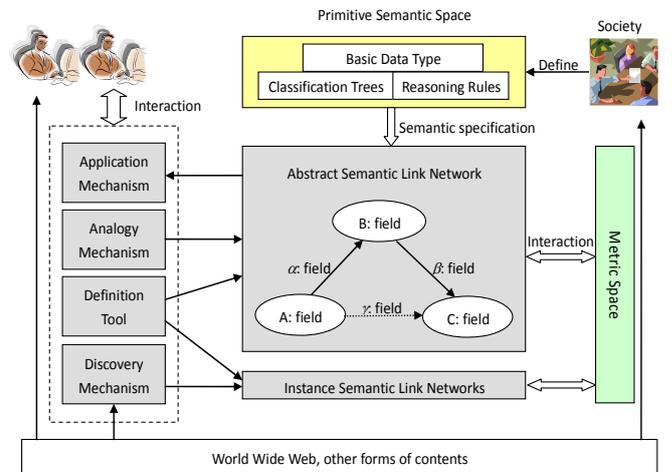


Fig.1. The SLN model and relevant techniques. The dashed block represents relevant techniques for creating and using the SLN.

the probability over the SLN. The probability of the existence of a semantic link is in positive proportion to the probability of the existence of its precondition relations of a reasoning rule. The probability of the existence of a node is determined by the probability it belongs to a classification in the semantic space.

3. **Abstract SLN.** It consists of abstract semantic nodes and abstract semantic links, which connect semantic nodes by abstract relations. An abstract SLN can be regarded as the schema of SLN where semantic nodes and semantic links are defined in semantic space.
4. **Instance SLNs.** An instance SLN consists of semantic nodes and links instantiated from the abstract SLN. An abstract SLN can generate several SLN instances by instantiating its semantic nodes and semantic links.

SLN Schema is a triple denoted as $SLN\text{-}Schema = \langle ResourceTypes, LinkTypes, Rules \rangle$. $ResourceTypes$ is a set of resource types, each of which is represented as $ResourceType = [name: field] \mid [name: field, \dots, name: field]$. $LinkTypes$ is a set of semantic link types belonging to $ResourceTypes \times ResourceTypes$, each of which is represented as $LinkType = [name: (ResourceType, ResourceType)]$. $Rules$ is a set of reasoning rules on $LinkTypes$, denoted as $Rules = \{ \alpha \beta \Rightarrow \gamma \mid \alpha, \beta, \gamma \in LinkTypes \}$. The field can be defined by the basic datatype, classification trees or rules in the primitive semantic space.

The following are two strategies to construct a SLN as a data model.

Schema-First Strategy. Define SLN schema first and then instantiate it according to application requirements.

This strategy requires users to share the same schema information: resource type, link type and reasoning rules. This also implies that users have consensus on the primitive semantic space.

SLN schema is useful in defining SLN for special interested groups or local applications. But it is not appropriate to define a rigid schema for massive Web applications, especially where resources are self-organized and expanding, and relations keep changing.

Self-Organized Strategy. Users freely define instance SLNs and their rules, and then link them to each other. A linked semantic data models can be obtained by analyzing existing SLNs, discovering semantic communities, making abstraction on semantic nodes and semantic links, and regulate the semantic structure of semantic nodes.

The self-organized strategy can adapt to the change of resources and relations. To raise efficiency, queries are more often routed within the same semantic community than across communities [43].

3.2 Operations of SLN

The following are some important SLN operations:

1. Locate all semantic nodes linked to the given node via a set of given semantic links. If the semantic links are default, it outputs all nodes

semantic links are default, it outputs all nodes linked to the given node.

2. Locate the semantic community a given node belongs to.
3. Locate the semantic communities a set of given nodes belongs to.
4. Locate the semantic community a given semantic link belongs to.
5. Locate a semantic path connected by a given pair of semantic nodes.
6. Locate a semantic community that semantically includes a given semantic community or semantic node.
7. Delete a semantic link. If it inputs a semantic link, then it deletes all of the semantic links appeared in the SLN. If it inputs a semantic link and two semantic nodes, it deletes the semantic link between the two semantic nodes.
8. Add a semantic link to SLN. It inputs one semantic link and two semantic nodes, and then adds the semantic link between the two nodes.
9. Delete an isolated semantic node in SLN.
10. Add a semantic node to SLN. It inputs a semantic node, a semantic link and a target semantic node in SLN, and then connects the semantic node to the target semantic node by the semantic link.

The operations between SLNs ‘ \cup ’, ‘ \cap ’ and ‘ $-$ ’ are graph-based operations listed as follows:

1. **Union.** It inputs SLN_1 and SLN_2 , and then outputs $SLN = SLN_1 \cup SLN_2$, the union of two SLNs as graph and the union of the rule sets of SLN_1 and SLN_2 .
2. **Intersection.** It inputs SLN_1 and SLN_2 , and then outputs $SLN = SLN_1 \cap SLN_2$ by calculating the intersection of two SLNs as graph and the intersection of two rule sets.
3. **Difference.** It inputs SLN_1 and SLN_2 , and then outputs $SLN = SLN_1 - SLN_2$ by removing their common semantic links from SLN_1 and keeping its own rules.
4. **Matching.** It inputs SLN_1 and SLN_2 , and then outputs a matching degree, which can be defined as a function of above operations.

SLN also provides the following operation to find influence set for high-level applications:

InfluenceSet(α). Given a relation α to find a subset of Rules that satisfies:

1. if α appears in the precondition of a rule in Rules, then the rule is also in InfluenceSet(α);
2. if α is the post-condition of a rule in Rules, then the rule is also in InfluenceSet(α); and,
3. if there exists a set of rules $rule_1, \dots, \text{and } rule_n$ in Rules such that they can link with each other for reasoning and α appears in the precondition of $rule_1$ or in the post-condition of $rule_n$, then InfluenceSet(α) includes $rule_1, \dots, \text{and } rule_n$.

The influence set of a relation is actually all of the relations that are influenced by or influence the given semantic relation.

More operations can be defined for various purposes, but what is the most basic set of operations? The following lemma answers this question.

Lemma 1. *The basic operation set of SLN consists of the following four operations: add an isolated node n to SLN $AddNode(n)$; delete an isolated node n from SLN $DelNode(n)$; add a semantic link α between nodes n and n' in SLN $AddLink(\alpha, n, n')$; and, delete a semantic link α between two nodes n and n' in SLN $DelLink(\alpha, n, n')$.*

Firstly, it is clear that operations in the basic set of operations cannot be expressed with each other. Secondly, for any two semantic link networks SLN_1 and SLN_2 , there exists a series of basic operations to transform one into another. SLN_1 and SLN_2 can become the same by continuously applying the basic operations in the following cases: use $DelNode$ to delete node n in SLN_1 if there exists an isolated semantic node n in SLN_1 but not in SLN_2 , use $AddNode$ to add an isolated node n to SLN_1 if there exists an isolated semantic node n that is not in SLN_1 but in SLN_2 , use $DelLink$ to remove α from SLN_1 if there exists a semantic link α in SLN_1 but not in SLN_2 , and use $AddLink$ to add α to SLN_1 if there exists a semantic link α in SLN_2 but not in SLN_1 . Thirdly, operations between SLNs can be implemented by basic operations since a SLN can be regarded as a semantic node. So, above lemma holds.

3.3 Relational Reasoning of SLN

Two connected semantic links could derive out a new semantic link if there is an applicable reasoning rule. The reasoning rules can be regarded as an operation “ \Rightarrow ” on semantic relations, e.g., rule $n-\alpha \rightarrow n', n'-\beta \rightarrow n'' \Rightarrow n-\gamma \rightarrow n''$ can be represented as a calculus on semantic relations: $\alpha \cdot \beta \Rightarrow \gamma$ (we say that α, β and γ participate in a reasoning). Each rule can be assigned a certainty degree defined in the metric space to represent the recommender’s confidence on this rule. Table 1 gives some heuristic reasoning rules for reference.

A SLN can be represented by a Semantic Relationship Matrix SRM, where element l_{ij} represents a set of semantic relations from resource r_i to r_j , $l_{ii}=eq$, and l_{ji} is the reverse relation of l_{ij} . If there are no semantic relations between r_i and r_j , $l_{ij}=l_{ji}=null$. The SRM of any SLN is unique if the order of nodes in the matrix is fixed.

Definition 1. *Reasoning closure of SLN denoted as SLN^+ is a reasoning complete SLN, no new semantic link can be derived out by applying the reasoning rules.*

The reasoning closure SLN^+ can be computed by the multiplication of the Semantic Relation Matrix SRM [40]. For a given semantic relation matrix $M=(M_{ij})^{n \times n}$, the result of $M_{i,k} * M_{k,j}$ is still a set. $M_{i,k} * M_{k,j}$ means that the i th node can reach the j th node with the semantic type in set $M_{i,k} * M_{k,j}$ by two steps. According to the definition of $M_{i,k} * M_{k,j}$, we can define $M^{k+1} = M^k \times M$, and $M_{ij}^{(k+1)}$ means that the i th node can reach the j th node with the semantic types in $M_{ij}^{(k+1)}$ by $k+1$ steps. Let X be the SRM of the

TABLE 1
SOME HEURISTIC REASONING RULES.

Rules	certainty	Explanation
$x \cdot eq \Rightarrow x$	certain	x : any relation
$eq \cdot x \Rightarrow x$		eq : equivalence relation
$ce \cdot ce \Rightarrow ce$	highest	ce : cause-effect relation or conditional probability
$seq \cdot seq \Rightarrow seq$	highest	seq : sequence relation
$imp \cdot ce \Rightarrow ce$	highest	imp : logical implication relation
$ce \cdot imp \Rightarrow ce$	high	imp is stronger than ce
$st \cdot ins \Rightarrow ins$	high	st : subtype relation ins : instance relation
$st \cdot ce \Rightarrow ce$	high	st : type relation keeps features
$st \cdot st \Rightarrow st$	high	st : type relation keeps features
$imp \cdot imp \Rightarrow imp$	high	transitivity of logical implication
$imp \cdot st \Rightarrow imp$	high	st : type relation keeps features
$imp \cdot ins \Rightarrow ins$	high	tightly coupled instance relation
$ce \cdot sim \Rightarrow ce$	medium	sim : similar relation
$ins \cdot ref \Rightarrow ref$	medium	ref : reference relation
$ce \cdot st \Rightarrow ce$	medium	st : type relation keeps features

SLN^+ , then $X=(M_1 + M_2 + \dots + M_k)$, where k is a positive integer determined by the recursion procedure. The maximum number of k is the length of the acyclic longest reasoning path.

Characteristic 1. *A SLN is equivalent to its reasoning closure in semantics.*

For the same SLN, given different reasoning rule sets will generate different reasoning closures.

Characteristic 2. *Two SLNs are equivalent if their reasoning closures are the same.*

Reasoning on SLN only depends on its reasoning rules, multiple semantic links or paths may exist between nodes representing relations of different aspects, so the following characteristic holds.

Characteristic 3. *$(\alpha \cdot \beta) \cdot \gamma$ may not equal to $\alpha \cdot (\beta \cdot \gamma)$, for any three sequentially connected relations α, β and γ .*

Characteristic 4. *The connection of two semantic links (relations) may derive out a new semantic link different from the existing semantic link between the same pair of nodes, that is, the generation of a new semantic link depends on the Rules of SLN.*

Above characteristic implies that a network of semantic links cannot fully determine its semantics. Given different Rules, a network of semantic links can stand for different semantics.

The SLN inherits the self-organization characteristics of the Web’s hyperlink network. It is easy to use, without any special training. It is a loosely coupled semantic data model that allows new nodes to be freely added to the network by establishing the semantic link(s) with existing semantic nodes. Once established, it supports various relational queries with rich semantic links and semantic relation reasoning. SLN can be implemented on the Web by RDF and Semantic Web Rule Language SWRL.

3.4 Some Roles of Semantic Link in e-Learning

Isolated knowledge is useless as it is neither easily retrieved nor easily expanded. Learning concerns internalizing and externalizing knowledge and establishing relations between knowledge of various types and from different sources. The following are some roles of semantic links in e-learning:

1. A knowledge unit can be expanded or localized through the *isPartOf* link.
2. A knowledge unit can be reused to solve similar problems or answer similar questions through the *similar* link.
3. A knowledge unit can be abstracted or specialized through the *subtype* link.
4. A knowledge unit can be accessed from other knowledge units through a semantic path, a sequential connection of semantic links.

Semantic links between resources can be established in two ways: user definition and automatic discovery. User definition relies on software tool with the interface for specifying semantic nodes and semantic links between two sets of resources or connecting a new resource to an existing resource by semantic link. A Web-based tool was developed to support users to easily define and browse SLN [41].

The diversity of semantic nodes in SLN enables an e-learning system to provide rich media for users. Based on Web 2.0, an e-learning system can realize the ideal of *one for all and all for one* during externalizing and internalizing knowledge.

The process of automatically generating semantic links can explain how this relation is established. The following three ways can be cooperatively used to automatically establish semantic links:

1. Discovering semantic links in a given set of resources by analyzing the contents of resources and their meta-data, and determining their relations according to the semantic links between contents (e.g., similar relation and co-occurrence relation), relations between meta-data, and relations between link structures.
2. Deriving new semantic links by relational reasoning and analogical reasoning on existing semantic links according to reasoning rules [42].
3. Inferring a semantic link according to the frequency of its appearance in SLN.

Adding one semantic link to existing SLN may generate more semantic links. With increment of semantic links between resources, an e-learning system becomes more capable of providing relational knowledge for users. The relational reasoning mechanism further helps users know the intrinsic relationship between resources in a large-scale network of resources. The learning process is involved in the definition and browsing of semantic links relevant to the learned knowledge.

4 SEMANTIC COMMUNITIES

In graph-based community, nodes are connected in tightly knit groups, between which there are only looser connections. The connection degree depends on the quantity of edges within community. GN algorithm works well in discovering community structure in the food web of the predator-prey interactions between species [13]. But previous food webs do not reflect such semantic relations as two species live in the same region and belong to the same category.

In SLNs, semantic links reflect the semantics of connections, and potential semantic links could be derived out. An example of SLN is the knowledge base of production rules, whose transitivity enables new rules to be derived from existing rules. Fig. 2 compares an ordinary graph (a) and the corresponding SLN (b).

Before studying community discovery on SLN, we need to define the notion of semantic community. The semantics of a SLN is defined in the primitive semantic space. If the SLN is defined strictly according to the characteristics of its semantic space, the semantic communities can form trees. If semantics of nodes is not available, the semantic links and reasoning rules largely represent the semantics of an SLN.

A distinguished characteristic of the SLN is its reasoning ability. When the semantics of semantic nodes is not available, if a semantic link cannot participate in reasoning with its neighbor links, it can be regarded as isolated from the network, and therefore, it makes less contribution to the semantics of a community. So different from the graph-based community notion, a semantic community consists of two parts: *structure* and *semantics*. A semantic community can be defined from structure and reasoning point of view as follows.

Definition 2. A reasoning-constraint semantic community is a SLN satisfying the following three conditions:

1. It is a connected graph.
2. It does not include such a semantic link that does not participate any reasoning.
3. The intra-community semantic links can participate in reasoning with each other much more times than with the inter-community semantic links.

The second condition of above definition implies that a semantic link should belong to another community if it cannot participate in reasoning with its neighbor semantic



Fig. 2. (a) A graph consisting of two communities. (b) Adding semantics to the edges could lead to disappearing of the two communities since new link d could be derived out if semantic links a and b satisfy rule $a-b \Rightarrow d$. A new semantic link e could be further derived from d and the existing semantic link c .

links. For example, a *coAuthor* relation should not be added to a community of family relations as it could not participate in reasoning with such relations as *fatherOf*, *motherOf* and *brotherOf*. This condition is used to discover semantic communities, where every semantic link participates reasoning at least once. The third condition ensures that semantic links within community should be tight and should be loose between communities. This definition allows two semantic communities share a semantic node. This also means that given different sets of semantic links over the same set of nodes represents different semantic communities.

One semantic link could derive out new semantic links with its adjacent semantic links, and the new semantic links could further derive out new semantic links with its adjacent semantic link according to the reasoning rules. Different semantic links play different roles in SLN. The total number of semantic links that can participate in reasoning with a semantic link reflects its importance in the network or the extent of other semantic links relying on it. The following definition reflects such an importance or reliance between semantic links, which will play role in discovering semantic communities in SLN.

Definition 3. *The semantic betweenness of a semantic link in a given SLN is the number of times it participates in reasoning according to reasoning rules.*

Some reasoning rules are closely related while others are loosely related. A set of closely related rules influences the formation of a semantic community.

Definition 4. *A rule-constraint semantic community is a semantic link network, where reasoning only carries out within community.*

The classification trees on concepts are another factor of discovering semantic communities in a SLN. The basic assumption is that the concepts in the same classification tree should be closely related with each other, and therefore they should be in the same semantic community.

Definition 5. *A SLN is called a classification-constraint SLN if all of its semantic nodes and semantic links (relations) appear in the same classification tree.*

Definition 6. *A classification-constraint semantic community is a SLN that satisfies:*

1. *Semantic nodes and relations belong to a common class (community root) in the classification trees; and,*
2. *The semantic distance between any pair of intra-community nodes \leq the semantic distance between any pair of intercommunity nodes.*

Above definition provides a way to find a semantic community hierarchy of a SLN.

5 DISCOVERING REASONING-CONSTRAINT SEMANTIC COMMUNITIES

5.1 Decomposition Approach

Here we introduce an algorithm named SLN-DeCom to discover semantic communities in SLN. It removes the

semantic links with the lowest semantic betweenness with reference to its closure SLN⁺.

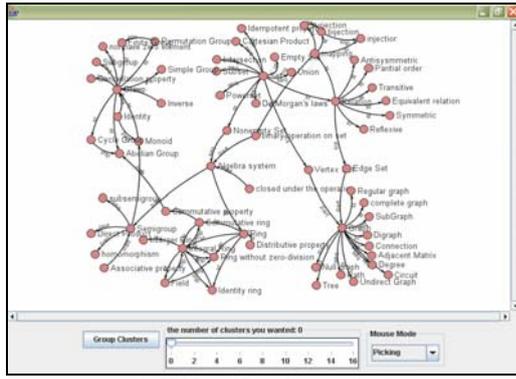
Algorithm SLN-DeCom (input: SLN; output: a community tree)

1. Construct the SLN⁺ of the input SLN, record the semantic betweenness of all semantic links and list them in descending order, and record all of the semantic links that have reasoned with any other semantic link.
2. Remove all the semantic links with zero semantic betweenness from the SLN (as removal of these semantic links will not affect the semantic betweenness of other semantic links).
3. Remove the semantic link(s) with the smallest semantic betweenness from the SLN if this does not generate isolated nodes.
4. Check the reasoning rule set Rules and find all of the semantic links that have reasoned with the semantic link removed by step 3. Decrease the semantic betweenness of the semantic links that have reasoned with the removed semantic link(s).
5. Repeat from step 2 until no semantic links is qualified to be removed or isolated node is found.

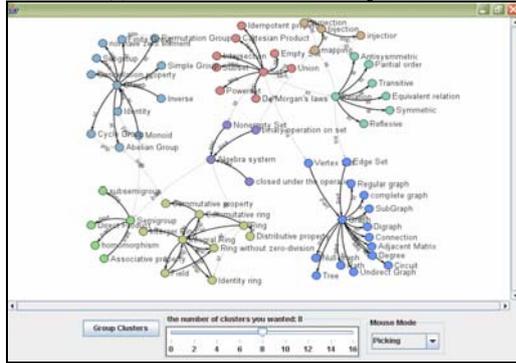
The algorithm only needs to calculate the SLN⁺ once. It can also avoid recalculating the semantic betweenness for all semantic links after removal of one semantic link by checking the influenced semantic links at step 4 according to rules.

A tree of communities could be formed during the semantic community discovery process of using the decomposition approach. The tree can help e-learning system search and understand a SLN. For example, it can reduce search space by determining which branch the target resides and explain the semantic community by top-down or bottom-up ways.

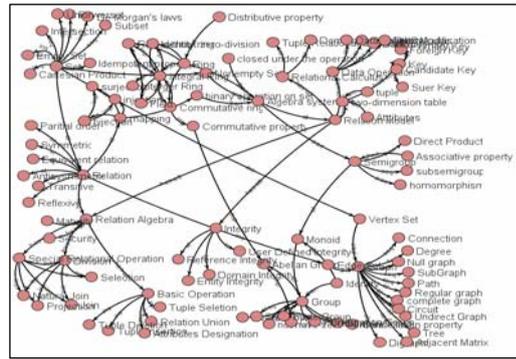
As an example of applying this algorithm, we carried out the following experiment. From the textbook of discrete mathematics, we select 65 concepts and theorems and link them according to their algebra relations as shown in Fig.3 (a). The communities discovered by using the SLN-DeCom algorithm are shown in Fig.3 (b). The obtained semantic communities match the real classifications in algebra. Fig.3 (c) shows a larger SLN with more concepts in the relation model. We then apply the SLN-DeCom to discover its semantic communities. Fig.3 (d) shows the following discovered meaningful communities: graph community, set and relation community, group and ring community and relation model community. We cannot obtain group community and ring community respectively until the whole SLN is divided into ten communities. The reason is that group and ring are separated from the group and ring community while the connection between them are the weakest in the SLN, whereas the connections between the nodes in the relational model are weaker than the connection between group and ring, which implies that the group community and ring community occur until the relational model community is divided into several communities. Such knowledge can be used in e-learning



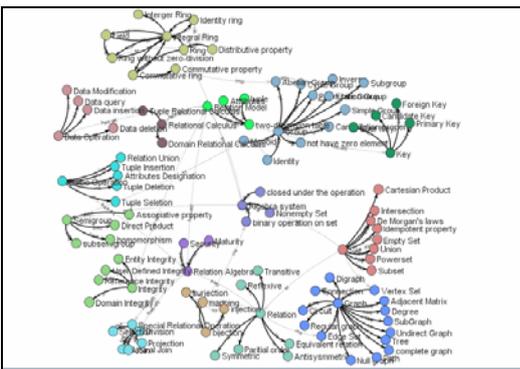
(a) A SLN of 65 nodes on algebra.



(b) Semantic communities in (a).



(c) A SLN of 102 semantic nodes.



(d) Semantic communities in (c).

Fig.3. Examples of discovering semantic communities in SLNs.

knowledge can be used in e-learning system to help push reasonable content to users.

The algorithm has the following characteristics.

Characteristic 5. Let N be the number of nodes in the network, and n be the minimum number of nodes in a community.

We can observe the following phenomena:

1. The maximum number of communities is $\lfloor N/n \rfloor$.
2. The discovered communities form a tree, where descendents are the sub-graphs of their common ancestor. This tree structure helps localize the operation
3. To delete a semantic link earlier or later does not influence the final result.

Two strategies can be adopted for community discovery: breadth-first or depth-first, but the same result will be reached because they are independent in the community discovery process once a community is divided into two.

5.2 Construction Approach

Many networks have such features: some nodes play more important role than others in forming communities. An idea of community discovery is to find some initial communities, adjusting and combining communities to discover more reasonable semantic communities according to the semantic relationships in the SLN⁺.

The following algorithm SLN-ConCom inputs the community intensity η to help decide whether two communities should be combined to one community. If more than η percent of nodes in one community are linked to the nodes in another community or vice versa, the two communities should be combined to one community. Experiments indicate that the appropriate value of η is 33%. The construction algorithm is described as follows:

Algorithm SLN-ConCom (Input: community intensity η ; Output: communities)

1. Calculate the degrees (total number of in-links and out-links) of all nodes in the given SLN and rank them in descending order to form a degree queue (arbitrarily arrange the order of the nodes with the same degree).
2. Construct the semantic closure SLN⁺.
3. The node with the highest degree and its neighbors constitute an initial community C_0 . Remove these nodes from the degree queue. Let $t=0$.
4. $t=t+1$; Let the first node k in the queue be the central node of a new community C_t . Remove k from the queue.
5. For every neighbor of node k , put the neighbor into one community in $\{C_0, \dots, C_t\}$ which has the largest number of nodes semantically linked to the neighbor in SLN⁺.
6. Check every community C_j ($j=0, \dots, t$). If more than η percent of the neighbors of node k belong to the community C_j or more than η percents of the nodes in the community C_j semantically link to node k in SLN⁺, then merge C_t to C_j and $t=t-1$ (because the number of communities does not increase).
7. Repeat from step 4 until the number of communities satisfies user requirement or all nodes have been assigned to the communities.

As an example, applying this algorithm to the SLNs shown in Fig.3 (a) and (c) obtains the same result as using SLN-DeCom. The above algorithm can be enhanced by making use of not only the important nodes but also the

important semantic links. This type of algorithms is suitable for those SLNs where nodes or links are important in semantics.

6. DISCOVERING RULE-CONSTRAINT AND CLASSIFICATION-CONSTRAINT SEMANTIC COMMUNITIES

Current graph-based community discovery approaches are not suitable for large networks where almost no semantic information is available. Different from ordinary graph, the semantic basis of SLN is its primitive semantic space, so semantic communities in SLN can be discovered by making use of the clustering features of the reasoning rules and the classification trees.

6.1 Discovering Rule-Constraint Semantic Communities by Rule Cluster

Rules can not only conduct reasoning on semantic links but also can reason with each other to form rule chains or trees. One rule can reason with the other if they share some relations.

Definition 7. A SLN is called rule-constraint SLN if all of its semantic relations appear in its Rules.

Definition 8. A rule cluster of the reasoning rules is a set of rules, which can only reason with the rules within the rule cluster. The minimum rule cluster on rules is such a rule cluster that cannot be further partitioned into smaller rule clusters.

Above definition implies the following characteristics.

Characteristic 6. (1) The minimum rule cluster is a connected graph of rules that can reason with each other. (2) If there exists a rule that can reason with the rules in two rule clusters, the two rule clusters are the same.

The SLN can be very large, but its rule set is usually much smaller. The rule cluster can help efficiently determine relevant semantic communities.

Definition 9. The maximum partition of Rules is such a reasoning rule set $\{Rules_1, Rules_2, \dots, Rules_n\}$ that each of which is a minimum rule cluster.

Characteristic 7. Let $Pre(X)$ and $Post(X)$ be the precondition set and post-condition set of the rule set of X respectively. For two rule clusters $Rules_i$ and $Rules_j$, the following items hold:

1. $Rules_i \cap Rules_j = \{\}$;
2. $Pre(Rules_i) \cap Post(Rules_j) = \{\}$;
3. $Post(Rules_i) \cap Pre(Rules_j) = \{\}$;
4. $Pre(Rule_i) \cap Pre(Rules_j) = \{\}$; and,
5. $Post(Rules_i) \cap Post(Rules_j) = \{\}$.

The following algorithm is for discovering rule clusters in the given rule set.

Algorithm DisRuleCluster (Input: Rules; Output: rule clusters)

1. Set one empty rule cluster.
2. Remove one rule from Rules and compare it with

the rules in existing rule clusters.

3. If characteristic 7 is satisfied, put the rule into a separate cluster.
4. Else merge it with the rule cluster(s) that leads to unsatisfaction of characteristic 7 as one cluster.
5. Repeat from step 2 until Rules becomes empty.

Characteristic 8. For a given rule-constraint SLN and a rule set Rules, if Rules can be clustered into a set of rule clusters $\{Rules_1, Rules_2, \dots, Rules_n\}$, then SLN can also be partitioned into a set of semantic communities $\{SLN_1, SLN_2, \dots, SLN_n\}$ and semantic reasoning of SLN_k depends only on $Rule_k$ ($k=1, 2, \dots, n$).

According to characteristic 6, 7 and 8, we give the following algorithm for discovering the semantic communities in a rule-constraint SLN.

Algorithm DisSemCom (Input: Rule-constraint SLN and rule clusters $\{Rules_1, Rules_2, \dots, Rules_n\}$; Output: $\{SLN_1, SLN_2, \dots, SLN_n\}$)

1. $k=1$;
2. Find all semantic links appeared in the $Pre(Rules_k)$, filter these semantic links from SLN, add them and relevant nodes to SLN_k , and then delete these semantic links in SLN;
3. $k=k+1$, repeat step 2 until all edges in SLN are deleted.

Discovering semantic communities in SLN enables operations on SLN to be localized within a local SLN specific to a rule cluster. It also enables operations on the SLN to keep focusing on a specific semantic community while semantic communities keep changing.

Theorem 1. Let $\{Rules_1, Rules_2, \dots, Rules_n\}$ be the maximum partition on the rule clusters of Rules. The rule-constraint SLN is partitioned into n semantic communities $\{SLN_1, \dots, SLN_n\}$ according to the clusters. Finding SLN closure SLN^+ is equivalent to finding each SLN_k^+ ($k=1, \dots, n$), i.e., $SLN^+ = SLN_1^+ \cup \dots \cup SLN_n^+$.

Proof. Since DisRuleCluster ensures $Rules = Rules_1 \cup Rules_2 \cup \dots \cup Rules_n$, we only need to check if $SLN^+ = SLN_1^+ \cup \dots \cup SLN_n^+$.

For any $e \in SLN^+$, there are two cases: $e \in SLN$ or $e \notin SLN$.

1. If $e \in SLN$, then there exists a $Rules_k$ such that $e \in Rules_k$, as SLN is rule-constraint. According to algorithm DisSemCom, we have $e \in SLN_k$, therefore $e \in SLN_k^+$.
2. If $e \notin SLN$, then there exists a rule $\alpha\beta \Rightarrow e$ in Rules, where α and β are relations in SLN or can be derived from relations in SLN by rules in Rules. According to DisRuleCluster, there exists a $Rules_k$ that includes these rules. According to DisSemCom, there exists SLN_k where all relations are in $Rules_k$. Since SLN_k^+ is the complete reasoning of SLN by using $Rules_k$, we have $e \in SLN_k^+$.
3. Since $SLN_1^+ \cup \dots \cup SLN_n^+ = SLN^+$, then there exists a $k \in [1, n]$, $e \in SLN_k^+$. Since $SLN_k^+ \subseteq SLN^+$, we have $e \in SLN^+$.

Summarizing above three points, we have $SLN^+ = SLN_1^+ \cup \dots \cup SLN_n^+$. \square

This characteristic suggests the following strategy of raising the efficiency of calculating the SLN closure SLN^+ :

Strategy for Calculating Semantic Closure. *Discover semantic communities in SLN first, and then calculate the SLN^+ of each semantic community.*

According to Theorem 2, we have the following lemmas.

Lemma 2. *The operation result of deleting a semantic link e in SLN is the same as deleting e in its semantic community.*

Lemma 3. *Adding a semantic link e to SLN is the same as adding e to its semantic community (If relation e does not exist in Rules, e itself forms a community).*

Characteristic 9. *If all relations in the new rule belong to a rule cluster, adding this new rule will not influence other rule clusters.*

Strategies for Adding a Rule to Rules:

1. If all relations in the new rule do not appear in existing rule cluster, the new rule forms a new rule cluster independent of existing rule cluster.
2. If all semantic relations in the precondition of the new rule belong to a rule cluster, then this new rule belongs to this rule cluster.
3. If all semantic relations in the precondition of the new rule belong to a rule cluster, and all semantic relations in the post-condition of the new rule belong to another rule cluster, then this new rule and the two rule clusters can be merged into one rule cluster.
4. If semantic relations in the precondition or postcondition of the new rule belong to different rule clusters, then merge these clusters and the new rule into one cluster.

The ability of discovering semantic communities by rule cluster enables SLN to be a self-organized semantic data model for organizing resources. Although a large SLN may be defined by different people and at different times, all operations on SLN can be localized within appropriate semantic communities regulated by rule clusters.

6.2 Discovering Classification-Constraint Semantic Communities by Classification Tree

Using classification semantics to discover communities in the SLN is the most direct and efficient way if the semantic information of semantic nodes and the semantic information of the classes (concepts) in the classification tree are available. In Web 2.0 applications, the concepts in classification tree, the semantic links, and the semantic nodes can be characterized by a set of tags. Matching between them can be implemented by matching their tag sets. Each tag has its frequency of usage in annotating resources.

Definition 10. *Semantic distance between two communities is the semantic distance between two concepts in the classification tree, each of which is the nearest common*

tion tree, each of which is the nearest common concept of semantic nodes of a community.

Given a semantic distance function, the following approach can construct a semantic community hierarchy for SLN bottom-up.

1. Take every semantic node of the given SLN as a semantic community.
2. Merge the semantic communities with the shortest semantic distance between them into one community.
3. Repeat from step 2 until there are only two communities left.
4. Add semantic links between semantic nodes within each community according to the given SLN.

The top-down approach consists of the following steps:

1. Take SLN as an initial community.
2. For every semantic link, calculate the semantic distance between its two ends' semantic nodes;
3. Delete the semantic link with the longest semantic distance ending nodes until the SLN is separated into two communities;
4. For each current community do from step 3 until all communities only include an isolated node.

The basic premise of above two approaches is that two semantic nodes far away from each other in semantic distance should belong to different semantic communities. Different from the GN algorithm, deleting one link does not influence the semantic distance of other links. Algorithms for discovering semantic communities in SLN can use more semantic information in nodes and links than ordinary graph, so they can perform well and serve for more advanced applications.

In real applications, especially in large-scale and self-organized applications, a SLN may not be completely rule-constraint or classification-constraint. In this case, we can find the constraint SLN first, set the unconstraint SLN apart as an unconstraint community, and then reassign the unconstraint semantic nodes that link to the constrained nodes by tightly bounded semantic links such as *subtype* and *similar* links to the constraint communities according to application requirement. The percentage of the constraint SLN to the SLN reflects the constraint degree of the primitive semantic space.

7 DISCOVERING EMERGING SEMANTICS IN SLN

The semantics of and between semantic nodes emerges with the creation and evolution of an SLN. An intelligent e-learning system should be able to help users discover the emerging semantics of and between semantic nodes in a complex SLN.

7.1 The Massive Emerge Principle

The importance of a Web page is influenced by the ranks of its neighbors [29]. The rising of a page's rank will influence the whole network through its neighbors. The rank of a newly joined page would be low as it takes time to become known to others. The bias for a new node to

get rise in rank is to link itself to high-rank nodes. The page ranks keep stable if no new pages and links are added. Page ranks in the hyperlink network follows the power-law distribution [1].

Different from the page rank, linking a new node or adding a new semantic link to a SLN could generate new semantic links by relation reasoning. A new node will be immediately known by relevant nodes within a community by relational reasoning. Fig 4 shows an example of adding a new node G to the SLN. Linking new node G to an existing node E via the *brotherOf* relation generates six semantic links denoted as dotted arrows.

Characteristic 10. *The richness of a semantic node is in positive proportion to the number and diversity (diverse types) of the semantic links it has and the richness of its neighbors.*

A richer semantic node can provide richer content and more semantic relations for others.

Characteristic 11. *The richness of a semantic link is in positive proportion to the following factors:*

1. *The number and richness of the semantic links it can reason with, the more the richer.*
2. *The times of the relation appeared in SLN, the more the richer.*
3. *The richness of its two ending nodes, the richer the richer.*

The richness of a SLN is in positive proportion to the richness of its links, nodes, relations and rules.

The Massive Emerge Principle. *The more diverse the richer.*

A richer link contributes to the richness of its ending nodes, and a richer node supports the richness of its connected links. So we have the following strategy.

Strategy for a New Node to be Rich. *Link to enrich semantic links, that is, the new semantic link should be relevant to the potential neighbor semantic links.*

The addition of a new semantic link to SLN reflects the purpose of the new node. If the semantic link information

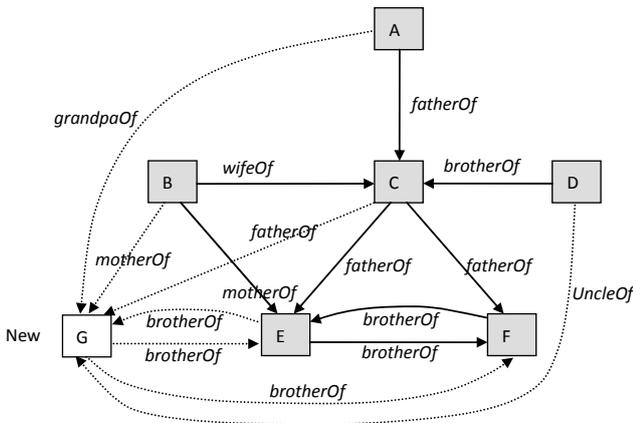


Fig. 4. Effect of adding a new node to a Semantic Link Network: the new node helps old nodes become richer, which in turn helps itself become rich quickly.

is unknown, then the strategy for a new node to become rich can be simplified as follows:

Link to the richer node, as the richer node owns more diverse semantic links, which have higher probability to own the semantic links that can participate in reasoning with the new semantic link.

Characteristic 12. *The behavior of adding semantic links or nodes to a Semantic Link Network tends to make a semantic path shorter.*

Different from the hyperlink network of the Web, the SLN itself can derive out implied semantic links, so the network itself can change the richness of nodes. This helps the new nodes share the richness in the network. By selecting appropriate nodes to connect, the network provides the chance for a new node to become rich quickly. At the same time, the new node helps the old nodes become richer (every node in Fig. 4 benefits from the new node), which in turn helps itself richer. The massive emerging principle provides a way to map a flat network into a metric space to help discover the emerging semantic nodes and semantic links in a complex SLN.

7.2 The Simplest Emerge Principle

Multiple semantic paths may exist between semantic nodes in SLN. A phenomenon can be observed: The less information a semantic path contains, the easier people understand and remember.

The Simplest Emerge Principle. *Among multiple shortest semantic paths, the shortest path with least types of semantic links takes the priority to emerge as the semantics between two semantic nodes.*

Above principle can be explained by Shannon and Weiner's theory of information entropy [33]: *the lower entropy a path has, the less information it contains, therefore its semantics can be more easily understood.*

The simplest emerge principle focuses on a particular semantic path while the richness emphasizes on the status of a semantic node or a semantic link in the whole network. The richness changes with the evolution of the network, while the entropy of a semantic path is relatively stable unless the path itself changes. So, searching the emerging semantic path can use the following path entropy as heuristic function:

where x denotes the semantic links on the semantic path as random variable, and $p(x)$ is the probability mass function of x [27], $p(x) = \frac{\text{the number of semantic relation } x \text{ in the shortest semantic path}}{\text{total number of relations in this path}}$. This formula implies that the entropy of a path with only one type of semantic relation is zero.

Another heuristic function that can be used in searching is the influence set of a semantic link since a semantic link can contribute to not increase the number of semantic link types when its neighbor links are in its influence set.

Definition 11. *The simplest semantic path between two semantic nodes in SLN is the semantic path with the least types of semantic links between the two nodes in corresponding semantic closure SLN^+ .*

The following are three heuristic ways to find the emerging semantic path between two semantic nodes:

1. Find all semantic paths between the two nodes, and then select the one with the least path entropy.
2. Select appropriate local heuristic function during search, e.g., the number of different types of semantic links occur so far is less than four to ensure an understandable semantic chain.
3. Select the neighbor semantic link according to its influence set during search. Since the influence set is a subset of *Rules*, it is small compared with the large semantic link network.

The massive emerging principle reflects the law of movement of SLN, and the simplest emerging principle reflects the law of recognizing and understanding an existence. They contradict and unificate to evolve a semantic link network.

7.3 Strategies for Explaining Semantic Communities

Semantic community discovery actually is a process of finding the representatives of a network. The hierarchy of semantic communities can represent the semantics of the whole network. How to explain the semantics of a basic community? The key is to find the most basic semantic structure in a SLN.

Definition 12. *The semantic cover of SLN is formed by removing the semantic links that can be derived out from Rules. The minimum semantic cover of SLN is a semantic cover where no semantic link l exists such that $(M - l)^+ = M^+$.*

The minimum semantic cover of SLN can keep the semantics of a SLN and has fewer semantic links than original SLN. If all semantic links are equally important, then the minimum semantic cover of SLN can be regarded as the basic semantic structure in the SLN. However, if semantic links are not equally important, then we can generate the following structure as the most basic semantic structure of a SLN:

1. Regard the minimum semantic cover of SLN as a ranked graph. The ranks on semantic nodes and semantic links reflect their importance defined according to the massive emerging principle.
2. Generate its maximum spanning tree.
3. Assign the original semantic relation on each edge of the tree to get a minimum semantic tree.

The reasons of regarding the minimum semantic tree as the most basic semantic structure are as follows:

1. It uses the least number of semantic links to connect all semantic nodes; and,
2. It includes the important semantic links.

Different strategies can be adopted to explain a semantic community according to different requirements.

Strategy for Explaining Community 1. *Finding the minimum semantic cover, and then explaining it as the semantics of the community.*

Strategy for Explaining Community 2. *Finding the minimum semantic tree, and then explaining it as the semantics of the community.*

Strategy for Explaining Community 3. *Find nodes with top-k emerging richness and with the richest semantic links or paths between them, and then explain the semantics represented by these nodes and paths as the simplified semantic community.*

Daily life experiences tell us that the shortest semantic path is understandable only when it consists of small number of semantic links. For example, the semantic path consists of the following three different semantic links 'A—supervisorOf→B, B—classmateOf→C, C—brotherOf→D' can be understood as 'A is the supervisor of the classmate of the brother of D'. Since a lengthy semantic path is difficult to be understood, an understandable community should have a scale limit of understanding.

Strategy for Explaining Semantic Path (Simplest priority). *Find the simplest semantic path through the given semantic links or semantic nodes, and then explain the semantics of the semantic path.*

8. DISCOVERING AND LEARNING IN SLN-BASED E-LEARNING ENVIRONMENT

8.1 SLN-based E-Learning Environment

An e-learning environment includes human and learning resources. With continuous interaction between human, resources and the environment, sharable resources are accumulated and better organized, human knowledge are developed, and therefore, more useful semantic relations are discovered. The increased resources, relations and knowledge in turn help discover more relations. With the expansion of the linked resources, discovering the hidden semantic community and the emerging semantic relations in a large network of resources becomes increasingly important.

A distributed SLN-based learning environment can be in client/server architecture or peer-to-peer (P2P) architecture. The client/server architecture consists of personal SLNs and local SLN servers as depicted in Fig.5. Users play two roles: SLN creators and learners. They construct and browse personal SLNs in their PCs. They can also download SLNs and the SLN relevant software from the corresponding SLN servers and upload their SLNs and the newly developed software to appropriate servers for sharing. The seemingly isolated local SLNs could be linked with each other if common semantic nodes or semantic links are shared by other local SLNs. Semantic communities may exist in personal SLNs, in the SLNs stored in the same server, and in the global SLN. The P2P architecture can establish a scalable self-organized network platform for the learning environment. Servers also play the role of client, nodes can join or depart the network at any time, and global information is usually

work at any time, and global information is usually not available due to the large scale network. Reference [43] introduced a way to make use of local structure to efficiently route queries in unstructured P2P network. The work shows that query over the P2P SLN could get better performance than the ordinary network without semantics. It also shows the role of classification in raising the performance of an unstructured P2P network.

A resource can be added to the environment in two ways: add the new resource to a category, or link the new resource to existing resource by a semantic link. Some semantic links can be automatically discovered if resources can be described by the primitive semantic space or are linked to the known resources.

8.2 Advantages of SLN in Supporting e-Learning

Compared with the Web's hyperlink structure, the SLN has the following advantages in supporting e-learning:

1. Support semantics-rich browsing and semantic reasoning at both the instance level and the abstraction level. Browsing the instance level helps users know the content of the next-hop by checking the surrounding semantic links. The semantic link reasoning rules can extend the foreseeing to multiple hops [41]. Browsing the abstraction level, users can obtain abstract knowledge about the underlying content.
2. Provide users with not only the answer but also relevant contents that semantically link to the answer.

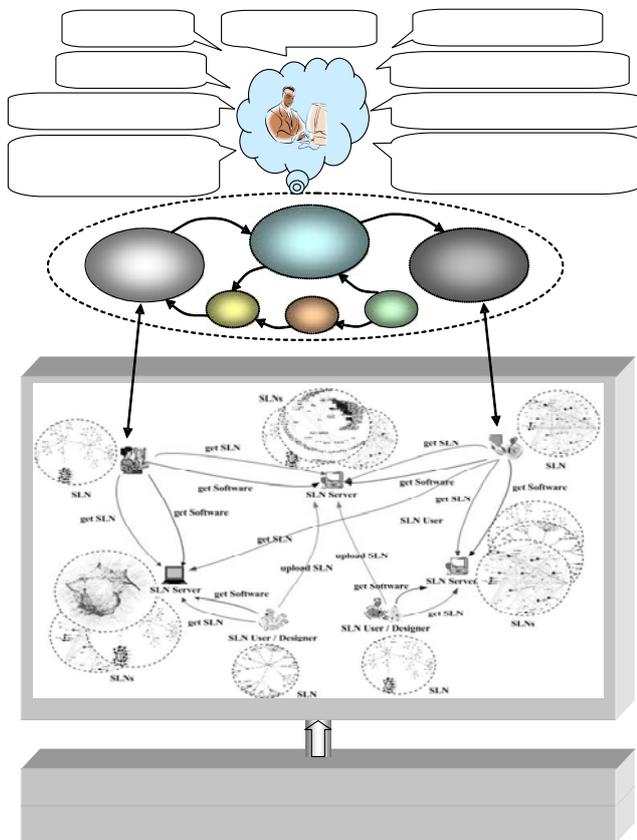


Fig. 5. Discovering and learning with the SLN-based environment.

3. Derive out the semantics of a node or propose conjectures by relational reasoning. For example, the semantics of a node can be fixed if it is a subtype of a node with known semantics.
4. SLN reasoning includes not only the relation reasoning but also the inductive, deductive and analogical reasoning [42]. The inductive reasoning and various statistic approaches can help generate the implied semantic relations in existing SLNs. Analogical reasoning and inductive reasoning of SLN can help understand new semantic relations by existing semantic relations, and can inspire creative thinking and broaden knowledge of learners.

Semantic links can help users know the learning content in multiple aspects and at different levels, for example:

1. The *causeEffect* (ce) link can help people know the effect when they know the cause, and know the cause when they know the effect. The *causeEffect* links can be chained for *causeEffect* reasoning.
2. The similar link can help people know similar contents when learning.
3. The sequential link can help arrange learning contents sequentially step by step.
4. The reference link can help people get appropriate references, which can help them better understand the content.
5. The implication link can help people know the underlying knowledge about the current content.

8.3 Roles of Discovering Semantic Communities and Emerging Relations in e-Learning

The suggested approaches to discovering semantic communities and emerging relations can provide an e-learning environment with the following functions:

1. Users can focus on learning relevant knowledge from a semantic community that matches their interest by discovering semantic communities where semantic nodes may come from different local SLNs.
2. Explain a concept by its subtypes, instances and similar concepts.
3. Explain a network of learning resources by its community hierarchy.
4. Explain the semantics of a community by its minimum semantic cover and maximum spanning
5. Filter out the special SLN with relations of definite types to match learning interest.
6. Find concepts that have semantic relations with a given concept to expand user's knowledge.
7. Automatically generate learning path. The *causeEffect*, *implication* and *sequential* semantic links can help e-learning system to generate a learning path in a complex SLN. Further, the reasoning feature can generate the shortest semantic path for the learning path. As shown in Fig.5, if a sequential relation between communities is found (e.g., $A \text{---seq---} B \text{---seq---} C$, and A is in community_A, B is

in $community_B$ and C is in $community_C$), the content recommendation mechanism can recommend content according to the sequence (e.g., recommend $community_A$ to the learner before $community_B$, and recommend $community_B$ before $community_C$).

8. Automatically cluster and recommend learning resources according to emerging semantic relations and user's input.
9. Discover the local emerging semantics of SLN according to local emerging semantic relations. This is often required in large-scale SLN
10. ~~Application~~ ~~the~~ influence of adding a semantic link or deleting a semantic link as well as adding a rule or deleting a rule according to the rule clusters. It is useful for users to know which semantic links or rules are influenced after operating SLN.

9. CONCLUSION

Discovery and learning are two inseparable aspects of the entire human knowledge development process. To support such a process, an e-learning system should be able to discover the semantic communities and the emerging semantic relations in dynamic complex semantic link network of learning resources.

Structure and semantics are two inseparable aspects of human's representation, organization and understanding of the linked contents. Previous graph-based community discovery approach can discover network structure, but they are not suitable for SLN, which reflects both structure and semantics. Discovering semantic communities in a SLN can help reveal the nature of a complex content network of semantic links.

Major contribution of this paper includes the following four aspects:

1. It develops the Semantic Link Network SLN as a loosely coupled semantic data model to synergy normal organization and self-organization for organizing various resources. It is suitable for organizing large-scale learning resources in a dynamic, decentralized and loosely coupled e-learning environment.
2. It proposes two types of approaches to discovering semantic communities in SLN: one is based on graph structure and reasoning (including the decomposition approach and the construction approach), and the other is based on the meta-semantics (rule cluster and classification tree), which provides solutions to discovering semantic communities in large-scale SLN.
3. It suggests principles and strategies to discover the emerging semantic relations, which can help learners understand the semantic relations in complex SLN.
4. It incorporates the proposed approaches, principles and strategies into a SLN-based e-learning environment, which enables users to discover and learn in a context of SLN.

At different stages of learning process, structure and semantics may take different priorities. The semantic relation reasoning and users' participation for learning influence the network structure. Such an e-learning environment can provide diverse means for users at different stages and different processes of learning and discovery. The following issues on SLN need further efforts: effective approaches to automatically discover semantic links from various resources according to some definitions on concepts and rules in the primitive semantic space; efficient navigation mechanism in a decentralized and loosely coupled SLN data model; definition of domain relations and rules; new approaches to discover emerging semantic communities and relations as well the criteria for evaluating their quality; and, probabilistic SLN as a data model for dealing with uncertain resources and uncertain relations.

ACKNOWLEDGMENT

Research was supported by National Basic Research Program of China (973 project no. 2003CB317000). I would like to thank my students Y. Sun, B.Xu, L. Cao, L.Feng, C.He and J. Zhang as well as former students H. Jin and L. Pang for their help with this research.

REFERENCES

- [1] R. Albert and A.L. Barabasi, "Statistical mechanics of complex networks," *Review of Modern Physics*, 74(1), (2002), pp.47-97.
- [2] Y. Y. Ahn, et al., "Semantic Web and Web 2.0: Analysis of Topological Characteristics of Huge Online Social Networking Services," *Proceedings of the 16th international conference on World Wide Web, WWW '07, May 2007, ACM Press.*
- [3] B. Aleman-Meza, et al., "Social Networks: Semantic Analytics on Social Networks: Experiences in Addressing the Problem of Conflict of Interest Detection," *Proceedings of the 15th international conference on World Wide Web, WWW '06, ACM Press.*
- [4] R. M. Assuncao, et al., "Efficient Regionalization Techniques for Socio-Economic Geographical Units Using Minimum Spanning Trees," *International Journal of Geographical Information Science*, 7(20) (2006) 797-811.
- [5] L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan, "Group Formation in Large Social Networks: Membership, Growth, and Evolution," *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '06, August 2006, ACM Press.*
- [6] T. Berners-Lee, J. Hendler, and O. Lassila, "Semantic Web," *Scientific American*, 284(5) (2001)34-43.
- [7] U. Brandes, "A Faster Algorithm for Betweenness Centrality," *Journal of Mathematical Sociology*, 25(2) (2001)163-177.
- [8] P. Brusilovsky, "KnowledgeTree: A Distributed Architecture for Adaptive e-Learning," *International World Wide Web Conference*, pp.104-113, New York, NY, USA, 2004.
- [9] D. Cai, et al, "Mining Hidden Community in Heterogeneous Social Networks," *Proceedings of the 3rd International Workshop on Link Discovery, LinkKDD '05, ACM Press, August 2005.*
- [10] R. Farrell, S. D. Liburd and J. C. Thomas, "Dynamic Assembly of Learning Objects," *In Proceedings of WWW2004, May, 2004, New York, USA, pp.162-169.*
- [11] L. Freeman, "A Set of Measures of Centrality Based Upon Betweenness," *Sociometry*, (40) (1977)35-41.

- [12] M. Girvan, and M. Newman, "Community Structure in Social and Biological Networks," *Proceedings of National Academy of Sciences of the United States of America*, (2002)8271-8276.
- [13] J. Golbeck and J. Hendler, "Inferring Binary Trust Relationships in Web-based Social Networks," *ACM Transactions on Internet Technology*, Vol. 6, Issue 4, November 2006, ACM Press.
- [14] R. Guimera, et al., A. "Self-similar Community Structure in Organizations," *arXiv:cond-mat/0211498*, (22) (2002).
- [15] L. Hossain, A. Wu, and K. K. S. Chung, "Social Networks and Coordination Patterns: Actor centrality correlates to project based coordination," *Proceedings of the 2006 20th Anniversary Conference on Computer Supported Cooperative Work CSCW '06*, ACM Press, November 2006.
- [16] H. Kautz, B. Selman, and M. Shah, "Referral Web: Combining Social Networks and Collaborative Filtering," *Communications of the ACM*, 3 (40) (1997) 63-65.
- [17] B. W. Kernighan and S. Lin, "An Efficient Heuristic Procedure for Partitioning Graphs," *Bell System Technical Journal*, 49, (1970), pp. 291-307.
- [18] R. Kumar, J. Novak, and A. Tomkins, "Structure and Evolution of Online Social Networks," *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining KDD '06*, ACM Press, August 2006.
- [19] Q. Li, R.W.H.Lau, T.K.Shih, and F.W.B.Li, Technology Supports for Distributed and Collaborative Learning over the Internet, *ACM Transactions on Internet Technology*, vol.8, no.2, (2008).
- [20] M. Makrehchi and M. S. Kamel, "Learning Social Networks from Web Documents Using Support Vector Classifiers," *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence WI '06* December 2006, IEEE Computer Society.
- [21] Y. Matsuo, et al, "POLYPHONET: an Advanced Social Network Extraction System from the Web," *Proceedings of the 15th International Conference on World Wide Web WWW '06*, ACM Press.
- [22] M. E. J. Newman and M. Girvan, "Finding and Evaluating Community Structure in Networks," *Physical Review E*, vol.69, no.2, (2004), DOI: 10.1103/PhysRevE.69.026113, <http://link.aps.org/abstract/PRE/v69/e026113>.
- [23] M. E. J. Newman, "Fast Algorithm for Detecting Community Structure in Networks," *arXiv:cond-mat/0309508*, (22) (2003).
- [24] M. E. J. Newman, "Who is the Best Connected Scientist? A Study of Scientific Coauthorship Networks," *Phys Rev. E* (64) (2001).
- [25] M. E. J. Newman, "Scientific Collaboration Networks. II. Shortest Paths, Weighted Networks, and Centrality," *Physical Review E*, (64) 016132.
- [26] F. Radicchi, et al., "Defining and identifying Communities in Networks," *PANS*, 101, 2004, pp.2658-2663.
- [27] A. Renyi, "On Measures of Entropy and Information," *Proceedings of the Fourth Berkeley Symposium on Math. Statist. Prob.*, University of California Press, Berkeley, 1961, pp. 547-561.
- [28] J. R. Tyler, D. M. Wilkinson, and B.A. Huberman, "Email as Spectroscopy: Automated Discovery of Community Structure within Organizations," *Proceedings of the First International Conference on Communities and Technologies*, 2003, pp.81-96.
- [29] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank Citation Ranking: Bringing Order to the Web," *Technical Report, Stanford Digital Libraries SIDL-WP-1999-0120*, (1999).
- [30] A. P. Pons, "Object Prefetching Using Semantic Links," *ACM SIGMIS Database*, vol.37, issue 1, (2006), pp. 97-109.
- [31] J. M. Pujol, R.Sangüesa and J. Delgado, "Extracting Reputation in Multi Agent Systems by Means of Social Network Topology," *Proceedings of the first international joint conference on Autonomous agents and multiagent systems, AAMAS '02*, July 2002, ACM Press.
- [32] M. J. Rattigan, M. Maier and D.Jensen, Graph Clustering With Network Structure Indices, *The 24th Annual International Conference on Machine Learning (ICML 2007)*, www.machinelearning.org/proceedings/icml2007/papers/407.pdf.
- [33] C.E. Shannon and W.Weiner, *The Mathematical Theory of Communication*. University of Illinois Press, 1963.
- [34] J. P. Vert, Adaptive Context Trees and Text Clustering, *IEEE Transactions on Information Theory*, 47, (5), (2001), 1884-1901.
- [35] D. M. Wilkinson and B.A. Huberman, "A Method for Finding Communities of Related Genes," *Proceedings of National Academy of Sciences of the United States of America*, (101) (2004), pp.5241-5248.
- [36] F. Wu and B. A. Huberman, "Finding Communities in Linear Time: A Physics Approach," *The European Physical Journal B*, vol. 38, (2004) 331-338.
- [37] B. Yang, W. K. Cheung and J. Liu, "Community Mining from Signed Social Networks," *IEEE Transactions on Knowledge and Data Engineering*, 19(10) (2007) 1333-1348.
- [38] B. Yang and J. Liu, "Discovering Global Network Communities Based on Local Centralities," *ACM Transactions on the Web*, Vol.2, No.1, Feb, 2008.
- [39] W. W. Zachary, "An Information flow model for conflict and fission in small groups," *Journal of Anthropological Research*, 33, (1977), pp.452-473.
- [40] H. Zhuge, *The Knowledge Grid*. World Scientific, Singapore, 2004.
- [41] H. Zhuge, R. Jia, and J. Liu, "Semantic Link Network Builder and Intelligent Semantic Browser," *Concurrency and Computation: Practice and Experience*, 16(14): (2004), pp.1453-1476.
- [42] H. Zhuge, "Autonomous Semantic Link Networking Model for the Knowledge Grid," *Concurrency and Computation: Practice and Experience*, 7(19) (2007), pp.1065-1085.
- [43] H. Zhuge and X. Li, "Peer-to-Peer in Metric Space and Semantic Space," *IEEE Transactions on Knowledge and Data Engineering*, 6(19) (2007), pp.759-771.



Hai Zhuge is a professor at the Chinese Academy of Sciences' Institute of Computing Technology. He is also the chief scientist and the former director of the academy's Key Lab of Intelligent Information Processing. He is the chief scientist of the China National Semantic and Knowledge Grid Research Project and the founder of the China

Knowledge Grid Research Group. His main research interests include Knowledge Grid, Resource Space Model (RSM), Semantic Link Network (SLN) and Knowledge Flow. He presented over ten keynotes in international conferences. He initiates the International Conference on Semantics, Knowledge and Grid (SKG, www.knowledgetgrid.net). He is an associate editor of the *Future Generation Computer Systems*, the associate editor-in-chief of *Journal of Computer Science and Technology*, and on the editorial board of *IEEE Intelligent Systems*. He is the author of *The Knowledge Grid* and *The Web Resource Space Model*. He was the top scholar in systems and software engineering during 2000 to 2004, according to a *Journal of Systems and Software* assessment

report. His publications appeared in CACM, Computer, IEEE TKDE, IEEE TPDS and ACM TOIT. He received 2007's Innovation Award of China Computer Federation for his fundamental theory of the Knowledge Grid. He is a senior member of IEEE and CCF. Email: zhuge@ict.ac.cn; www.knowledgegrid.net/~h.zhuge.