	OPAALS PROJECT Contract n° IST-034824

WP4: Distributed Accountability, Identity, and Trust

Del4.4 - Distributed Collaborative Knowledge Sharing Framework

	Project funded by the European Community under the "Information Society Technology" Programme

Contract Number: IST-034824

Project Acronym: OPAALS

Deliverable N°: 4.4

Due date: 31/09/2007

Delivery Date:28/02/2008

Short Description: describe the framework and novel tools for collaborative knowledge sharing, also present novel algorithms for extracting and interpreting knowledge for sharing knowledge

Author: Huaiguo Fu, WIT

Partners contributed: WIT

Made available to: OPAALS Consortium and European Community

Versioning		
Version	**Date**	**Name, organization**
1.1	10 Oct 2007	Huaiguo Fu, WIT
1.2	15 Apr 2008	Huaiguo Fu, WIT

Quality check

Internal Reviewers: Amir reza Razavi, University of Surrey

Gerard Briscoe, LSE

Dependences:

Work Packages	Indicate which WP could benefit from this Deliverables and if possible with planned tasks, including a clear justification
Partners	indicate which partner could benefit from the reading of this deliverable
Domains	specify the scientific domains involved in this deliverables and clarify to which extend they are covered by this report
Targets	specify which is the target of this deliverable (Domain researchers, public administrations, SMEs, Scientific communities on specific domains, etc.)

Deliverable for OPAALS Task 4.2: Distributed Collaborative knowledge sharing

OPAALS Project

Deliverables 4.4

Distributed Collaborative knowledge sharing Framework

Huaiguo FU

Telecommunications Software Systems Group

Waterford Institute of Technology
Waterford, Ireland

OPAALS

Open Philosophies for Associative Autopoietic Digital Ecosystems

Contents

List of Figures

Chapter 1

Introduction

Contents

1.1 The state of the art in Knowledge Sharing

Knowledge sharing is becoming the basic and important behavior and culture of individual, organization, community and society. Knowledge sharing is also the key power of the de-

velopment of individual, organization, community and society. Knowledge sharing can help members of organizations, communities or individual to share common knowledge to achieve some common tasks.

Knowledge sharing attracts many research works [35, 10, 13, 19, 20, 26, 22, 36, 2] and the applications of knowledge management [28, 9, 24, 23, 31].

Knowledge sharing in the digital ecosystem is a new topic. We need to develop new models and techniques to adapt the features of the digital ecosystem and address the challenges of distributed collaborative knowledge sharing.

1.2 Challenges of distributed collaborative knowledge sharing

In the natural world, an ecosystem is a system whose members benefit from each other's participation via symbiotic relationships. In digital environment, a sharing collaborative system is formed by the digital components such as digital data, information and knowledge. Collaborative knowledge sharing between participants of the ecosystem is one core of the digital ecosystem. The digital ecosystem should be an effective platform to enable the participants of the ecosystem to learn and share divers knowledge such as domain knowledge, skills, expertise, useful information, experiences and perspectives from others.

Knowledge is the high level digital component. Knowledge is useful and appropriate collection of information, and understanding patterns. Sharing knowledge between the members of ecosystem is one main character of ecosystem. In the digital ecosystem, we need to face two main challenges for distributed collaborative knowledge sharing: what to share and how to share.

1.2.1 What to share

What to share in the digital ecosystem is the first challenge for distributed collaborative knowledge sharing. Sometimes there are too much shared knowledge and it's hard to know what is suitable. Sometimes we need to share some knowledge, but it is hard to find it. In the digital ecosystem, we need to investigate the problems of what-to-share and discuss the solution to address the problems.

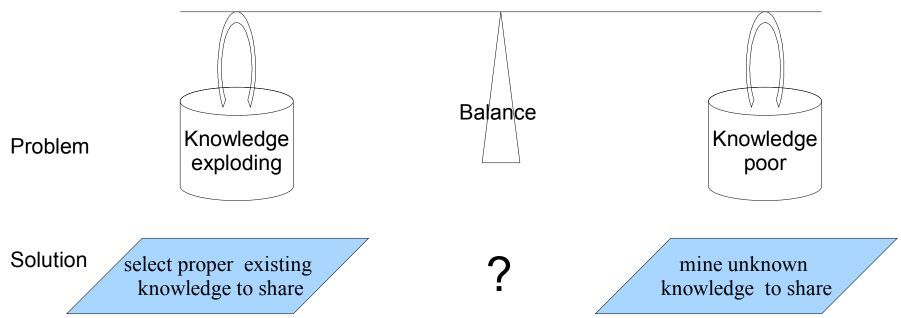

Figure 1.1: Problems and solutions for what-to-share

Figure 1.1 shows the three problems and two solutions. On the one hand, we are in the times of knowledge exploding. No one has the ambition to know or learn all the knowledge for one subject, and it is impossible for us to master all the knowledge completely for a special field, either. Of course, we need not to know all knowledge. We only need to learn and share some limited and necessary knowledge. Obviously, we must learn how to learn, how to get proper information from the ocean of knowledge and how to master latest and useful knowledge. It is one key issue how the digital ecosystem analyze and learn to provide and select proper, necessary and useful knowledge to share.

On the other hand, "data rich, information poor and knowledge poor" is a general situation for many companies. More and more valuable data is available, but the related and valuable information and knowledge to the end user are not exploited from the data. The digital ecosystem should address the issue to extract useful knowledge from data. The digital ecosystem not

only share existing knowledge, but also mine non-trivial, previously unknown and potentially useful information and knowledge to share.

Between above two problems, there is the third problem: balance of knowledge selection and knowledge discovery according the demand of user. For a knowledge requirement, we can search the proper knowledge from existing knowledge or extract knowledge from data. We need to develop the techniques of self-organizing and self-adapting to analyze the requirement and then choose different way to learn and get knowledge.

1.2.1.1 Data, information and knowledge

What to share in the digital ecosystem is the base to develop a framework for collaborative knowledge sharing. We need to understand the definitions of data, information and knowledge. Data leads to information, and information leads to knowledge.

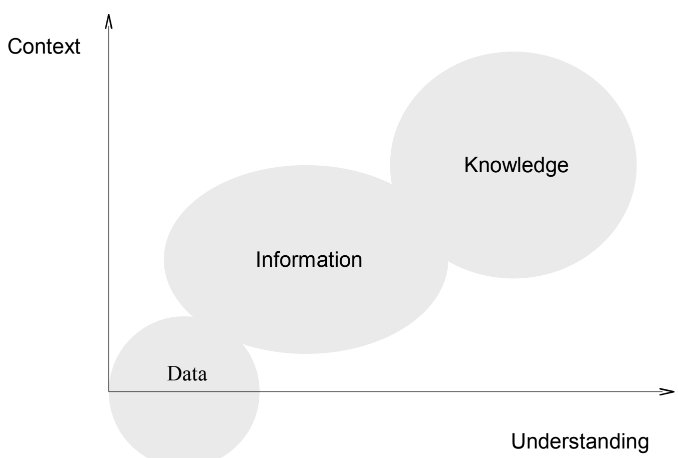

Figure 1.2: Data, information and knowledge

Data is collection of digital symbols without context. Data comes in a lot of types or formats such as numbers, words or pictures. Information is data that has context and meaning. Therefore we can understand information rather than data. The data becomes information once it is put into a framework or structure that provides context. Common ways in which you may

have used data and converted it into information are in using spreadsheet or database programs to solve problems. Information can be converted to knowledge when information is useful to understand about a subject, and then make decisions, form judgements, take opinions or make a forecast.

1.2.1.2 knowledge discovery in digital ecosystem

In the digital ecosystem, some knowledge can be existing, but some knowledge can be previously unknown, implicit, hidden in large data. We therefore should extract the knowledge from large data with the techniques such as data mining or KDD (Knowledge Discovery in Databases). The techniques of data mining are widely used in research and application to look for relationships and knowledge that are implicit in large volumes of data and are interesting in the sense of impacting an organization's practice. For example, data mining techniques can help companies to provide better, customized services and support decision making. Hence, the techniques of data mining can be used to extract knowledge for the digital ecosystem.

1.2.2 How to share

How to share knowledge is another challenge for collaborative knowledge sharing. We need to develop efficient, dynamic, flexible and scalable models and services for distributed knowledge sharing in the digital ecosystems.

1.2.2.1 Platforms of knowledge sharing

There are different models to share knowledge in different environments. We consider three different sharing environments or platforms: Organization, Community and Digital ecosystem networks (see Figure 1.3).

One organization such as a company or an university is a social arrangement which engages collective goals, which controls its own performance in a bounded and fixed environment. Organization has its own inside knowledge system. The members of an organization share

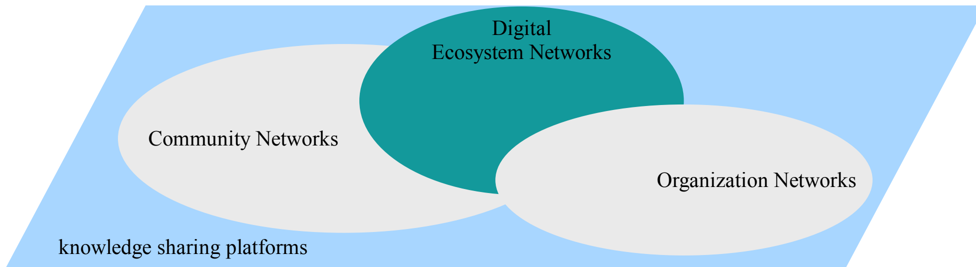

Figure 1.3: Platforms of knowledge sharing

knowledge via intranet, local networks, internet or in person. Organization networks are a formal and structured platform that focuses on domain knowledge sharing that are critical to the organization.

One community is a group of people who have common interest or work. The members of a community can be professionals within an organization, or in several organizations, or they can simply form a non-work-related community. They may not all know each other, yet feel a sense of community because they have similar interests and face similar challenges. They can share tips, hints, ideas, and knowledge on-line, or in person. They realize the value of sharing knowledge with their peers, and learning from each other. Communities are often informal and loosed.

Organization networks and community networks are two overlapped platforms. Organization networks and community networks can not cover all sharing platforms. For example, individual knowledge sharing via internet needs the new platform. Therefore we propose digital ecosystem networks to cover all sharing platforms.

1.2.2.2 Active, passive and collaborative knowledge sharing

The behavior of sharing can be active, passive or collaborative. We focus on the collaborative knowledge sharing. For example, the Figure 1.4 shows P2P based collaborative knowledge sharing.

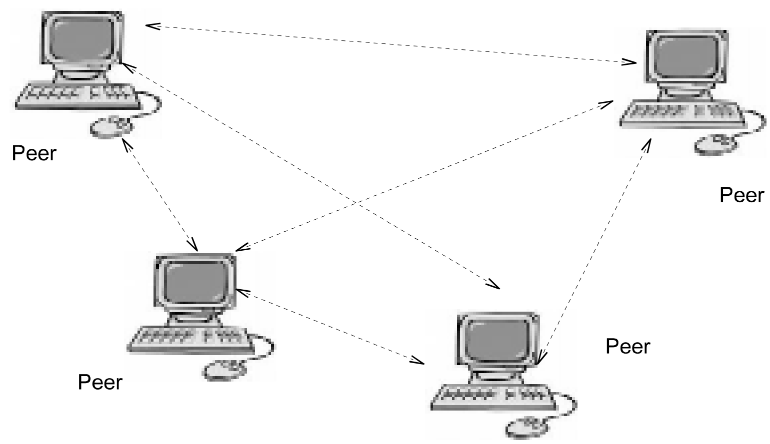

Figure 1.4: P2P based collaborative knowledge sharing

1.3 Contributions

Our research works focus on distributed knowledge sharing for digital ecosystem. We have the following contributions:

- Development of a framework for collaborative knowledge sharing between participants of the ecosystem

  We propose the framework for collaborative knowledge sharing that includes Infrastructure, Sharing platforms, Sharing requirement, Knowledge resource, Knowledge engine and Knowledge Sharing. Knowledge engine is for acquirement of knowledge that responds to where and how to find knowledge. Knowledge Sharing is for sharing knowledge in digital ecosystem. It focuses on the answer to how to share knowledge.

- Propose new strategy of sharing distributed knowledge for digital ecosystems.

  We propose the integration of community networks, organization networks and digital ecosystem networks three overlapped platforms for sharing distributed knowledge for digital ecosystems. The organization networks provide a platform for local knowledge

sharing, mapping and storage by one organization, e.g. one enterprise. The community networks overlap with organization networks to share distributed knowledge by common topics for a community, e.g. OKS research group. The digital ecosystem networks provide more wide, open and dynamic knowledge sharing space by the integration of community networks , organization networks and other knowledge that are not included in community networks and organization networks.

- Propose Data Mining and KDD for digital ecosystems

- Propose Formal Concept Analysis (FCA) for digital ecosystems. The formal concepts can be used to explore, present and interpret the knowledge from the large data in digital ecosystem so that facilitates knowledge sharing in digital ecosystem.

  FCA is an ontology analysis model. Using FCA, we can analyze, define, refine, merge, and share ontologies [34, 17].

- Development of new algorithms of distributed knowledge sharing for digital ecosystem

  We are developing new algorithms for extracting knowledge in digital ecosystems. The algorithm is based on formal concept lattice. Theoretical foundation of concept lattice founds on the mathematical lattice theory. Lattice is a popular mathematical structure for modeling conceptual hierarchies. Concept lattice is a method for deriving conceptual structures out of data. From concept lattice, we can study the relations between objects and attributes in a formal context, and how objects can be hierarchically grouped together according to their common attributes. Certain object subset and the set of their common attributes can represent each other, such duality sets form a formal concept, which the attribute subset is called intent and the object subset is called extent. Among the formal concepts, there exists an order relation, they form a complete lattice: concept lattice. Each node in the lattice is a concept and the corresponding graph (Hasse diagram) that is considered as the generalization and specialization relationships between concepts. Such graphical structure represents directly and visually the relations of conceptual hierarchies. It allows us to analyze and mine the complex data for such as classification, association rules mining, clustering, etc. Furthermore, concept lattice also provides an effective tool of knowledge visualization.

1.4 Organization

The rest of this deliverable is organized as follows. The next chapter presents the framework for collaborative knowledge sharing between participants of the ecosystem. The chapter 3 presents novel tools for collaborative knowledge sharing. The novel algorithms for extracting and interpreting knowledge is introduced in chapter 4. The deliverable ends with a conclusion in chapter 5.

Chapter 2

Framework for collaborative knowledge sharing

Contents

In the digital ecosystem, the first challenge is ”what knowledge we can share in the digital ecosystem”. Digital ecosystem is an open, demand-driven, self-organizing collaborative environment. When we need to share some knowledge, the system should provide the inherent or existing knowledge, or the system can investigate, compare, summarize, merge, decompose and analyze the relevant data, information or knowledge to extract the required knowledge from data if the knowledge is previously unknown, implicit, hidden in large data. The system should teach the participant to learn and understand the knowledge.

How to share knowledge is another challenge for collaborative knowledge sharing. We

need to develop efficient, dynamic, flexible and scalable models and services for distributed knowledge sharing in the digital ecosystems.

In this chapter, we introduce a framework for collaborative knowledge sharing between participants of the ecosystem. The framework has the following features:

- Address the challenges for collaborative knowledge sharing between participants of the ecosystem;
- Analyze the environment of sharing knowledge and propose new platforms of sharing distributed knowledge for digital ecosystems: community networks, organization networks and digital ecosystem networks. The networks can be virtual networks. The sharing platforms can integrate all digital resource for knowledge sharing. For example, the platforms can provide the behaviours recorded through distributed accounting to facilitate collaboration through the trading of information using community currencies in a manner that optimises the operation of the ecosystem as a whole;
- Describe the knowledge stream in the framework and propose to use Sharing Requirement to control the knowledge stream with eco-features.

2.1 Framework for collaborative knowledge sharing

Knowledge is the core of the system of for collaborative knowledge sharing. Knowledge discovery, knowledge generation, knowledge collection, knowledge transformation, knowledge distribution, and knowledge sharing constitute a cycle of knowledge sharing. We propose a framework for distributed knowledge sharing (see figure 2.1) to control the knowledge stream in the cycle in the digital ecosystems and facilitate collaborative knowledge sharing.

The framework includes 6 parts: Infrastructure, Sharing platforms, Sharing requirement, Knowledge resource, Knowledge engine and Knowledge Sharing. The core of the framework is sharing platforms that provide the digital eco-environment of collaborative knowledge sharing. We propose the integration of community networks, organization networks and digital ecosystem networks three overlapped platforms for sharing distributed knowledge in digital

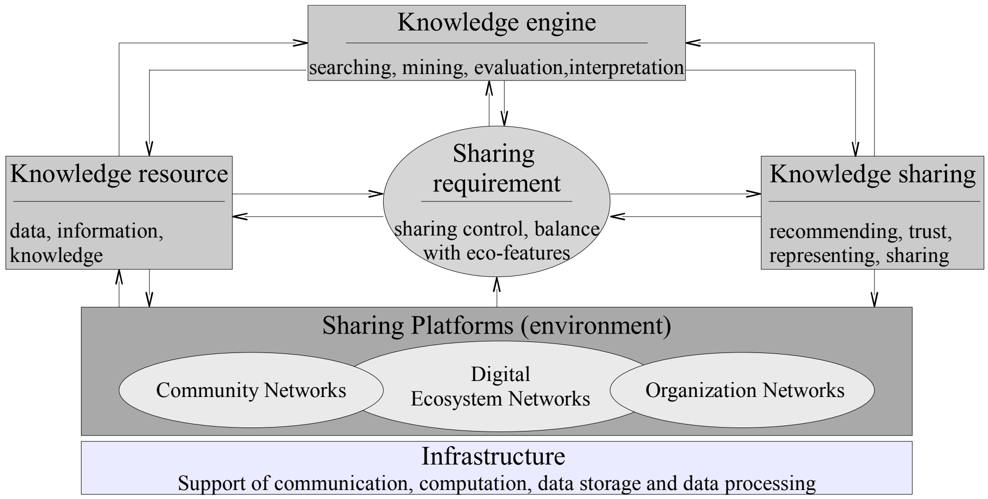

Figure 2.1: Framework for knowledge sharing

ecosystems. The sharing platforms contain all digital resource in the digital ecosystems so that the platforms can afford the knowledge resource including various data, information and knowledge. Especially, the sharing requirements are derived from the sharing platforms.

Knowledge engine is for acquirement of knowledge that responds to where and how to find knowledge. Knowledge Sharing is for sharing knowledge in digital ecosystem. It focuses on the answer to how to share knowledge.

We introduce the main parts of the framework in the following sections.

2.2 Sharing requirements

The digital ecosystem is an user-driven and demand-driven system. The source of sharing knowledge is sharing need. The digital ecosystem and participants of the ecosystem all can produce the sharing requirements. The requirements from participants of the ecosystem are specific sharing requirements. The features of ecosystem, self-adapting, self-organizing and self-management of the digital ecosystem can enable the sharing environment to automatically

generate sharing requirements to recommend sharing knowledge to the participants of the ecosystem.

Sharing requirement service can analyze, interpret and transform the different needs, organize Knowledge resource and Knowledge engine to provide required knowledge, and deploy Knowledge distributing and sharing. Sharing requirement service will control and balance other services to adapt the sharing task according to the eco-features and the features of sharing environment.

2.3 Sharing environment

The sharing environment is constituted by three different and associated platforms: community networks, organization networks and digital ecosystem networks. The sharing environment is the base of knowledge sharing and it facilitates to provide efficient, dynamic, flexible and scalable models and services for distributed knowledge sharing in the digital ecosystems.

The organization networks provide a platform for local knowledge sharing, mapping and storage by one organization, e.g. one enterprise. The community networks overlap with organization networks to share distributed knowledge by common topics for a community, e.g. OKS research group. The digital ecosystem networks provide more wide, open and dynamic knowledge sharing space by the integration of community networks, organization networks and other knowledge that are not included in community networks and organization networks.

The sharing platforms provide an open, self-adapting, self-organizing environment with the following services:

- Knowledge organization, classification and clustering
- Dynamic distributed knowledge indexing
- Dynamic description of knowledge with keywords, abstract, links and references
- Embedded engine of data mining
- Web-based searching and extracting

2.4 Knowledge engine

Knowledge engine is for acquirement of knowledge that responds to where and how to find knowledge. Knowledge engine provides methods of knowledge searching if the knowledge is already existed, and the techniques and algorithms of data mining if knowledge is implicit in data. When knowledge is acquired, Knowledge engine will provide the services of evaluation, interpretation for knowledge. Knowledge engine also contains learning service to learn to adapt the environment and acquire and understand sharing knowledge.

2.5 Knowledge sharing

Knowledge Sharing focuses on the answer to how to share knowledge. Knowledge Sharing will provide the methods, models and strategies to share distributed knowledge. Knowledge Sharing also has the services to trust, accept, store, distribute and update knowledge in digital ecosystem.

2.6 Infrastructure

In this framework, the basic infrastructure is Peer-to-peer (or abbreviated P2P) architecture that is a type of network in which each workstation has equivalent capabilities and responsibilities. This differs from client/server architectures. Generally, P2P networks are used for sharing files, but a P2P network can also mean Grid Computing. The infrastructure of P2P provides a novel distributed environment, computational model, and unprecedented opportunities for unlimited computing and storage resources. It's distinguished from conventional distributed computing by its focus on large-scale resource sharing, innovative applications, and, in some cases, high-performance orientation. Grids can be used as effective infrastructures for distributed high-performance computing and data processing. P2P environment provides high performance computing facilities and transparent access to them in spite of their remote location, different administrative domains and hardware and software heterogeneous charac-

teristics. In this framework, P2P provide the support of communication, computation, data storage and data processing for distributed knowledge sharing in digital ecosystem.

Chapter 3

Novel tools for collaborative knowledge sharing

Contents

In this chapter, we present novel tools: Data Mining and KDD, and Formal Concept Analysis for collaborative knowledge sharing.

3.1 Data mining and KDD for digital ecosystems

In the digital ecosystem, some knowledge is existing, but some knowledge is previously unknown, implicit, hidden in large data. So we should extract the knowledge from large data. The techniques of data mining (DM) [11, 18] are widely used in research and application to look

for relationships and knowledge that are implicit in large volumes of data and are interesting in the sense of impacting an organization's practice. Hence, we propose to use the techniques of data mining to extract knowledge for the digital ecosystem. We will discuss main issues of application of data mining for the digital ecosystem.

Data mining is the automated analysis of large volumes of data for extracting knowledge that are implicit in data and are interesting in the sense of impacting an organization's practice. The techniques of data mining are widely used in research and application. For example, data mining can be applied in biology, medicine, physics, and engineering. Data mining techniques can help companies to provide better, customized services and support decision making.

Knowledge Discovery in Databases (KDD) is a process of discovering non-trivial, previously unknown and potentially useful information from a huge collection of databases. Data mining is a step of KDD process, but also referred to KDD. The KDD process is an interactive process. Because it involves many steps with decisions made by users, different KDD process may have different steps. However, there are some basic steps in common (see figure 3.1):

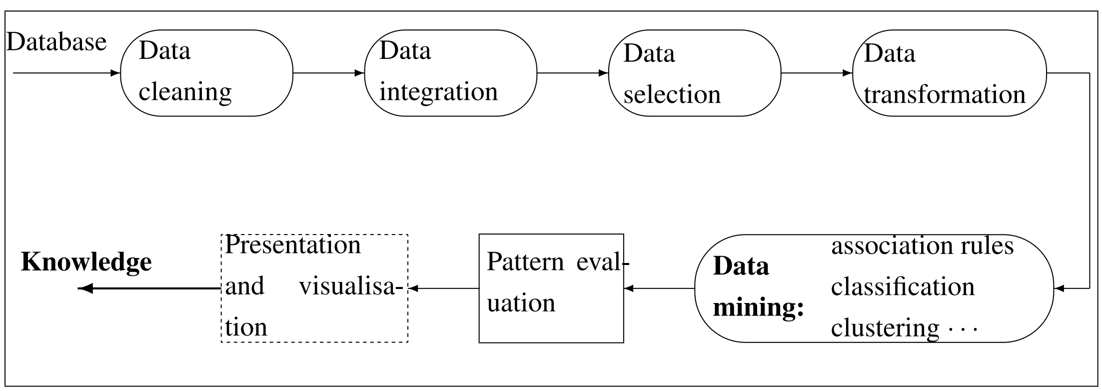

Figure 3.1: Data mining and KDD process

With years of research and developments, the computer-aided analysis such as data mining, machine learning, and statistics analysis of databases plays an increasingly important role in knowledge discovery and data analysis.

For example, we can use classification to extract models that predict for a novel molecule whether it will be active or inactive, for example, w.r.t. the protection of human cells against a

virus. Such a classifier can then be used to focus the expensive and time-consuming real chemical tests on the most promising candidates. The technology of association analysis and cluster analysis also can be used for extracting similarity, identification of co-occurring sequences and structures, interactions and relationships among some of proteins or genes.

In the digital ecosystem, knowledge is core for distributed knowledge sharing. We need develop the following models according to the features of knowledge:

- Knowledge Extraction and discovery
- Knowledge Sharing
- Knowledge Adapting and Transferring
- Knowledge Identifying
- Knowledge Evaluation
- Knowledge Visualization and presentation etc.

3.2 Formal concept structure for digital ecosystems

We propose to use FCA as a tool for data analysis, information management and knowledge representation in Digital ecosystem.

Formal Concept Analysis (FCA) provides a natural platform for data analysis and knowledge representation. FCA is different from some of the traditional, statistical means of data analysis and knowledge representation because of its focus on human-centered approaches. From the formal concepts, we can analyze data such as revealing stronger association or relation between itemset and the set of their common objects, classifying objects, generating implications of attributes or knowledge rules, extracting the hierarchical relation among formal concepts, etc. Concept lattice facilitate exploring, searching, recognizing, identifying, analyzing, visualizing, restructuring and memorizing conceptual structures.

The core of FCA is concept lattice. Theoretical foundation of concept lattice founds on the mathematical lattice theory [3, 15]. Lattice is a popular mathematical structure for modeling conceptual hierarchies. Concept lattice is a method for deriving conceptual structures out of data. From concept lattice, we can study the relations between objects and attributes in a formal context, and how objects can be hierarchically grouped together according to their common attributes. Certain object subset and the set of their common attributes can represent each other, such duality sets form a formal concept, which the attribute subset is called intent and the object subset is called extent. Among the formal concepts, it exists an order relation, they form a complete lattice: concept lattice. Each node in the lattice is a concept and the corresponding graph (Hasse diagram) that is considered as the generalization and specialization relationships between concepts. Such graphical structure represents directly and visually the relations of conceptual hierarchies. It allows us to analyze and mine the complex data for such as classification, association rules mining, clustering, etc [27].

The application of concept lattice has been an area of active and promising research in various fields such as knowledge discovery, information retrieval, software engineering and machine learning. For example, concept lattice structure [27] has shown to be an effective tool for finding frequent concepts for association rules [32, 33, 37].

Furthermore, concept lattice also provides an effective tool of knowledge visualization. Visualization plays an important role in data analysis. For example, Concept lattice can be used as an effective tool of visualization of biomedical data mining. Complex structures and sequencing patterns of genes and proteins are most effectively presented in graphs, trees, cubes, and chains by various kinds of visualization tools. Such visually appealing structures and patterns facilitate pattern understanding, knowledge discovery, and interactive data exploration.

Definition 3.2.1 Formal context *is defined by a triple* (O, A, R)*, where* O *and* A *are two sets, and* R *is a relation between* O *and* A*. The elements of* O *are called objects or transactions, while the elements of* A *are called items or attributes.*

For example, Figure 3.2 represents a formal context (O, A, R). $O = \{1, 2, 3, 4, 5, 6, 7, 8\}$ is the set of objects, and $A = \{a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8\}$ is the set of items. The crosses in the table describe the relation R of O and A.

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
1	×	×					×	
2	×	×					×	×
3	×	×	×				×	×
4	×		×				×	×
5	×	×		×		×		
6	×	×	×	×		×		
7	×		×	×	×			
8	×		×	×		×		

Figure 3.2: An example of formal context

A formal context is usually represented by the binary data, but in practice, the values of attribute are not binary, we can transform many-valued formal context to binary values formal context by concept scaling [15].

Definition 3.2.2 *Two* **closure operators** *are defined as $O_1 \rightarrow O_1''$ for set O and $A_1 \rightarrow A_1''$ for set A.*

$$O_1' := \{a \in A \mid oRa \textit{ for all } o \in O_1\}$$

$$A_1' := \{o \in O \mid oRa \textit{ for all } a \in A_1\}$$

These two operators are called the **Galois connection** for (O, A, R). These operators are used to determine a formal concept.

Definition 3.2.3 *A* **formal concept** *of (O, A, R) is a pair (O_1, A_1) with $O_1 \subseteq O$, $A_1 \subseteq A, O_1 = A_1'$ and $A_1 = O_1'$. O_1 is called extent, A_1 is called intent.*

For example, (68, $a_1a_3a_4a_6$) is a formal concept of the formal context of Figure 3.2. $a_1a_3a_4a_6$ is intent of (68, $a_1a_3a_4a_6$), and 68 is extent of (68, $a_1a_3a_4a_6$).

Definition 3.2.4 *We say that there is a hierarchical order between two formal concepts $(O_1$, $A_1)$ and $(O_2$, $A_2)$, if $O_1 \subseteq O_2$ (or $A_2 \subseteq A_1$). And $(O_1$, $A_1)$ is called the* **sub-concept** *of*

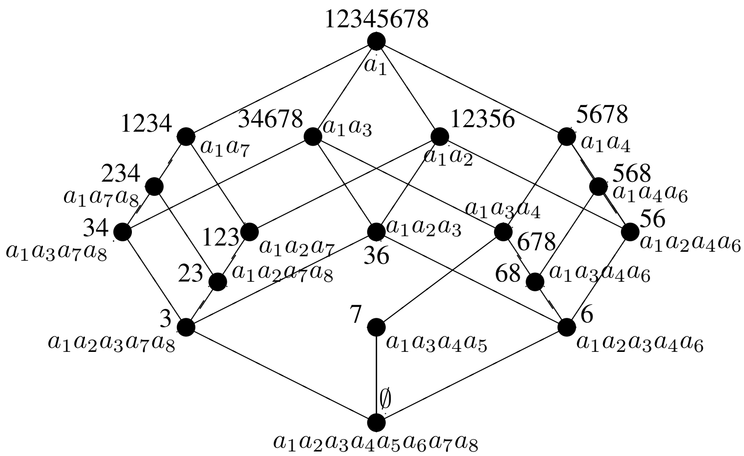

Figure 3.3: An example of concept lattice

(O_2, A_2), or (O_2, A_2) is called the **super-concept** *of (O_1, A_1), if there is no concept (O_3, A_3), $A_2 \subseteq A_3 \subseteq A_1$ or $O_1 \subseteq O_3 \subseteq O_2$.*

All formal concepts with the hierarchical order of concepts form a complete lattice called **concept lattice***.*

For example, all formal concepts and the concept lattices on the formal context (see Figure 3.2) are shown in Figure 3.3.

Definition 3.2.5 *An itemset $C \subseteq A$ is a* **closed itemset** *iff $C'' = C$.*

Thus, a closed itemset is intent of a formal concept. This definition is very important for closed itemset algorithm.

A formal concept or closed itemset describes more stricter relation between objects and attributes than itemset of association rules mining. For itemset, one attribute set maps to an object set, it's injective; but for formal concept, there is a bijection between one attribute set and one object set. The intent of a concept is a closed itemset and it's a maximal itemset.

For example, $\{a_1,a_7\}$ is a closed itemset of the data context of Figure 3.2.

Definition 3.2.6 *If C_1 and C_2 are closed itemsets, $C_1 \subseteq C_2$, then we say that there is a hierarchical order between C_1 and C_2.*

All closed itemsets with the hierarchical order of closed itemsets form of a complete lattice called **closed itemset lattice**.

3.3 New strategy: Cluster analysis and Association analysis for the same data

Both cluster analysis and association analysis are important tasks of data mining. In some applications, we need both cluster analysis and association analysis for the same data. Each task takes very high time cost to deal with large data. In order to reduce expensive cost of the two mining tasks for large data set of transactions, we propose one strategy to unify cluster analysis and association analysis.

3.3.1 Introduction

Both cluster analysis [21] and association analysis [1] are important tasks of data mining. In recent years, cluster analysis and association analysis have attracted a lot of attention among the fields of research and applications. Cluster analysis and association analysis play an important role in data mining applications such as text mining, Web mining, information retrieval and biomedical informatics, and many others. A variety of techniques and approaches of cluster analysis and association analysis have been developed and successfully applied to real-life data mining problems. However, due to large amounts of data continue to grow inexorably in size and complexity, the techniques and approaches of cluster analysis and association analysis suffer from the challenges such as very large data, high-dimensional data, distributed heterogeneous data, and complex data, etc. In some applications, we need both cluster analysis and association analysis for the same data. Each task takes very high time cost to deal with large

data. Although cluster analysis and association analysis are separated tasks for research and applications, in order to reduce the expensive cost of data mining tasks, we propose to unify the cluster analysis and association analysis for mining the database of transactions. This is the key motivation to unify cluster analysis and association analysis. Furthermore, we can unify cluster analysis and association analysis for database of transactions due to the following reasons:

1) Both of them analyze the relationship between the elements of data set. In fact, the two tasks extract the same essential relationship: similarity. Only the description and bounds of the relationship are different. This is the precondition to unify cluster analysis and association analysis.

   Cluster analysis is a process of identification and categorization of subsets of objects, and a division of similar objects of a data set into different homogeneous subsets or clusters of the data set. The objects in each cluster share some common trait that are similarity or proximity according to some defined distance measure. The degree of association is strong between objects of the same cluster and weak between objects of different clusters.

   Association analysis is used to discover items that tend to appear together frequently within a data set, and to extract rules, such as implication or correlation, which relate co-occurring items. The main task and the key problem of association analysis is to extract the the frequent itemsets (i.e. the sets of items that frequently appear together). Frequent patterns mining has been the focus of great interest for data mining research and applications. Frequent patterns play an essential role in many data mining tasks that try to find interesting patterns from databases, such as association rules, correlations, sequences, classifiers, clusters, etc.

   In fact, cluster analysis and association analysis can mine the clusters or frequent patterns for items or transactions respectively. Frequent pattern reveals one kind of similarity between elements of data. Cluster analysis may reveal associations and relationships in data that may contribute to mining the models or rules from data. So the elements in a frequent pattern are similar, and the similar elements may have the same frequency.

2) Mining closed sets can be an essential step for cluster analysis and association analysis. Some existing works show we can extract the clusters and frequent patterns from closed sets [6, 32]. Cluster analysis and association analysis may share the closed sets for mining the same data set. So we need not to extract closed sets separately for cluster analysis and association analysis.

3) Closed sets mining provides a solution to interpret the clusters and frequent patterns.

   For the most of techniques and approaches of cluster analysis and association analysis, it's hard to interpret the mining results. For example, it's hard to interpret the clusters and frequent pattern produced with existing mining techniques. It's also hard to give the signification of the distance measure in most of clustering methods.

   Closed sets is derived from formal concept analysis (FCA). The formal concept can help us to interpret the closed sets. Closed sets mining facilitates pattern interpretation. In human thinking and life, the objects are clustered by concepts and attributes, and we can interpret attribute patterns and object patterns with concepts. So the concept-based methods can be used for the interpretation of the clusters and frequent patterns.

The idea of unifying cluster analysis and association analysis focuses on the database of transactions. The main framework of the idea is:

- Generating the data context with the description of items or transactions from the database of transactions

- Mining closed sets and the lattice of closed sets of database of transactions with FCA

- In each closed sets, adding extended information such as support, similarity, and interpretation, etc. We propose a new structure of each node of lattice. The node contains attribute set, object set, the number of objects, support and similarity description.

- Generating the clusters and closed frequent patterns with the interpretation

3.3.2 Framework of unifying cluster analysis and association analysis

In this section, we propose a framework of unifying cluster analysis and association analysis (see Figure 3.4).

From the database of transactions, we can generate data context that should be described by the items and transactions. And then an efficient algorithm should be applied to generate formal concepts. When the formal concepts are produced, some extended information should be extracted with formal concepts, according to the need of the mining task, to form extended concepts. Extended concepts can contain intent, extent, support and similarity description. Knowledge lattice can be generated with extended concepts. Finally, closed frequent patterns and clusters can be produced from the same knowledge lattice or extended concepts.

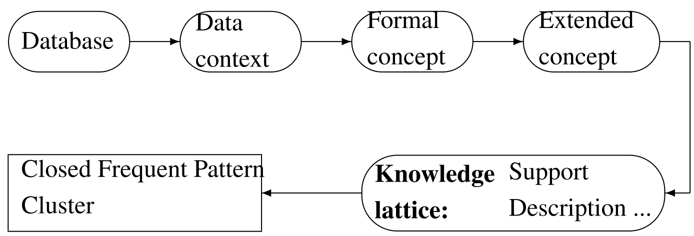

Figure 3.4: Framework of unifying cluster analysis and association analysis

Data context is the base of the mining task. Data context need to have understandable description for each item and transaction. Sometimes we need to reduce, transpose or order the data context. For example, when data have high dimension, especially the the size of object set is smaller than the size of item set, we can transpose the data context to generate formal concepts for mining high-dimensional data. Analyzing the most of lattice algorithms, we find that one algorithm can focuss on items or transactions of data context. The performances of an algorithm can be different according to the number of items and transactions.

In this framework, the generation of formal concepts and knowledge lattice is the essential step. The key of the applications is the performance of the algorithm of generation of the formal concepts or closed itemsets. Several algorithms were proposed to generate concepts or concept lattices on a formal context, for example: Bordat [5], Ganter (NextClosure algorithm) [14], Chein [8], Norris [29], Godin [16] and Nourine [30], etc.

Chapter 4

New algorithms of distributed knowledge sharing for digital ecosystem

Contents

We have developed new algorithms for extracting knowledge in digital ecosystems. Some ideas are presented here.

The purpose of developing these algorithms is to provide a tool to extract knowledge from large data. These algorithms can be applied for practical knowledge sharing such as OKS in OPAALS.

4.1 Scalable algorithm for association analysis

4.1.1 Analysis of the search space

In order to discuss the decomposition of the search space, we give the following definition.

Definition 4.1.1 *Given an attribute $a_i \in A$ of the context (O, A, R), a set E, $a_i \notin E$. We define $a_i \otimes E = \{\{a_i\} \cup X$ for all $X \subseteq E\}$.*

$$\begin{aligned} A_k &= \{a_k\} \cup \left(\bigcup_{\forall X_i \in \cup A_j} \{\{a_k\} \cup X_i\} \right) \\ &= a_k \otimes \{a_{k+1}, a_{k+2}, \cdots, a_m\} \quad k+1 \leq j \leq m \end{aligned}$$

We can decompose the search space into many partitions such as A_m, A_{m-1}, A_{m-2}, A_{m-3} or combination of some of them. In each partition we can look for the closed itemsets independently. But the problem is whether each partition contains closed itemsets. In order to ensure it is not empty for each partition, we need to order the formal context.

Definition 4.1.2 *A formal context is called* **ordered data context** *if we order the items of formal context by number of objects of each item from the smallest to the biggest one, and the items with the same objects are merged as one item. We note ordered data context $(O, A^{\triangleleft}, R)$ of the formal context (O, A, R).*

Definition 4.1.3 *An item a_i of a formal context (O, A, R), all subsets of $\{a_i, a_{i+1}, \ldots, a_{m-1}, a_m\}$ that include a_i, form a search sub-space (for closed itemset) that is called* **folding search sub-space (F3S)** *of a_i, denoted $F3S_i$.*

Summing up the analysis of the search space of closed itemsets, we can order the formal context as ordered data context, the search space of closed itemsets is: $F3S_m \cup F3S_{m-1} \cup F3S_{m-2} \cup F3S_{m-3} \cup \cdots \cup F3S_i \cdots \cup F3S_1 \cup \emptyset$, and then decompose the search space into some partitions. We can generate closed itemsets in each partition.

4.1.2 The new algorithm

Definition 4.1.4 *Given an itemset* $A_1 \subset A$*,* $A_1 = \{b_1, b_2, \ldots, b_i, \ldots, b_k\}$*,* $b_i \in A$*.* A_1 *is an infrequent itemset.* **The candidate of next closed itemset** *of* A_1*, noted* $C'^{\bowtie}_{A_1}$*, is* $A_1 \uplus a_i = (A_1 \cap (a_1, a_2, \ldots, a_{i-1}) \cup \{a_i\})''$*, where* $a_i < b_k$ *and* $a_i \notin A_1$*,* a_i *is the biggest one of* A *with* $A_1 < A_1 \uplus a_i$ *following the order:* $a_1 < \ldots < a_i < \ldots < a_m$*.*

We propose a new algorithm that can be used to generate closed itemsets or frequent closed itemsets. The principle of the algorithm is presented by following steps:

- Decompose the search space into some partitions
  - Convert (O, A, R) to $(O, A^{\lhd}, R)$ where $A^{\lhd} = \{a_1^{\lhd}, a_2^{\lhd} \ldots, a_i^{\lhd}, \ldots, a_m^{\lhd}\}$
  - In order to balance the number of closed itemsets of partitions, some items of $A^{\lhd}$ are chosen to form an order set P
    1) $P = \{a_{P_T}^{\lhd}, a_{P_{T-1}}^{\lhd} \ldots, a_{P_k}^{\lhd}, \ldots, a_{P_1}^{\lhd}\}$, $|P| = T$, $a_{P_k}^{\lhd} \in A^{\lhd}$
    2) $a_{P_T}^{\lhd} < a_{P_{T-1}}^{\lhd} < \ldots < a_{P_k}^{\lhd} < \ldots < a_{P_2}^{\lhd} < a_{P_1}^{\lhd} = a_m^{\lhd}$
    3) A parameter DP is used to choose $a_{P_k}^{\lhd}$ $(0 < DP < 1)$, where $DP = \frac{|\{a_1^{\lhd}, \cdots, a_{P_k}^{\lhd}\}|}{|\{a_1^{\lhd}, \cdots, a_{P_{k-1}}^{\lhd}\}|}$
  - Get the partitions: $[a_{P_k}^{\lhd}, a_{P_{k+1}}^{\lhd})$ and $[a_{P_T}^{\lhd})$
    1) Interval $[a_{P_k}, a_{P_{k+1}})$ is the search space from item a_{P_k} to $a_{P_{k+1}}$
    2)

$$\begin{aligned} \left[a_{P_k}^{\lhd}, a_{P_{k+1}}^{\lhd}\right) &= \bigcup_{P_k \le i < P_{k+1} \le P_T} (F3S_i) \\ \left[a_{P_T}^{\lhd}\right) &= F3S_{P_T} \end{aligned}$$

- Generate next frequent closed itemset from an itemset A_1 for each partition

4.2 Frequent closed informative itemset mining

In this section, we present the principle of the novel algorithm FCII (Frequent Closed Informative Itemsets mining).

4.2.1 The principle of algorithm FCII

Definition 4.2.1 *Given a context (O, A, R), an item a_i is called* **maximal item***, if $a_i \in A$, for all $a_j \in A$, $i \neq j$ and $\{a_i\}' \not\subset \{a_j\}'$.*

From the definition 4.2.1, we can conclude the following proposition:

Proposition 4.2.1 *Given a context (O, A, R), if item $a_i \in A$ and a_i is a maximal item, and for all $a_j \in A$, $i \neq j$ and $\{a_i\}' \neq \{a_j\}'$, then $\{a_i\}$ is a closed itemset.*

Proof : For any $a_i \in A$, we have $\{a_i\} \subseteq \{a_i\}''$ and $\{a_i\}''$ is a closed itemset. a_i is a maximal item and for all $a_j \in A$, $i \neq j$, $\{a_i\}' \neq \{a_j\}'$, then we have $\{a_i\} = \{a_i\}''$. Thus, $\{a_i\}$ is a closed itemset.

Given a formal context and minimum support, we propose the algorithm to generate frequent closed itemsets by following steps:

1) First of all, the algorithm needs to generate the ordered frequent context.

Definition 4.2.2 *Given the minimum support m, a formal context is called* **ordered frequent context** *if we only choose the items which number of transactions is not less than m, and order these items of formal context by number of transactions of each item from the smallest to the biggest one, and the items with the same objects are merged as one item. We note ordered frequent context $(O, A^{\lhd}, R)_m$ of the formal context (O, A, R).*

One example of ordered frequent context is shown in figure 4.2.

By the proposition 4.2.1 the definition 4.2.2, we have:

Proposition 4.2.2 *Given an ordered frequent context* $(O, A^{\triangleleft}, R)_m$ *, if item* $a_i \in A^{\triangleleft}$ *and* a_i *is a maximal item, then* $\{a_i\}$ *is a frequent closed itemset.*

2) The second step: from the ordered frequent context, every maximal item forms a frequent closed itemset of the first level. When we get the closed itemset, we can add other information to the itemset to generate the frequent closed informative itemset.

Definition 4.2.3 *A closed itemset is called* **frequent closed informative itemset** *if the support of the itemset is not less than the minimum support and the itemset has other information such as the set of transactions and the support of the itemset.*

For example, the frequent closed informative itemset in first level for the formal context of figure 3.2 is a_1 with other information $S(8)C(12345678)$ which S(8) means the support is 8 and C(12345678) means the transaction set or cluster is $\{1, 2, 3, 4, 5, 6, 7, 8\}$.

In fact, the closed itemset and its transaction set form one formal concept. We can use the formal concept, frequency of itemset and related domain knowledge to interpret the mining results.

3) The third step: from above level, if the support of the frequent closed informative itemset is equal to m, this itemset ends to generate itemsets of next level because the itemsets of next level will not be frequent. Otherwise we will deal with the projection context (see definition 4.2.4) of this itemset by step 1 to step 3. And then from the ordered frequent context of the projection context, we can generate all frequent closed informative itemsets of this level.

Definition 4.2.4 *Given an item* a_i *and a formal context* (O, A, R)*,* $(\{a_i\}',\ A{-}\{a_i\},\ R)$ *is called* **projection context** *of* a_i*.*

4) At the end, we can get all frequent closed informative itemsets. For example, given the minimum support 2, the algorithm generates all frequent closed informative itemsets (see Figure 4.1) from the formal context of Figure 3.2.

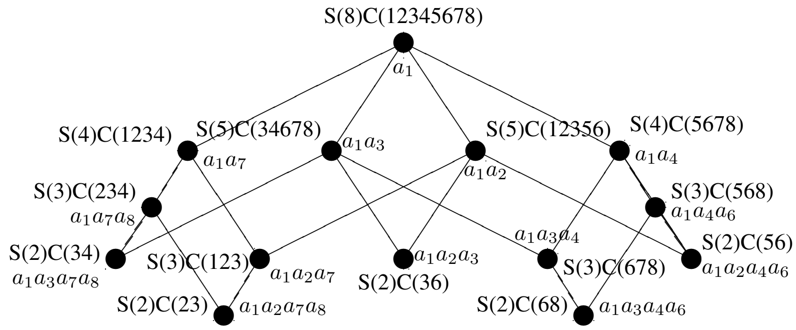

Figure 4.1: An example: frequent closed informative itemset mining with algorithm FCII. (minimum support is 2; for each node, S means support, C means cluster)

4.2.2 An example

For the formal context of Figure 3.2, given the minimum support = 2, we show how to use the algorithm to generate all frequent closed informative itemsets.

For the first step, the algorithm generates the ordered frequent context (see Figure 4.2) based on minimum support 2. a_5 is not included in the ordered frequent context because it's support is less than 2.

	a_8	a_6	a_7	a_4	a_3	a_2	a_1
1			×			×	×
2	×		×			×	×
3	×		×		×	×	×
4	×		×		×		×
5		×		×		×	×
6		×		×	×	×	×
7				×	×		×
8		×		×	×		×

Figure 4.2: Ordered frequent context of the formal context of Figure 3.2

From the ordered frequent context, only a_1 is not included by other items. Hence, $\{a_1\}$ forms a frequent closed itemset of the first level and $\{a_1 : S(8)C(12345678)\}$ is the frequent

closed informative itemset.

The support of a_1 is bigger than 2, so we consider the ordered frequent context of the projection context of a_1 (see Figure 4.3). By the same way, we can generate all frequent closed informative itemsets of this level: $\{a_1a_7 : S(4)C(1234)\}$, $\{a_1a_3 : S(5)C(34678)\}$, $\{a_1a_2 : S(5)C(12356)\}$, $\{a_1a_4 : S(4)C(5678)\}$, as a_7, a_3, a_2 and a_4 are not included by other items in the ordered frequent context of Figure 4.3.

	a_8	a_6	a_7	a_4	a_3	a_2
1			×			×
2	×		×			×
3	×		×		×	×
4	×		×		×	
5		×		×		×
6		×		×	×	×
7				×	×	
8		×		×	×	

Figure 4.3: Ordered frequent context of the projection context of a_1

If the support of the frequent closed informative itemset is 2, this itemset ends to generate itemsets of next level. Otherwise for the next level, we use the same approach to generate all frequent closed informative itemsets (see Figure 4.1).

4.2.3 Extension of the new algorithm for clustering analysis

The frequent closed informative itemset includes information of the node such as formal concept (the closed itemsets and the set of transactions) and the support of this frequent closed itemset. This feature can facilitate interpretation of mining results.

Conceptual clustering [12, 25, 4] can seek clusters by concept structures so that the concept can describe the clusters. One approach of conceptual clustering is based on concept lattice [7]. The algorithm FCII can be extended to generate conceptual overlapping clusters for cluster analysis. We can give a minimum support based on the number of clusters and then generate

some frequent closed informative itemsets to generate conceptual overlapping clusters. This is ongoing work.

Chapter 5

Conclusion and Further work

Contents

5.1 Conclusion

This deliverable presented the following works:

- Development of a framework for collaborative knowledge sharing between participants of the ecosystem

  We propose the framework for collaborative knowledge sharing that includes Infrastructure, Sharing platforms, Sharing requirement, Knowledge resource, Knowledge engine and Knowledge Sharing. Knowledge engine is for acquirement of knowledge that responds to where and how to find knowledge. Knowledge sharing is for sharing knowledge in digital ecosystem. It focuses on the answer to how to share knowledge.

- Present novel tools such as Data Mining and KDD, and Formal Concept Analysis for collaborative knowledge sharing.

- Development of new algorithms of distributed knowledge sharing for digital ecosystem

  We are developing new algorithms for extracting knowledge in digital ecosystems. The algorithm is based on formal concept lattice. Theoretical foundation of concept lattice founds on the mathematical lattice theory. Lattice is a popular mathematical structure for modeling conceptual hierarchies. Concept lattice is a method for deriving conceptual structures out of data. From concept lattice, we can study the relations between objects and attributes in a formal context, and how objects can be hierarchically grouped together according to their common attributes. Certain object subset and the set of their common attributes can represent each other, such duality sets form a formal concept, which the attribute subset is called intent and the object subset is called extent. Among the formal concepts, there exists an order relation, they form a complete lattice: concept lattice. Each node in the lattice is a concept and the corresponding graph (Hasse diagram) that is considered as the generalization and specialization relationships between concepts. Such graphical structure represents directly and visually the relations of conceptual hierarchies. It allows us to analyze and mine the complex data for such as classification, association rules mining, clustering, etc. Furthermore, concept lattice also provides an effective tool of knowledge visualization.

5.2 Further work

One main aim of the OPAALS is to develop an integrated theoretical foundation for Digital Ecosystems research spanning three widely different disciplinary domains: social science, computer science, and natural science. Our proposed tools, data mining and FCA, have wide applications for social science, computer science, and natural science. We will develop a common models and platform of knowledge acquirement, knowledge sharing and knowledge management for social science, computer science, and natural science.

The techniques of data mining and FCA have many potential applications for collaborative knowledge sharing of the ecosystem. We will develop suitable models and algorithms to analyze, extract, acquire, deliver and share knowledge of the ecosystem. This development can be integrated with other WPs in Phase 2 of the project such as WP5 (Integration with the Dig-

ital Ecosystem platform) and WP10 (Sustainable Research Community Building in the Open Knowledge).

Ongoing research works will also include development of other efficient techniques and approaches for the framework for collaborative knowledge sharing.

Bibliography

[1] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A.I. Verkamo. *Advances in Knowledge discovery and Data Mining*, chapter Fast discovery of association rules, pages 307–328. AAAI/MIT Press, 1996.

[2] H. J. Ahn, H. J. Lee, K. Cho, and S. J. Park. Utilizing knowledge context in virtual collaborative work. *Decis. Support Syst.*, 39(4):563–582, 2005.

[3] G. Birkhoff. *Lattice Theory*. American Mathematical Society, Providence, RI, 3rd edition, 1967.

[4] G. Biswas, J. B. Weinberg, and Douglas H. Fisher. Iterate: A conceptual clustering algorithm for data mining. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, (28):100C111, 1998.

[5] J. Bordat. Calcul pratique du treillis de galois d'une correspondance. *Mathématiques, Informatiques et Sciences Humaines*, 24(94):31–47, 1986.

[6] C. Carpineto and G. Romano. Galois: An order theoretic approach to conceptual clustering. In *Proc. of the Machine Learning conf.*, pages 33–40, 1993.

[7] C. Carpineto and G. Romano. Galois: An order-theoretic approach to conceptual clustering. In *Proceedings of ICML'93*, pages 33–40, Amherst, Juillet 1993.

[8] M. Chein. Algorithme de recherche des sous-matrices premières d'une matrice. *Bull. Math. R.S. Roumanie*, 13 (61)(1):21–25, 1969.

[9] K. Dalkir. *Knowledge management in theory and practice*. Butterworth-Heinemann, 2005.

[10] V. Dignum, F. Dignum, and J.J. Meyer. An agent-mediated approach to the support of knowledge sharing in organizations. *The Knowledge Engineering Review*, 19(02):147–174, 2005.

[11] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors. *Advances in Knowledge Discovery and Data Mining*. MII Press, Mento Park, 1996.

[12] D. H. Fisher. Knowledge acquisition via incremental conceptual clustering. *Machine Learning*, (2):139–172, 1987.

[13] G. Fulop, L. Hernadi, J. Kormos, and Z. C. Kovacs. The role of document classification structure in knowledge sharing: Improving knowledge sharing within the organization by creating sample document repositories. In *ECKM*, pages 206–212, 2005.

[14] B. Ganter. Two basic algorithms in concept analysis. Technical report, Technische Hochschule Darmstadt, 1984.

[15] B. Ganter and R. Wille. *Formal Concept Analysis. Mathematical Foundations*. Springer, 1999.

[16] R. Godin, R. Missaoui, and H. Alaoui. Incremental concept formation algorithms based on Galois (concept) lattices. *Computational Intelligence*, 11(2):246–267, 1995.

[17] Thomas R. Gruber and Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5:199–220, 1993.

[18] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, 2001.

[19] M. Handzic and Angela Lee. Knowledge sharing via technology. In *ECKM*, pages 221–226, 2005.

[20] M. T. Hansen. Knowledge networks: Explaining effective knowledge sharing in multiunit companies. *Organization Science*, 13(3):232–248, 2002.

[21] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323, 1999.

[22] A. Jorge. Remote collaborative data mining through online knowledge sharing. In *PRO-VE '02: Proceedings of the IFIP TC5/WG5.5 Third Working Conference on Infrastructures for Virtual Enterprises*, page 646, Deventer, The Netherlands, The Netherlands, 2002. Kluwer, B.V.

[23] B. Kogut. The network as knowledge: generative rules and the emergence of structure. *Strategic Management Journal*, 21(3):405–425, 2000.

[24] B. Kogut and U. Zander. Knowledge of the firm, combinative capabilities, and the replication of technology. *Organization Science*, 3(3):383–397, 1992.

[25] M. Lebowitz. Experiments with incremental concept formation: Unimem. *Machine Learning*, (2):103–138, 1987.

[26] P. J. Ludford, R. Priedhorsky, K. Reily, and L. Terveen. Capturing, sharing, and using local place information. In *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 1235–1244, New York, NY, USA, 2007. ACM.

[27] E. Mephu Nguifo, M. Liquiere, and V. Duquenne. *JETAI Special Issue on Concept Lattice for KDD*. Taylor and Francis, 2002.

[28] I. Nonaka. A dynamic theory of organizational knowledge creation. *Organization Science*, 5(1):14–37, 1994.

[29] E.M. Norris. An algorithm for computing the maximal rectangles in a binary relation. *Revue Roumaine de Mathématiques Pures et Appliquées*, 23(2):243–250, 1978.

[30] L. Nourine and O. Raynaud. A fast algorithm for building lattices. *Information Processing Letters*, 71:199–204, 1999.

[31] D. E. O'Leary. Knowledge acquisition from multiple experts: An empirical study. *Management Science*, 44(8):1049–1058, 1998.

[32] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Efficient mining of association rules using closed itemsets lattices. *Journal of Information Systems*, 24(1):25–46, 1999.

[33] J. Pei, J. Han, and R. Mao. CLOSET: An efficient algorithm for mining frequent closed itemsets. In *ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, pages 21–30, 2000.

[34] Gerd Stumme and Alexander Maedche. Fca-merge: Bottom-up merging of ontologies. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence*, pages 225–230, 2001.

[35] C. Y. Wah, B. Loh, T. Menkhoff, and H.-D. Evers. Theorizing, measuring, and predicting knowledge sharing behavior in organizations - a social capital approach. In *HICSS*. IEEE Computer Society, 2005.

[36] Y. Wang and J. Vassileva. Trust-based community formation in peer-to-peer file sharing networks. In *WI '04: Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence*, pages 341–348, Washington, DC, USA, 2004. IEEE Computer Society.

[37] M. J. Zaki and C. Hsiao. CHARM: An efficient algorithm for closed itemset mining. Technical Report 99-10, Rensselaer Polytechnic Institute, 1999.