มาตรวัดความเหมือนแนวความคิดแบบใช้ความถี่

A FREQUENCY-BASED CONCEPT SIMILARITY MEASURE

จิราภรณ์ ทัดรัตน์

JIRAPOND TADRAT

วิทยานิพนธ์นี้สำหรับการศึกษาตามหลักสูตรปริญญาปรัชญาดุษฎีบัณฑิต

สาขาวิชาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2554

KMITL – 2011 – SC – D – 002 - 026

A FREQUENCY-BASED CONCEPT SIMILARITY MEASURE

JIRAPOND TADRAT

A THESIS SUBMITTED IN FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY IN COMPUTER SCIENCE

FACULTY OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2011

KMITL-2011-SC-D-002-026

หัวข้อวิทยานิพนธ์	มาตรวัดความเหมือนแนวความคิดแบบใช้ความถี่
นักศึกษา	นางสาวจิราภรณ์ ทัดรัตน์
รหัสประจำตัว	49062904
ปริญญา	ปริญญาดุษฎีบัณฑิต
สาขาวิชา	วิทยาการคอมพิวเตอร์
พ.ศ.	2554
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รศ. ดร.วีระ บุญจริง

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาโครงสร้างการจัดเก็บความรู้ของการอ้างเหตุผลด้วยกรณี โดยนำวิธีการวิเคราะห์แนวคิดแบบฟอร์มัลของข้อมูลมาประยุกต์ใช้ สำหรับการจัดเก็บคลังความรู้ วิธีนี้เป็นการวิเคราะห์ข้อมูลให้อยู่ในรูปแบบลำดับชั้น ที่สามารถสนับสนุนการแก้ปัญหาของการอ้างเหตุผลด้วยกรณี อย่างไรก็ตามโครงสร้างที่พัฒนานี้ต้องการวิธีวัดความคล้ายที่จำเพาะเพื่อใช้สำหรับการแก้ปัญหาของกรณีใหม่ ดังนั้นงานวิจัยนี้ได้นำเสนอวิธีวัดความคล้ายระหว่างกรณีใหม่และกรณีที่เคยได้รับการแก้ปัญหาที่ถูกเก็บไว้ในคลังความรู้ เพื่อหากรณีที่ผ่านมาที่มีปัญหาใกล้เคียงกับกรณีใหม่มากที่สุด ในการประเมินประสิทธิภาพของวิธีที่นำเสนอในงานนี้ ได้มีการพัฒนาระบบการจำแนกกลุ่มโดยใช้วิธีที่นำเสนอและนำข้อมูลมาตรฐานจากยูซีไอเพื่อทดสอบความถูกต้องของการจำแนกข้อมูล และเปรียบเทียบกับอื่นๆ ผลลัพธ์แสดงให้เห็นว่าวิธีที่นำเสนอในงานวิจัยนี้สามารถปรับปรุงความถูกต้องของการจำแนกข้อมูลได้ดีกว่าวิธีอื่นๆ

Thesis Title	A Frequency-Based Concept Similarity Measure
Student	Miss Jirapond Tadrat
Student ID.	49062904
Degree	Doctor of Philosophy
Programme	Computer Science
Year	2011
Thesis Advisor	Assoc. Prof. Dr. Veera Boonjing

ABSTRACT

The purpose of this research is to develop a knowledge base structure for case-based reasoning using formal concept analysis. The new hierarchical knowledge structure supports both case based classification and case based problem solving. However, the new structure requires a specific similarity metric for solving a new problem. Therefore, the research proposes a new frequency-based similarity metric to retrieve the most applicable case from knowledge base. Experiments on the UCI data sets show that the new similarity metric significantly gives higher classification accuracy than other similarity metrics.

ACKNOWLEDGEMENTS

The author would like to thank my family for giving opportunities, pushing me, understanding me, and helping me to achieve my Ph.D. Also, a warm thank to Sophon Muangprathub for encouraging and fulfilling financial support during the entire period of study.

I would like to express my deeply many thanks to my thesis advisor, Assoc. Prof. Dr. Veera Boonjing, whose all advises and very good support from the initial to the final level enabled me to develop an understanding of the research.

I am heartily thankful to my thesis co-advisor, Asst. Prof. Dr. Puntip Pattaraintakorn, whose encouragement, guidance and inspiration for happily learning research methods. I would like to thank her for enduring everything to me, trying to understand me, and helping me all aspects.

The authors also gratefully acknowledge the helpful comments and suggestions of the committee, which have improved the presentation.

I would like to thank Office of Academic Administration of King Mongkut's Institute of Technology Ladkrabang, Prince of Songkla University, Thai Higher Education Commission and National Centre of Excellence in Mathematics for supporting my Ph.D. study.

I would like to thank my friends, Nakorn Sakeaw, for helping, supporting and encouraging all aspects e.g., about programming, stationery. Thanks also to Asst. Prof. Dr. Laor Boongasame for encouraging and supporting me. Lastly I would like to thank my lovely friends, Dr. Doungrat Chitcharoen, Dr. Arthit Intarasit, Jarunee Saelee, Prajak Jongkraijak, and others for encouraging and supporting me.

Jirapond Tadrat

TABLE OF CONTENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figures **Page**

CHAPTER 1

INTRODUCTION

1.1 Statements of Problem

A case-based reasoning (CBR) [2, 3, 18, 27] is a method to problem solving that learns from prior experiences. The tasks of CBR system are often divided into; classification and problem-solving CBR [18]. Classification CBR uses previous cases as reference points for new problem. In contrast, problem-solving CBR uses previous cases to suggest the most applicable solutions to new situation. Both tasks store a set of pairs' problem descriptions and solution in their knowledge base for reusing in the future. Traditional CBR consists of four steps [2, 18] as follows: retrieve the most similar cases, reuse existing knowledge of previous cases to solve new problem, revise suggested solutions and retain useful parts of this experience for future problem solving as shown Fig. 1.1.

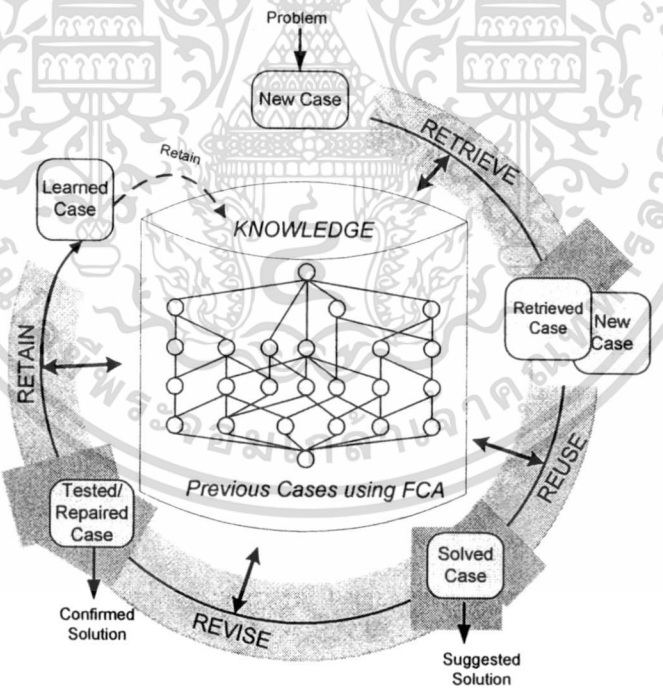


Figure 1.1 System overview of CBR process (adapted from [2])

The structure of knowledge base that directly supports four steps above will make a great effect on efficiency and performance of CBR. Formal concept analysis (FCA) can elicit knowledge embedded in previous cases to solve new problems. FCA is especially well-suited to

support CBR system when problem at hand involving hierarchical structure [7]. In addition, implication drawn from FCA can suggest solutions from dependency inside knowledge base [25]. Thus, we apply FCA to build a knowledge base for CBR. Nevertheless, the knowledge base obtained from FCA technique, called concept lattice, requires a specific retrieval process to solve new problem. Mostly, traditional similarity measures of FCA are based on set theory and consider two formal concepts as binary weighting (e.g., [4], [20], [31]). However, weights determined from all contents should be considered to enhance precision and recall of the retrieved formal concept. Hence, we propose a new similarity measure to serve this objective.

1.2 Research Objectives

The main objective of this thesis is to develop a better knowledge base by using FCA and propose its new similarity measure based on appearance frequency of formal concepts for CBR. The first goal of the thesis is illustrated by implementing in a part of CBR system for both classification and problem-solving tasks. The second goal is demonstrated by retrieving cases in classification CBR task to show more accuracy classification.

1.3 Scope of Thesis

The scopes of the study are:

1. The proposed system employs a concept lattice in FCA as a knowledge structure. A concept lattice is discussed to be useful in knowledge construction both classification CBR and problem-solving CBR.

2. A case based classification is implemented for demonstrating classification accuracy by using the UCI benchmark data sets.

3. The proposed retrieval method uses a new concept similarity measure based on appearance frequency of formal concepts.

4. The proposed similarity measure is compared with other similarity measures based on FCA.

1.4 Results

The benefits of the research are as follows:

1. A better knowledge base by using formal concept analysis is proposed to support CBR. Experimental results are also included to demonstrate the applicability of this construction for both classification and problem-solving in CBR.

2. Dependency induced from our concept lattice knowledge base can help to suggest informative solutions for problem-solving CBR.

3. Concept lattice knowledge base provides more accuracy classification for hierarchical data structure when comparing with non-hierarchical data structure.

4. The proposed similarity measure is introduced. Experiment results on standard data sets show that the new similarity measure gives classification accuracy better than existing similarity measures.

1.5 Research Methodology

A part of classification CBR system based on concept lattice knowledge base in this research is implemented. Afterwards, the proposed similarity measure and other similarity measures are used to retrieve previous experience in concept lattice for solving new problem. Several benchmark data sets from the UCI repository are used to evaluate classification accuracy of implemented system. In addition, the proposed similarity measure is compared with other existing similarity measures to show our retrieval performance.

1.6 Organization of Thesis

The remaining chapters of this thesis are organized in the following way.

Chapter 2 provides background of CBR and FCA. In addition, related works are reviewed. They are classified into two groups: (1) knowledge representation and (2) FCA and similarity measures in CBR. In particular, current research based on the proposed methods is discussed.

Chapter 3 presents FCA knowledge base supporting CBR system with a frequency-based concept similarity measure.

Chapter 4 contains experimental evaluation. This chapter shows experiment results and their analysis.

Chapter 5 presents conclusion and recommendation.

CHAPTER 2

LITERATURE REVIEWS

This chapter provides background of CBR and FCA as presented in Section 2.1. Related works are reviewed for knowledge representation is Section 2.2. In Section 2.3, we briefly review case based classification system. Section 2.4 presents FCA and similarity measures in CBR. In particular, current research based on the proposed methods is discussed.

2.1 Backgrounds

This section addresses background of CBR and an overview of its methodology. In the mean time, a motivation of building a better CBR is pointed. In addition, theoretical background of FCA is described. These definitions and theories are applied in this research to enhance overall performances of the system.

2.1.1 Case-Based Reasoning

A case-based reasoning (CBR) [2, 3, 18, 27] is a method to problem solving that learns from prior experiences. A single case represents specific knowledge tie to a context. Several cases are stored in the case base. The case base (after several learning experiences) will be constructed as the knowledge base. A CBR system uses this knowledge base for the future problem solving. Traditional CBR processes as illustrated in Fig. 1.1 are (i) retrieving from previous cases, (ii) reusing the information in that case, (iii) revising the solution and (iv) retaining a new experience into the knowledge base [2, 18].

From Fig. 1.1, an initial description of a problem defines a new case. Retrieval process is the first step that matches a (partial) problem description (or new case) with previous cases for finding the most similar case. An efficient retrieval method is a method that retrieves the best matching case without calculating every similarity between a new problem and each case in knowledge base. Next, reusing and revising process will combine retrieved case and new problem to present solution. Finally, a new experience is retained each time a problem has been solved, making it immediately available for future problems. Each new experience is gradually increased

into knowledge base. It is possible that redundancy of cases may affect to effective recall. Thus, construction of knowledge base is a key success for a CBR system.

CBR is a recent approach to problem solving and learning that has got a lot of attention over the last few years. The tasks of CBR system are often divided into; interpretive CBR (or classification CBR) and problem-solving CBR [3, 18, 27]. Classification CBR uses previous cases as reference points for new situation. In contrast, problem-solving CBR uses previous cases to only suggest the most applicable solutions to new situation. Both tasks store cases that are the properties as well as problem descriptions and its solution in knowledge base for reusing in the future.

For this thesis, classification CBR task is referred to as a case based classification. Case based classification is a method classifying an unlabeled case (new problem) by retrieving the most similar case and reusing its class label. It can solve new problem while knowledge base is gradually increased by nature. In contrast, some classification techniques (e.g., rule-based classifier) hardly extend or refine their process during problem solving stage. Traditionally, the incremental knowledge base of case based classification is based on record-cases that face to time- and space-consuming for both construction and problem solving stages. Moreover, the sequentially matching between new problem and record-cases results in only one case. This single solution requires solution fix and causes more difficulties in the reusing step. Therefore, a suitable solution to these problems is a well-organized knowledge structure together with incremental nature. Towards this goal, FCA will be employed to build a knowledge base in concept lattice format described in the next section.

2.1.2 Formal Concept Analysis

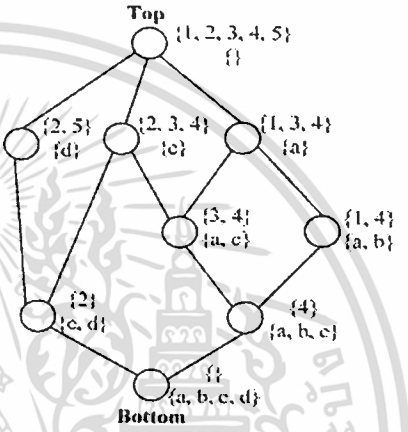
Formal concept analysis (FCA) [15, 16, 42], invented by Rudolf Wille, is not only a method for data analysis and knowledge representation, but also a formal formulation for concept formation and learning. It is widely used for information science to describe natural attributes of information representation in hierarchical structure model [26]. It provides relationship of generalization and specialization among concepts through concept lattice [7]. Practically, FCA starts with a formal context which contains values 0 or 1 in an information system. Below, we introduce basic definitions and idea of FCA taken from [16].

Definition 1. *A formal context* K:= (*G, M, I*) *consists of two sets G and M and a relation I between G and M. The elements of G are called the objects and the elements of M are called the attributes of the context. In order to express that an object g is in a relation I with an attribute m, we write gIm or* $(g,m) \in I$ *and read it as "the object g has the attribute m".*

Another form of formal context is called a *cross table.* It can be used to identify groups of cases with share attributes in binary relation format. Fig. 2.1(a) shows an example of a formal context in a cross table form.

	a	b	c	d
1	1	1	0	0
2	0	0	1	1
3	1	0	1	0
4	1	1	1	0
5	0	0	0	1

(a) Formal context



(b) Concept lattice

Figure 2.1 Knowledge representations from formal context into concept lattice

CBR system in this thesis, an object is represented with a case. Attributes are referred to as sets of problem descriptions and solutions. A formal context is considered as a case base. This research introduces definition of an association between object and its attributes as follows.

Definition 2. *For a set* $A \subseteq G$ *of objects,* A' *is defined as follows*

$$A' := \{m \in M \mid gIm \text{ for all } g \in A\}.$$

Correspondingly, for a set $B \subseteq M$ *of attributes,* B' *is defined as follows*

$$B' := \{g \in G \mid gIm \text{ for all } m \in B\}.$$

Definition 3. *A formal concept of the formal context* (*G, M, I*) *is a pair* (*A, B*) *with* $A \subseteq G$, $B \subseteq M$, $A' = B$ *and* $B' = A$. *We call A the extent and B the intent of the formal concept* (*A, B*). $\mathfrak{B}(G, M, I)$ *denotes the set of all formal concepts of the formal context* (*G, M, I*).

Definition 4. *If* (A_1,B_1) *and* (A_2,B_2) *are formal concepts of formal context* K:= (*G, M, I*), (A_1,B_1) *is called a subconcept of* (A_2,B_2) *(or* (A_2,B_2) *is a superconcept of* (A_1,B_1)*), provide that* $A_1 \subseteq A_2$ *(or* $B_2 \subseteq B_1$ *) and is denoted by* $(A_1,B_1) \leq (A_2,B_2)$. *The relation* $\leq$ *is called the hierarchical order*

(or simply order) of the formal concepts. The set of all formal concepts of (G, M, I) ordered in this way denoted by $\underline{\mathfrak{B}}(G, M, I)$ and is called the concept lattice of the formal context (G, M, I).

The line diagram in Fig. 2.1(b) shows an example of the concept lattice for the formal context in Fig. 2.1(a). It demonstrates 9 concepts. To extract knowledge for solving new problem, The Basic Theorem on Concept Lattices (see more detail in [16]) is used. This theorem provides implications between attributes that are used to identify solution in this work. It provides the group of cases identified by attributes.

Theorem 1. *(The Basic Theorem on Concept Lattice) Let T be an index set and, for every $t \in T$. The concept lattice $\underline{\mathfrak{B}}(G, M, I)$ is a complete lattice in which infimum and supremum are given by:*

$$\bigwedge_{t\in T}(A_t, B_t) = \left(\bigcap_{t\in T} A_t, \left(\bigcup_{t\in T} B_t\right)''\right),$$

$$\bigvee_{t\in T}(A_t, B_t) = \left(\left(\bigcup_{t\in T} A_t\right)'', \bigcap_{t\in T} B_t\right).$$

A complete lattice V is isomorphic to $\underline{\mathfrak{B}}(G, M, I)$, denoted by $V \cong \underline{\mathfrak{B}}(G, M, I)$, if and only if there are mappings $\tilde{\gamma}: G \longrightarrow V$ and $\tilde{\mu}: M \longrightarrow V$ such that $\tilde{\gamma}(G)$ is supremum-dense in V, $\tilde{\mu}(M)$ is infimum-dense in V and gIm is equivalent to $\tilde{\gamma}g \leq \tilde{\mu}m$ for all $g \in G$ and all $m \in M$.

The mappings $\tilde{\gamma}g$ and $\tilde{\mu}m$ in Theorem 1 indicate how formal context can be identified in the concept lattice. This is elaborated by the following definition.

Definition 5. *For an object $g \in G$ we write g' instead of $\{g\}$ for the object intent $\{m \in M \mid gIm\}$ of the object g. Correspondingly, $m := \{g \in G \mid gIm\}$ is the attribute extent of the attribute m. Retaining the symbols used in Theorem 1, we write γg for the object concept (g, g) and μm for the attribute concept (m, m).*

FCA is begun with formal context which contains only 0 and 1. However, in real-world data there are not only two values (0 or 1) in a database. This is called many-valued context (Definition 6).

Definition 6. *A many-valued context* (G, M, W, I) *consists of set of* G, M *and* W *and a ternary relation* I *between* G, M *and* W *(i.e.,* $I \subseteq G \times M \times W$*) for which it holds that*

$$(g, m, w) \in I \text{ and } (g, m, v) \in I \text{ always imply } w = v.$$

The elements of G are called objects, those of M (many-valued) attributes and those of Wattribute values. If W has n elements, it is called n-value context. We read $(g, m, w) \in I$ as "the attribute m has the value w" for the object g. We write $m(g) = w$ as $(g, m, w) \in I$. In order to obtain a concept lattice from a many-valued context, there is to be transformed into formal context. The transformation process is described by *conceptual scales*.

Definition 7. *A scale for the attribute* m *of many-valued context is a (one-valued) context* $S_M := (G_m, M_m, I_m)$ *with* $m(G) \subseteq G_m$. *The objects of a scale are called scale values, the attributes are called scale attributes.*

The scales of each context are joined to make one-value context (formal context), which the simplest method is called plain scaling. In plain scaling, the derived formal context is obtained from many-valued context (G, M, W, I) and the scale contexts S_m, $m \in M$ where the attribute set of S_m is replaced by $M_m := m \times M_m$. Thus, the new formal context (G, N, J) can derive from many-valued context with respect to plain scaling with

$$N := \bigcup_{m \in M} M_m,$$

and $gJ(m, n) :\Leftrightarrow m(g) = w$ and $wI_m n$.

In this thesis a many-valued is transformed from context into formal context form, called conceptual scaling, for supporting FCA by using scale. In CBR, an implication between any two attributes measures dependency by considering problem descriptions and solutions as subconcepts and superconcepts, respectively. Let C and D be sets of problem descriptions and solution where $C, D \subseteq M$, and $C \cap D = \emptyset$. An implication among attributes in M where $M = C \cup D$, is a pair of subsets of M, denoted by $C \rightarrow D$.

Proposition 1. *An implication* $C \rightarrow D$ *holds in* (G, M, I) *if and only if* $D \subseteq C$. *It then automatically holds in the set of all concept intents as well.*

An implication $C \rightarrow D$ hold in (G, M, I) if and only if $C \rightarrow m$ holds for each $m \in D$. $C \rightarrow m$ hold if and only if $(m, m) \geq (C, C)$, i.e., if $\mu m \geq \wedge \{\mu n | n \in C\}$. This means that we have to check in a concept lattice whether the concept denoted by m is located above the infimum of all formal concepts denoted by n from C.

2.2 Knowledge Representation

Traditional approaches of knowledge representation in CBR can be classified into many categories such as feature vector representations, structured representations, textual representations, and so on that are surveyed in [10, 12, 27]. In general, these knowledge representations require particular retrieval methods. For this reason, we review both knowledge representation and its retrieval process. Both of them are usually regarded as the most important features in the CBR cycle. They directly influence the validity and efficiency of CBR systems [12, 18]. However, the proposed structure of knowledge base in this research involves feature vector representations and structured representations. Thus, we only focus on both knowledge representations in this section.

The first structure, feature vector representation, represents a case as attribute-value pairs, collected in data table form or called exemplars. Similarity measure approaches in this structure are based on distance concept between previous cases and new problem. This distance concept generally gives similarity value in an interval [0, 1]. Usually the retrieved cases are k most similar to the new problem, referred to *k-nearest neighbor* (k-NN). Thus, the similarity measures for this structure will support the retrieval of the k cases that are maximally similar and relevant to new problem. These similarity approaches are the basic technique to usefully apply for retrieval in other structures [17, 21, 30, 32, 36]. For example in [6, 22, 34], the authors used a similarity measure to assess relevant case and to collect the same group of problem descriptions. Thus, when their systems retrieve more similar case to new problem, the relevant case will be retrieved. Although, feature vector representation can retrieve k relevant cases, the inclusion of structured representation will lead to obtain more useful knowledge.

Structure representation is another popular method to build case base structure in CBR system. There are many approaches such as graph representation [8, 29], object-oriented representation [3, 24, 39], and hierarchical representation [32]. More specifically, concept lattice is one type of structure representation that will be described in the next subsection. For graph

representation, the author used attribute graph to represent previous cases in course timetabling problem domain [29]. Each case consists of node and edge where node represents course labels, constrains of course and the number of period, and edge represents constrains of relation between courses. Unfortunately, retrieved case leads to contradict of courses. This drawback may be due to undirected graph. Next, Champin et al. [8] proposed similarity measure of directed graph. It measures the similarity of two labeled graphs to identify their common features. Their proposed model is a flexible model for knowledge similarity. It provides qualitative similarity information which can be useful for further adaptation.

To capture both case and general knowledge, [3] proposed an object-oriented and frame-based representation system. The author used similarity assessment in two-steps process. The first step used similarity measure to retrieve a set of potentially similar cases. Next step, this set of case was added to general domain knowledge and generated explanations for feature-to-feature matches. In [24, 39], the authors used the concept of generalized cases based on an object-oriented representation. The generalized cases cover a subspace of the problem-space. The similarity assessment is derived from dependencies of the closely-related problem. In [24], the authors formulated a similarity assessment for generalized cases described by continuous attributes as a nonlinear programming problem and then applied an optimization-based retrieval method. Tartakovski et al. [39] extended the case representation to support mixed, discrete, and continuous attributes. They also formulated similarity assessment as a special case of a mixed integer nonlinear optimization problem, and proposed an optimization-based retrieval method operating on a given index structure.

For hierarchical representation, the problem descriptions are decomposed into subproblems in [33]. A set of subproblems is separately solved and then recombined to produce a suitable solution. An advantage of this representation is that it allows a whole case or its parts to be accessed and used by the case-based reasoner, and the constraints can be used to guide adaptation.

For another way of representation (e.g., logic representation, textual representation), Meng et al. [23] represented cases with universal logic relation between attributes. They used factor-structure connection and λ-similarity to compare previous case and new problem. These similarity approaches are suitable for qualitative or quantitative attributes. In [41], this paper surveyed textual CBR that involved extraction of free text in each case to keywords automatically. Next, these keywords are built into textual representation structure.

2.3 Case Based Classification

Classification is one of an important task in the field of data mining due to its wide application. The goal of classification is to build a concise model of the distribution of class labels in term of predictor attributes, called *classifier*. The resulting classifier is then used to assign class labels to the testing instances where the values of the predictor features are known, but the value of the class label is unknown. A large number of techniques have been developed rapidly to improve the accuracy of classification surveyed in [19]. Case based classification is one technique to classify a new problem by retrieving the most similar case from a knowledge base and reusing its class label. Mostly, traditional case-based classifier systems were developed based on similarity function to retrieve prior cases. The goal of similarity-based classifier is to retrieve previous cases to solve new problem. Jurisica et al. [17] proposed a case-based classifier system called *TA3*. They used context-based similarity to retrieve relevant cases and then used them for the classification task. Salamo et al. [30] developed a case based classifier system called BASTIAN by using rough sets for weighting method and attribute reduction in the case base.

Case-based classification by FCA was also carried out. Belen et al. [7] used FCA as a complementary technique to enrich the domain taxonomy which provided facilitation and suggestion the solution in new problem. In [38], the authors proposed a framework to construct knowledge base in CBR system based on rough sets and FCA. Recently, they improved such framework by using fuzzy sets [37]. The result is that the usage of fuzzy sets supports to build the knowledge structure in FCA technique successfully. These frameworks were implemented and applied successfully to achieve the rules from knowledge structure of FCA in [25, 35, 36]. The advantages are alleviate overfitting problem, reduce cases and elicit attribute dependency in knowledge base.

2.4 FCA and Similarity Measures

FCA is successfully applied in several CBR systems [6, 25, 36]. Belen et al. [6] utilized FCA for problem solving CBR system. They report that FCA provided good facilitation and suggestion of solutions for new problem. In [36, 38], the authors proposed knowledge base construction in a CBR system using FCA. Nevertheless, its similarity measure in [36, 38] is computed separately by employing vector model idea. It will be more efficient to compute

directly from FCA knowledge base. Thus, this thesis fulfills this gap by using a new similarity measure invented for concept lattice structure.

Several researchers developed similarity measures to retrieve formal concept as surveyed in [3, 10, 13, 14, 20, 31]. Lengnink [20] defined similarity measures to find similar and relevant concepts: local similarity and global similarity as follows. For any two concepts (A, B) and (C, D) in a formal concept, local similarity measure, s_l, and global similarity measure, s_g, respectively are defined as

$$s_l\big((A,B),(C,D)\big) = \frac{1}{2}\left(\frac{|A\cap C|}{|A\cup C|} + \frac{|B\cap D|}{|B\cup D|}\right), \tag{2.1}$$

$$s_g\big((A,B),(C,D)\big) = \frac{1}{2}\left(\frac{|A\cap C|}{|G|} + \frac{|B\cap D|}{|M|}\right). \tag{2.2}$$

Saquer et al. [31] proposed a similarity measure for concept approximation by using local similarity. It is simple and can approximate extensions when there are only problem descriptions. To define a scope of retrieved results in semantic web, Dau et al. [10] developed a new combined local and global similarity measure obtained from user. Formica [13,14] proposed an adapted version of (2.1) for semantic web with weight of formal concept specified by user. In fact, this weight should be determined from the contents of data.

In addition, Alqadah et al. [3] improved existing similarity measures based on set theory which is described below. For any two sets of intension in formal concepts x and y, Jaccard index (s_{Jac}), Sorenesen coefficient (s_{Sor}) and Symmetric difference (s_{Xor}) are defined as

$$s_{Jac}(x,y) = \frac{|x\cap y|}{|x\cup y|}, \tag{2.3}$$

$$s_{Sor}(x,y) = \frac{2*|x\cap y|}{|x|+|y|}, \tag{2.4}$$

$$s_{Xor}(x,y) = 1 - \frac{|(x\setminus y)\cup(y\setminus x)|}{|x\cup y|}. \tag{2.5}$$

From the above studies, weights are selected based on user's requirements or by matching user's query and previous cases in binary relation form. Alternatively, we should determine weight directly from data. Thus, we specifically propose a new similarity measure based on vector model that consider problem descriptions and solution with in a concept lattice.

These exiting similarity measures will be used to compare with our frequency-based concept similarity measure for experiment in chapter 4. We demonstrate retrieval process from concept lattice for classification CBR using above similarity measures. Let we given concept lattice knowledge base shown in Fig. 2.2 described more detail in section 3.2.

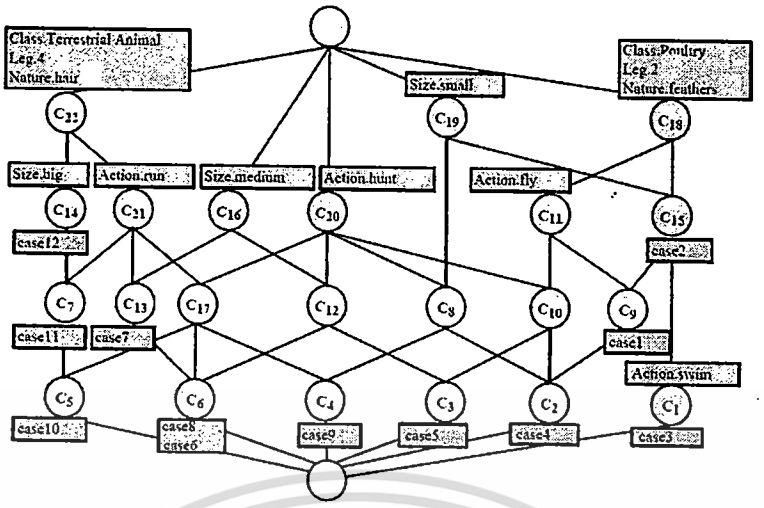


Figure 2.2 Concept lattice knowledge base for case-based classification

We begin with using local and global similarity measures as equations (2.1) and (2.2), respectively. We use previous input new problem i.e.,

B = {Size.big, Leg.4, Nature.smooth, Action.run, Action.swim, Action.hunt}.

We can show calculation of input new problem and concept lattice in Fig. 2.2 with Table 2.1. To classify the new problem, we obtain similarity between two formal concepts, (A, B) and (C, D), where A is extent in concept lattice, C is extent of new problem obtained approximation with B i.e., C ={case10}, D is intent in concept lattice, $|G|$=12, and $|M|$= 13.

Table 2.1 Concept similarity measure using local and global similarity measures

No.	$\|A \cap C\|$	$\|A \cup C\|$	$\|B \cap D\|$	$\|B \cup D\|$	$S_l = \frac{1}{2}(\frac{\|A \cap C\|}{\|A \cup C\|} + \frac{\|B \cap D\|}{\|B \cup D\|})$	$S_g = \frac{1}{2}(\frac{\|A \cap C\|}{\|G\|} + \frac{\|B \cap D\|}{\|M\|})$
C_{22}	1	7	1	8	0.134	0.080
C_{21}	1	6	2	8	0.208	0.119
C_{20}	1	6	1	6	0.167	0.080
C_{19}	0	5	0	7	0.000	0.000
C_{18}	0	5	0	10	0.000	0.000
C_{17}	1	4	3	8	0.313	0.157
C_{16}	0	4	0	7	0.000	0.000
C_{15}	0	4	0	10	0.000	0.000
C_{14}	1	3	2	8	0.292	0.119
C_{13}	0	3	1	10	0.050	0.038
C_{12}	0	3	1	7	0.071	0.038
C_{11}	0	3	0	10	0.000	0.000
C_{10}	0	2	1	10	0.050	0.038
C_9	0	2	1	10	0.050	0.038

Table 2.1 *(Continue)*

No.	$\|A \cap C\|$	$\|A \cup C\|$	$\|B \cap D\|$	$\|B \cup D\|$	$S_l = \frac{1}{2}(\frac{\|A \cap C\|}{\|A \cup C\|} + \frac{\|B \cap D\|}{\|B \cup D\|})$	$S_g = \frac{1}{2}(\frac{\|A \cap C\|}{\|G\|} + \frac{\|B \cap D\|}{\|M\|})$
C_8	0	2	3	5	0.300	0.115
C_7	1	2	3	8	0.438	0.157
C_6	0	2	3	9	0.167	0.115
C_5	1	1	4	8	**0.750**	**0.196**
C_4	0	1	3	9	0.167	0.115
C_3	0	1	1	11	0.045	0.038
C_2	0	1	1	11	0.045	0.038
C_1	0	1	1	10	0.050	0.038

From Table 2.1, a formal concept C_5 in Fig 2.2 will be retrieved with using both local and global similarity measures. Thus, the solution of input new problem is Terrestrial Animal class.

In addition, we use Jaccard index (s_{Jac}), Sorenesen coefficient (s_{Sor}) and Symmetric difference (s_{Xor}), mentioned in equation (2.3)-(2.5), to compare with our approach. We illustrate an example for case based classification using three methods. From concept lattice knowledge base in Fig. 2.2, it is used to be case base for solving the new problem, where x is input new problem and y is intent in concept lattice. We obtain similarity calculation between new problem and previous cases shown in Table 2.2.

Table 2.2 Concept similarity measure using Jaccard index, Sorenesen coefficient, and Symmetric difference similarity measures

No.	$\|x \cap y\|$	$\|x \cup y\|$	$\|(x \backslash y) \cup (y \backslash x)\|$	$S_{Jac} = \frac{\|x \cap y\|}{\|x \cup y\|}$	$S_{Sor} = \frac{2 \cdot \|x \cap y\|}{\|x\| + \|y\|}$	$S_{Xor} = 1 - \frac{\|(x \backslash y) \cup (y \backslash x)\|}{\|x \cup y\|}$
C_{22}	1	8	7	0.125	0.222	0.125
C_{21}	2	8	6	0.250	0.400	0.250
C_{20}	1	6	5	0.167	0.286	0.167
C_{19}	0	7	7	0.000	0.000	0.000
C_{18}	0	10	9	0.000	0.000	0.100
C_{17}	3	8	5	0.375	0.545	0.375
C_{16}	0	7	7	0.000	0.000	0.000
C_{15}	0	10	10	0.000	0.000	0.000
C_{14}	2	8	6	0.250	0.400	0.250
C_{13}	1	10	7	0.100	0.182	0.300
C_{12}	1	7	6	0.143	0.250	0.143
C_{11}	0	10	10	0.000	0.000	0.000
C_{10}	1	10	9	0.100	0.182	0.100
C_9	1	10	10	0.100	0.182	0.000
C_8	3	5	5	**0.600**	**0.750**	0.000

Table 2.2 *(Continue)*

No.	$\|x \cap y\|$	$\|x \cup y\|$	$\|(x\backslash y) \cup (y\backslash x)\|$	$S_{Jac} = \frac{\|x \cap y\|}{\|x \cup y\|}$	$S_{Sor} = \frac{2 * \|x \cap y\|}{\|x\| + \|y\|}$	$S_{Xor} = 1 - \frac{\|(x\backslash y) \cup (y\backslash x)\|}{\|x \cup y\|}$
C_8	3	5	5	**0.600**	**0.750**	0.000
C_7	3	8	5	0.375	0.545	0.375
C_6	3	9	6	0.333	0.500	0.333
C_5	4	8	4	0.500	0.667	**0.500**
C_4	3	9	6	0.333	0.500	0.333
C_3	1	11	9	0.091	0.167	0.182
C_2	1	11	10	0.091	0.167	0.091
C_1	1	10	9	0.100	0.182	0.100

From Table 2.2, a formal concept C_8 in Fig 2.2 will be retrieved with using Jaccard index (s_{Jac}) and Sorenesen coefficient (s_{Sor}) similarity measures. The solution of input new problem is {case4, case9}. These solutions lead to conflict of class label. Thus, these similarity measures are weakness point. For Symmetric difference (s_{Xor}), a formal concept C_5 in Fig 2.2 will be retrieved. Thus, the solution of input new problem is Terrestrial Animal class.

CHAPTER 3

A FREQUENCY-BASED CONCEPT SIMILARITY MEASURE

FCA, a well known structure is called a concept lattice, can be used to build an appropriate structure for knowledge base. It can represent previous cases in formal concepts form contained redundant information. But this is very useful since one can choose which of both parts of the concept should be used in a given situation [16]. Moreover, FCA can support solving processes of case based classification. Namely, it can promote to enhance efficient retrieval method because each formal concept can identify set of sharing cases as same problem descriptions without calculating every case. This structure requires similarity measure especially. Mostly, traditional similarity measures of FCA are based on set theory and consider two formal concepts as binary weighting (e.g., [3], [20], [31]). However, an effective similarity measure should take into account both local weighting and global weighting achieved from considering all formal concepts.

For reusing and revising step, FCA provide draw both explicit and implicit knowledge. Explicit knowledge can describe both problems with solution (attributes) and cases (objects) of information represented in the hierarchical structure model (e.g., [8], [26], [42]). Implicit knowledge can elicit knowledge embedded of previous cases with its implication property (e.g., [6], [36], [40]). These properties are used to acquire solution of new problem in our system. In final step, FCA has incremental structure to facilitate dynamic knowledge base.

3.1 A Frequency-Based Concept Similarity Measure

Case retrieval in concept lattice can be done by two distinct ways: lattice traversal and similarity measure. Our target is to use the latter due to its accuracy and timely manners. A new concept similarity measure is based on appearance frequency of formal concepts [35]. To invent a new concept similarity measure, we exploit an idea of vector space model which is a classical model of information retrieval [28] as described below.

Let C_P be a formal concept of formal context (G, M, I) represents a pair (E_P, I_P) of previous cases. Let $E_P \subseteq G$, $I_P \subseteq M$ where E_P is a set of previous cases that have similar problem description(s) and solution while I_P comprises of all problem descriptions and solution shared by all those cases. A new problem is defined as $C_N := (E_N, I_N)$, where E_N is a set of retrieved cases to

achieve a solution, and I_N is a set of new problem descriptions provided by user. Initially E_N is calculated from C_P. We have that $E_N = E_P$ if its pair I_P gives $\max|I_N \cap I_P|$. Thus, a new concept similarity measure between a formal concept and new problem is defined as $Sim(C_N, C_P)$. The closer the value of $Sim(C_N, C_P)$ is to 1, the greater the similarity of C_N and C_P.

Definition 8 *Given a formal concept of previous case $C_P = (E_P, I_P)$ and a formal concept of new problem $C_N = (E_N, I_N)$ in a formal context (G, M, I), concept similarity measure is defined as*

$$Sim(C_P, C_N) = \frac{1}{2}\left(\frac{\sum_{u \in I_N \cap I_P} \left(log \frac{N}{Fa_u}\right)^2}{\left[\sum_{i \in I_N}\left(log \frac{N}{Fa_i}\right)^2 \sum_{j \in I_P}\left(log \frac{N}{Fa_j}\right)^2\right]^{1/2}} + \frac{\sum_{v \in E_N \cap E_P} \left(log \frac{N}{Fc_v}\right)^2}{\left[\sum_{k \in E_N}\left(log \frac{N}{Fc_k}\right)^2 \sum_{l \in E_P}\left(log \frac{N}{Fa_l}\right)^2\right]^{1/2}} \right)$$

where N is a total number of formal concepts, Fa_u, Fa_i and Fa_j are a frequency of attributes u, i and j, respectively, $\{u, i, j\} \in M$, and Fc_v, Fc_k and Fc_l are a frequency of cases v, k and l, respectively, $\{v, k, l\} \in G$.

Theorem 2 *$Sim(C_N, C_P)$ is said to be the degree of similarity between formal concept C_P and formal concept C_N in concept lattice $\mathfrak{B}(G, M, I)$ if $Sim(C_N, C_P)$ satisfies the following conditions* [11] :

(1) $0 \le Sim(C_N, C_P) \le 1$

(2) $Sim(C_N, C_P) = 1$ *if* $C_P = C_N$

(3) $Sim(C_N, C_P) = Sim(C_P, C_N)$

(4) $Sim(C_N, C_O) \le Sim(C_P, C_N)$ and $Sim(C_N, C_O) \le Sim(C_P, C_O)$ *if* $C_N \subseteq C_P \subseteq C_O$, $C_O \in \mathfrak{B}(G, M, I)$.

Proof. From Definition 8, $C_P = (E_P, I_P)$, $C_N = (E_N, I_N)$ we give $C_P \cap C_N = (E_P \cap E_N, I_P \cap I_N)$. To prove that

(1) $0 \le Sim(C_N, C_P) \le 1$. This condition is considered into three cases as follows:

1) $C_N = C_P$. ($Sim(C_P, C_N) = 1$)

2) $C_N \cap C_P = \varnothing$.($Sim(C_P, C_N) = 0$)

3) $C_N \cap C_P \neq \varnothing$. ($Sim(C_P, C_N)$ =(0,1])

Case 1: $C_N = C_P \rightarrow$[($E_P = E_N$) and ($I_P = I_N$)]. Namely, $|E_P| = |E_N| = p$ and $|I_P| = |I_N| = q$. Thus, from Definition 8, we have

$$Sim(C_P, C_N) = \frac{1}{2}\left(\frac{\left(\log\frac{N}{Fa_1}\right)^2 + \cdots + \left(\log\frac{N}{Fa_q}\right)^2}{\left[\left(\left(\log\frac{N}{Fa_1}\right)^2 + \cdots + \left(\log\frac{N}{Fa_q}\right)^2\right)\left(\left(\log\frac{N}{Fa_1}\right)^2 + \cdots + \left(\log\frac{N}{Fa_q}\right)^2\right)\right]^{1/2}} + \frac{\left(\log\frac{N}{Fc_1}\right)^2 + \cdots + \left(\log\frac{N}{Fc_p}\right)^2}{\left[\left(\left(\log\frac{N}{Fc_1}\right)^2 + \cdots + \left(\log\frac{N}{Fc_p}\right)^2\right)\left(\left(\log\frac{N}{Fc_1}\right)^2 + \cdots + \left(\log\frac{N}{Fc_p}\right)^2\right)\right]^{1/2}}\right)$$

$$= \frac{1}{2}\left(\frac{\left(\log\frac{N}{Fa_1}\right)^2 + \cdots + \left(\log\frac{N}{Fa_q}\right)^2}{\left[\left(\left(\log\frac{N}{Fa_1}\right)^2 + \cdots + \left(\log\frac{N}{Fa_q}\right)^2\right)^2\right]^{1/2}} + \frac{\left(\log\frac{N}{Fc_1}\right)^2 + \cdots + \left(\log\frac{N}{Fc_p}\right)^2}{\left[\left(\left(\log\frac{N}{Fc_1}\right)^2 + \cdots + \left(\log\frac{N}{Fc_p}\right)^2\right)^2\right]^{1/2}}\right)$$

$$= \frac{1}{2}(1+1) = 1.$$

Case 2: $C_N \cap C_P = \varnothing \rightarrow [\, E_P \cap E_N = \varnothing$ and $I_P \cap I_N = \varnothing]$. Thus, $|\, E_P \cap E_N \,| = 0$, $|\, I_P \cap I_N \,| = 0$. Thus, from Definition 8, we have

$$\mathrm{Sim}(C_P, C_N) = \frac{1}{2}(0+0) = 0.$$

Case 3: $C_N \cap C_P \neq \varnothing \rightarrow [\, E_P \cap E_N \neq \varnothing$ or $I_P \cap I_N \neq \varnothing\,]$. For this case, we give $|\, E_P \cap E_N \,| = r$ and $|\, I_P \cap I_N \,| = s$. From Case 1, $|E_P| = |E_N| = p$ and $|I_P| = |I_N| = q$ then $r \leq p$ and $s \leq q$. Thus, from Definition 8, we have

$$Sim(C_P, C_N) = \frac{1}{2}\left(\frac{\left(\log\frac{N}{Fa_1}\right)^2 + \cdots + \left(\log\frac{N}{Fa_s}\right)^2}{\left[\left(\left(\log\frac{N}{Fa_1}\right)^2 + \cdots + \left(\log\frac{N}{Fa_q}\right)^2\right)\left(\left(\log\frac{N}{Fa_1}\right)^2 + \cdots + \left(\log\frac{N}{Fa_q}\right)^2\right)\right]^{1/2}} + \frac{\left(\log\frac{N}{Fc_1}\right)^2 + \cdots + \left(\log\frac{N}{Fc_r}\right)^2}{\left[\left(\left(\log\frac{N}{Fc_1}\right)^2 + \cdots + \left(\log\frac{N}{Fc_p}\right)^2\right)\left(\left(\log\frac{N}{Fc_1}\right)^2 + \cdots + \left(\log\frac{N}{Fc_p}\right)^2\right)\right]^{1/2}}\right)$$

$$= \frac{1}{2}\left(\frac{\left(\log\frac{N}{Fa_1}\right)^2 + \cdots + \left(\log\frac{N}{Fa_s}\right)^2}{\left[\left(\left(\log\frac{N}{Fa_1}\right)^2 + \cdots + \left(\log\frac{N}{Fa_q}\right)^2\right)^2\right]^{1/2}} + \frac{\left(\log\frac{N}{Fc_1}\right)^2 + \cdots + \left(\log\frac{N}{Fc_r}\right)^2}{\left[\left(\left(\log\frac{N}{Fc_1}\right)^2 + \cdots + \left(\log\frac{N}{Fc_p}\right)^2\right)^2\right]^{1/2}}\right)$$

$$= \frac{1}{2}\left(\frac{\left(\log\frac{N}{Fa_1}\right)^2 + \cdots + \left(\log\frac{N}{Fa_s}\right)^2}{\left(\log\frac{N}{Fa_1}\right)^2 + \cdots + \left(\log\frac{N}{Fa_q}\right)^2} + \frac{\left(\log\frac{N}{Fc_1}\right)^2 + \cdots + \left(\log\frac{N}{Fc_r}\right)^2}{\left(\log\frac{N}{Fc_1}\right)^2 + \cdots + \left(\log\frac{N}{Fc_p}\right)^2}\right).$$

From $r \leq p$ and $s \leq q$, we have $\frac{\left(\log\frac{N}{Fa_1}\right)^2 + \cdots + \left(\log\frac{N}{Fa_s}\right)^2}{\left(\log\frac{N}{Fa_1}\right)^2 + \cdots + \left(\log\frac{N}{Fa_q}\right)^2} \leq 1$ and $\frac{\left(\log\frac{N}{Fc_1}\right)^2 + \cdots + \left(\log\frac{N}{Fc_r}\right)^2}{\left(\log\frac{N}{Fc_1}\right)^2 + \cdots + \left(\log\frac{N}{Fc_p}\right)^2} \leq 1$. Thus,

$Sim(C_P, C_N) = \frac{1}{2}((0,1] + (0,1]) \leq 1$

In summary, the first condition certainly guarantees that $\mathbf{0 \leq Sim(C_N, C_P) \leq 1}$.

(2) $\boldsymbol{Sim(C_N, C_P) = 1}$ ***if*** $\boldsymbol{C_P = C_N}$. This condition can consider from Case 1 in the first condition. Hence, this condition guarantees that $C_P = C_N \rightarrow Sim(C_N, C_P) = 1$.

(3) $\boldsymbol{Sim(C_N, C_P) = Sim(C_P, C_N)}$. Originally, $C_P = (E_P, I_P)$, $C_N = (E_N, I_N)$, we can prove the first condition with three cases. In this condition, we assume to swap C_P and C_N with $C_N = (E_P, I_P)$, $C_P = (E_N, I_N)$ to prove that this condition is symmetry property in metrics. Obviously, set theory has switch property i.e. $E_N \cap E_P = E_P \cap E_N$ and $I_N \cap I_P = I_P \cap I_N$. In addition, in bottom part of our formula is multiply that has switch property. Thus, this condition is obvious symmetry property.

(4) $\boldsymbol{Sim(C_N, C_O) \leq Sim(C_P, C_N)}$ **and** $\boldsymbol{Sim(C_N, C_O) \leq Sim(C_P, C_O)}$ ***if*** $\boldsymbol{C_N \subseteq C_P \subseteq C_O}$. We give $C_N \subseteq C_P \subseteq C_O$, then $E_N \subseteq E_P \subseteq E_O$ or $I_N \subseteq I_P \subseteq I_O$. We have $|E_N| \leq |E_P| \leq |E_O|$ or $|I_N| \leq |I_P| \leq |I_O|$. To prove that

1) $Sim(C_N, C_O) \leq Sim(C_P, C_N)$ and

2) $Sim(C_N, C_O) \leq Sim(C_P, C_O)$

$$Sim(C_N, C_O) = \frac{1}{2}\left(\frac{\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|I_N\cap I_O|}}\right)^2}{\left[\left(\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|I_N|}}\right)^2\right)\left(\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|I_O|}}\right)^2\right)\right]^{1/2}} + \frac{\left(\log\frac{N}{Fc_1}\right)^2+\cdots+\left(\log\frac{N}{Fc_{|E_N\cap E_O|}}\right)^2}{\left[\left(\left(\log\frac{N}{Fc_1}\right)^2+\cdots+\left(\log\frac{N}{Fc_{|E_N|}}\right)^2\right)\left(\left(\log\frac{N}{Fc_1}\right)^2+\cdots+\left(\log\frac{N}{Fc_{|E_O|}}\right)^2\right)\right]^{1/2}}\right) \quad (1)$$

$$Sim(C_N, C_P) = \frac{1}{2}\left(\frac{\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|I_N\cap I_P|}}\right)^2}{\left[\left(\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|I_N|}}\right)^2\right)\left(\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|I_P|}}\right)^2\right)\right]^{1/2}} + \frac{\left(\log\frac{N}{Fc_1}\right)^2+\cdots+\left(\log\frac{N}{Fc_{|E_N\cap E_P|}}\right)^2}{\left[\left(\left(\log\frac{N}{Fc_1}\right)^2+\cdots+\left(\log\frac{N}{Fc_{|E_N|}}\right)^2\right)\left(\left(\log\frac{N}{Fc_1}\right)^2+\cdots+\left(\log\frac{N}{Fc_{|E_P|}}\right)^2\right)\right]^{1/2}}\right), \quad (2)$$

$$Sim(C_N, C_O) = \frac{1}{2}\left(\frac{\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|I_O\cap I_P|}}\right)^2}{\left[\left(\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|I_O|}}\right)^2\right)\left(\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|I_P|}}\right)^2\right)\right]^{1/2}} + \frac{\left(\log\frac{N}{Fc_1}\right)^2+\cdots+\left(\log\frac{N}{Fc_{|E_O\cap E_P|}}\right)^2}{\left[\left(\left(\log\frac{N}{Fc_1}\right)^2+\cdots+\left(\log\frac{N}{Fc_{|E_O|}}\right)^2\right)\left(\left(\log\frac{N}{Fc_1}\right)^2+\cdots+\left(\log\frac{N}{Fc_{|E_P|}}\right)^2\right)\right]^{1/2}}\right). \quad (3)$$

Shortly, we define

$A=\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|I_N|}}\right)^2$, $B=\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|I_P|}}\right)^2$, $C=\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|I_O|}}\right)^2$, *where* $A \leq B \leq C$.

$D=\left(\log\frac{N}{Fc_1}\right)^2+\cdots+\left(\log\frac{N}{Fc_{|E_N|}}\right)^2$, $E=\left(\log\frac{N}{Fc_1}\right)^2+\cdots+\left(\log\frac{N}{Fc_{|E_P|}}\right)^2$, $F=\left(\log\frac{N}{Fc_1}\right)^2+\cdots+\left(\log\frac{N}{Fc_{|E_O|}}\right)^2$, *where* $D \leq E \leq F$.

$u=\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|I_N\cap I_O|}}\right)^2$, $v=\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|I_N\cap I_P|}}\right)^2$, $w=\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|I_O\cap I_P|}}\right)^2$, *where* $v \leq u$, $v \leq w$.

$x=\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|E_N\cap E_O|}}\right)^2$, $y=\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|E_N\cap E_P|}}\right)^2$, $z=\left(\log\frac{N}{Fa_1}\right)^2+\cdots+\left(\log\frac{N}{Fa_{|E_O\cap E_P|}}\right)^2$, *where* $y \leq x$, $y \leq z$.

Case 1: $Sim(C_N, C_O) \leq Sim(C_P, C_N)$ or $Sim(C_N, C_O) - Sim(C_P, C_N) \leq 0$. From (1) and (2), we can prove that

$$\left(\frac{u}{\sqrt{AC}}+\frac{x}{\sqrt{DF}}\right)-\left(\frac{v}{\sqrt{AB}}+\frac{y}{\sqrt{DE}}\right) \leq 0$$

$$\left(\frac{u}{\sqrt{AC}}-\frac{v}{\sqrt{AB}}\right)+\left(\frac{x}{\sqrt{DF}}-\frac{y}{\sqrt{DE}}\right) \leq 0$$

Consider front part, $v \leq u$, $A \leq B \leq C$. Hence, the result of this part is negative integer. Similarly, the result of this part is negative integer because $y \leq x$, $D \leq E \leq F$. For overall result show that it is negative integer number. Thus, this case is certainly proved that $Sim(C_N, C_O) \leq Sim(C_P, C_N)$. In the meantime, Case 2 of this condition is considered as same as Case 1.

3.2 CBR Application

This section shows how to use the proposed methods (both knowledge base construction and a new similarity measure) in CBR tasks.

3.2.1 Classification CBR

In this subsection, we demonstrate retrieval process from concept lattice by using our proposed similarity measure for classification CBR. Let we are given a zoo data set (Table 3.1) where each case is described by problem descriptions (columns 2-5) and its solution (column 6).

This table involves classification of two types of animal: Poultry and Terrestrial Animal. We construct our knowledge base from Table 3.1 by using FCA as shown in Fig. 3.1. More detail of this concept lattice creation can be found in [25, 36, 38] given in Appendix.

Concept lattice in Fig. 3.1 consists of formal concept, C_1- C_{22}, intensions (upper labels) and extensions (lower labels). The number in ( ) symbol refers to a frequency of cases (attributes). To retrieve, we compute similarity value between new problem and each formal concept. For example, given a new unseen problem C_N with

I_N = {Size.big, Leg.4, Nature.smooth, Action.run, Action.swim, Action.hunt}.

Table 3.1 An example of zoo data set

Case	Size	Leg	Nature	Action	Class
Case1	small	2	feathers	fly	Poultry
Case2	small	2	feathers	-	Poultry
Case3	small	2	feathers	swim	Poultry
Case4	small	2	feathers	fly, hunt	Poultry
Case5	medium	2	feathers	fly, hunt	Poultry
Case6	medium	4	hair	hunt, run	Terrestrial Animal
Case7	medium	4	hair	run	Terrestrial Animal
Case8	medium	4	hair	hunt, run	Terrestrial Animal
Case9	medium	4	hair	hunt, run	Terrestrial Animal
Case10	big	4	hair	hunt, run	Terrestrial Animal
Case11	big	4	hair	run	Terrestrial Animal
Case12	big	4	hair	-	Terrestrial Animal

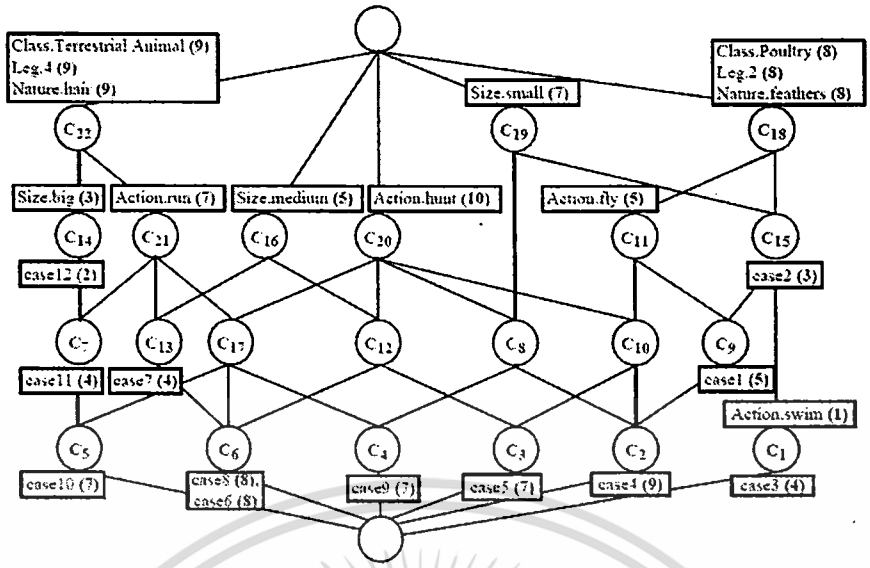


Figure 3.1 Our knowledge base in the format of concept lattice

Table 3.2 Concept similarity measure obtained from Fig. 3.1

No.	$\sum_u \left(\log\left(\frac{N}{Fa_u}\right)\right)^2$	$\sum_v \left(\log\left(\frac{N}{Fc_v}\right)\right)^2$	$\sum_j \left(\log\left(\frac{N}{Fa_j}\right)\right)^2$	$\sum_l \left(\log\left(\frac{N}{Fc_l}\right)\right)^2$	$Sim(C_P, C_N)$
C_{22}	0.151	0.247	0.452	3.061	0.206
C_{21}	0.398	0.247	0.699	1.977	0.313
C_{20}	0.117	0.247	0.247	1.279	0.287
C_{19}	0.000	0.000	0.247	2.110	0.000
C_{18}	0.000	0.000	0.579	2.110	0.000
C_{17}	0.515	0.247	0.817	0.881	0.428
C_{16}	0.000	0.000	0.414	1.181	0.000
C_{15}	0.000	0.000	0.826	1.862	0.000
C_{14}	0.899	0.247	1.201	1.880	0.416
C_{13}	0.398	0.000	1.113	0.934	0.108
C_{12}	0.000	0.000	0.531	0.988	0.000
C_{11}	0.000	0.000	0.993	0.812	0.000
C_{10}	0.117	0.000	1.110	0.398	0.032
C_9	0.000	0.000	1.240	0.565	0.000
C_8	0.117	0.000	0.365	0.398	0.055
C_7	1.147	0.247	1.448	0.795	0.551
C_6	0.515	0.000	1.231	0.386	0.133
C_5	1.264	0.247	1.565	0.247	**0.788**
C_4	0.515	0.000	1.064	0.247	0.143
C_3	0.117	0.000	1.524	0.247	0.027
C_2	0.117	0.000	1.358	0.151	0.029
C_1	1.802	0.000	2.628	0.548	0.317

The aim of this classification CBR is to classify C_N to either classes Poultry or Terrestrial Animal. By Definition 8, we obtain similarity between C_N and C_P as shown in Table 3.2, where $\sum_{i\in I_N}\left(\log\left(\frac{N}{Fa_i}\right)\right)^2 = \left(log\frac{22}{3}\right)^2 + \left(log\frac{22}{9}\right)^2 + \left(log\frac{22}{7}\right)^2 + \left(log\frac{22}{1}\right)^2 + \left(log\frac{22}{10}\right)^2 =$ 3.066, and $\sum_{k\in E_N}\left(\log\left(\frac{N}{Fc_k}\right)\right)^2 = \left(log\frac{22}{7}\right)^2 = 0.247$.

As one can see, a formal concept C_5 in Fig. 3.1 will be retrieved with $Sim(C_5, C_N)$ = 0.788, where E_5 is {case10}, and I_5 is {Size.big, Leg.4, Nature.hair, Action.hunt, Action.run, Class.Terrestrial Animal}. Thus, a solution of new unseen problem, C_N, is Terrestrial Animal class. This new unseen problem can be retained to solve new problems in CBR system in this work. Please note that, if problem descriptions of new problem are exactly as same as existing previous case, then similarity value is 1 ($Sim(C_P, C_N) = 1$) and classification accuracy is 100%.

3.2.2 Problem-Solving CBR

In this section, we illustrate problem-solving CBR by an example given in [6]. It describes travel agency domain where every case represents description of the journeys offered by a travel agency. Fig. 3.2 depicts a knowledge base in a concept lattice form of this data which contains 7 cases (see more details in [6]). The objective of this problem solving example is to suggest journey from user's query: *Skiing*.

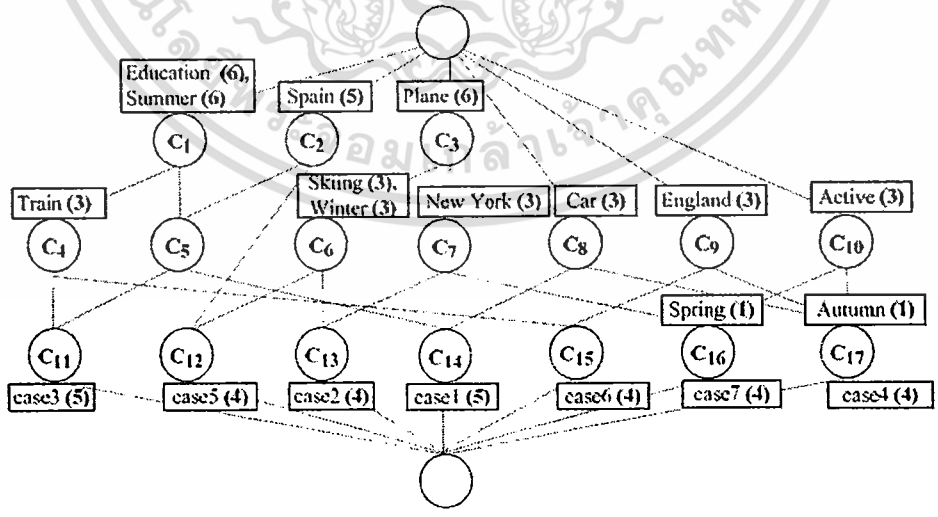


Figure 3.2 Knowledge base in concept lattice form of travel agency domain [6]

Table 3.3 Concept similarity measure obtained from Fig. 3.2

No.	$\sum_u \left(\log\left(\frac{N}{Fa_u}\right)\right)^2$	$\sum_v \left(\log\left(\frac{N}{Fc_v}\right)\right)^2$	$\sum_j \left(\log\left(\frac{N}{Fa_j}\right)\right)^2$	$\sum_l \left(\log\left(\frac{N}{Fc_l}\right)\right)^2$	$Sim(C_P, C_N)$
C_1	0.000	0.000	0.409	0.960	0.000
C_2	0.000	0.000	0.282	0.960	0.000
C_3	0.000	0.790	0.205	1.185	0.408
C_4	0.000	0.000	0.977	0.677	0.000
C_5	0.000	0.000	0.692	0.565	0.000
C_6	0.568	0.000	1.340	0.790	0.325
C_7	0.000	0.000	0.772	0.790	0.000
C_8	0.000	0.790	0.568	0.677	0.540
C_9	0.000	0.000	0.568	0.790	0.000
C_{10}	0.000	0.395	0.568	0.790	0.250
C_{11}	0.000	0.000	1.259	0.282	0.000
C_{12}	0.568	0.000	1.622	0.395	0.296
C_{13}	0.568	0.395	1.907	0.395	**0.626**
C_{14}	0.000	0.000	1.259	0.282	0.000
C_{15}	0.000	0.000	1.544	0.395	0.000
C_{16}	0.000	0.395	2.854	0.395	0.353
C_{17}	0.000	0.000	3.217	0.395	0.000

From Fig. 3.2, we compute similarity between C_N and C_P as shown in Table 3.3. $Sim(C_{13}, C_N) = 0.626$, where E_{13} is {case2}, and I_{13} is {Plane, Skiing, Winter, New York}. We can read from Table 3.3 that C_{13} is retrieved so that we suggest case2. Now, implications between problem descriptions can be used to suggest solution from dependency inside this concept lattice. More detail of implications and its creation can be found in [25]. Our proposed concept similarity and concept lattice format can assist to identify an initial point to suggest informative solution as bottom-up search approach. After we retrieved C_{13}, we obtain

$$Skiing \wedge Winter \rightarrow Plane,$$

$$Skiing \rightarrow Winter,$$

$$NewYork \rightarrow Plane.$$

We can interpret that "if travelers want to go skiing, then they should travel during winter season and they should go to New York by plane".

CHAPTER 4

EXPERIMENTAL EVALUATION

4.1 Data Sets and Environments

We implement case based classification system by using our proposed algorithm for knowledge base of system. Afterwards, we use our similarity measure and other similarity measures (equations (2.1)-(2.5) in Chapter 2) to retrieve previous experience in concept lattice for solving new problem. We use four benchmark data sets from the UCI repository [4]: Balance-Scale, Zoo, Car and Hayes-Roth. A feature of these data sets is hierarchical or non-hierarchical data structure. The detail of these data sets show in Table 4.1. These data sets are discrete data in many-valued formal context. They are transformed by transformational scaling as plain scaling [16]. The transformed attributes in column 4 is new attributes for formal context to build concept lattice. For environment of our implementation, we used the Toshiba protege M600 notebook with Intel(R) Core(TM) 2 Duo 1.66 GHz, and 2 GB of RAM. The operating system was the Windows Vista Home Basic operating system, which executed our program. We programmed the applications by Java with JavaTM 2 SDK, Standard Edition Version 1.4.2 which was built in the NetBeans version 6.0.1.

Table 4.1 The detail of benchmark data sets

Data set	No. case	No. attribute	No. transformed attribute
Balance-Scale	625	4	23
Zoo	101	17	28
Car	1728	6	25
Hayes-Roth	160	5	18

4.2 Accuracy Evaluation

In this research, the performance of the proposed case-based classification is evaluated with classification accuracy using ten-fold cross-validation. Cross-Validation is a statistical method of evaluating and comparing learning algorithms by dividing data into two segments: one used to learn or train a model and the other used to validate the model. In typical cross-validation, the training and validation sets must cross-over in successive rounds such that each data point has

a chance of being validated against. The basic form of cross-validation is *k*-fold cross-validation, where *k* is ten in this work. Fig. 4.1 show division of data set for cross-validation.

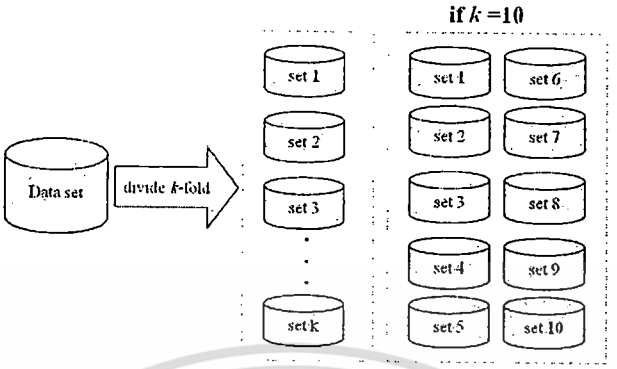
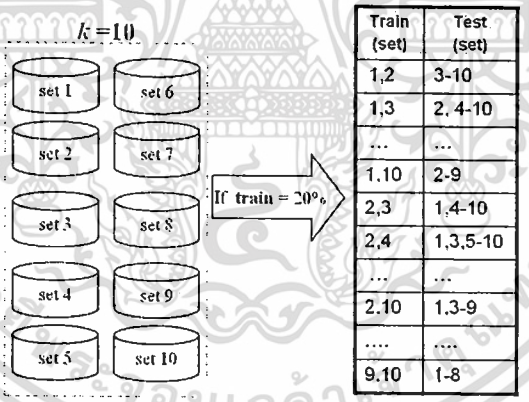


Figure 4.1 *k*-fold cross-validation for each data set

Each data set in our experiment is divided into ten folds as Fig. 4.1. Afterwards, divided data set are crossed-over to reduce bias of data for testing classification accuracy. For example, we can select the divided data set 20% for training set and the rest (80%) for test set shown in Fig. 4.2. The classification accuracy of each training data set derives from the average of classification accuracy from all test data set.



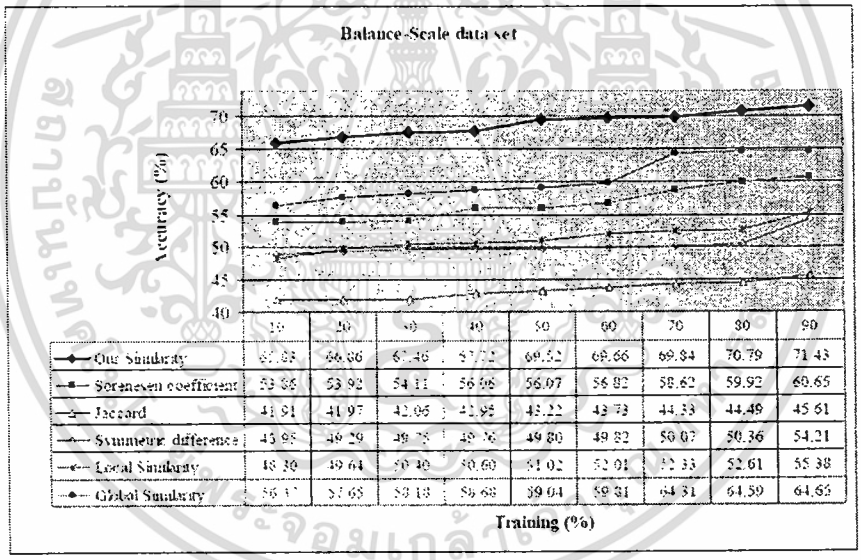
Train (set)	Test (set)
1,2	3-10
1,3	2, 4-10
...	...
1,10	2-9
2,3	1,4-10
2,4	1,3,5-10
...	...
2,10	1,3-9
....	....
9,10	1-8

Figure 4.2 The divided data set for training 20%

There are two possible goals in cross-validation: (i) To estimate performance of the learned model from available data using one algorithm. In other words, to gauge the generalizability of an algorithm, and (ii) To compare the performance of two or more different algorithms and find out the best algorithm for the available data, or alternatively to compare the performance of two or more variants of a parameterized model.

4.3 Experimental Results

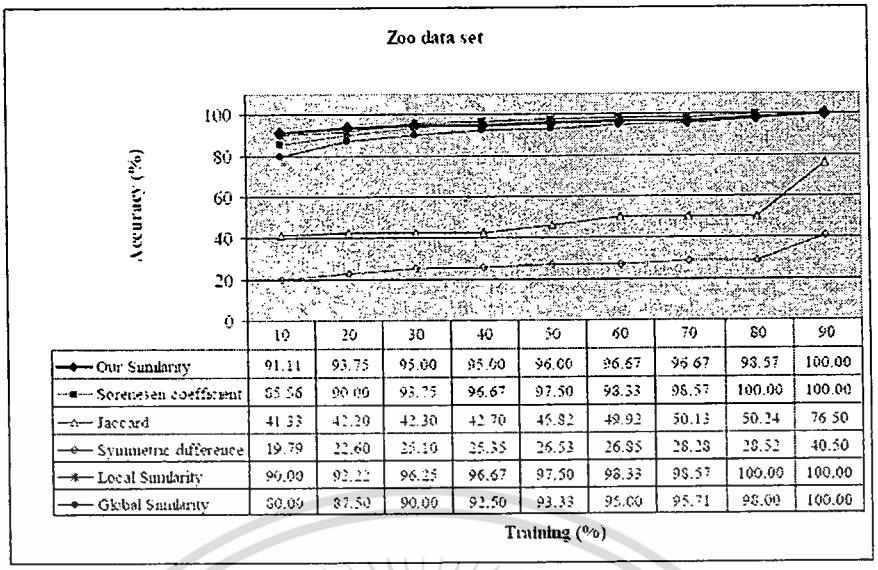
We randomly divide each data set to two sets: training and test sets. Experiments are done on different proportions of these sets e.g., 10% for training set and the rest (90%) for test set and so on. Then, 10-fold cross validation is performed to validate classification accuracy. Fig. 4.3-4.6 show a comparison of obtained classification accuracy for four data sets. Unlike other 5 similarity measures depicted in Fig. 4.3- 4.6, a frequency-based concept similarity measure is outperformed the others. Other similarity measures are based on binary relation and deteriorate classification accuracy whereas our similarity measure is based on vector model and enhances it. Let us observe Fig. 4.4 and 4.5 (hierarchical data) that maximum accuracies are 100%. This result suggest that using our frequency-based concept similarity measure to FCA knowledge base supports hierarchical data structure better than non-hierarchical data structure.



	10	20	30	40	50	60	70	80	90
Our Similarity	62.88	66.86	67.46	67.72	69.52	68.66	69.84	70.79	71.43
Sorenseen coefficient	53.88	53.92	54.11	56.06	56.07	56.82	58.62	59.92	60.65
Jaccard	41.91	41.97	42.06	42.95	43.22	43.73	44.33	44.49	45.61
Symmetric difference	43.95	46.29	48.78	49.76	49.80	49.82	50.03	50.36	54.21
Local Similarity	48.30	49.64	50.40	50.60	51.02	52.01	52.33	52.61	55.38
Global Similarity	56.17	57.65	58.13	58.68	59.04	59.31	64.31	64.59	64.65

Figure 4.3 A comparison of our method and others for Balance-Scale data set

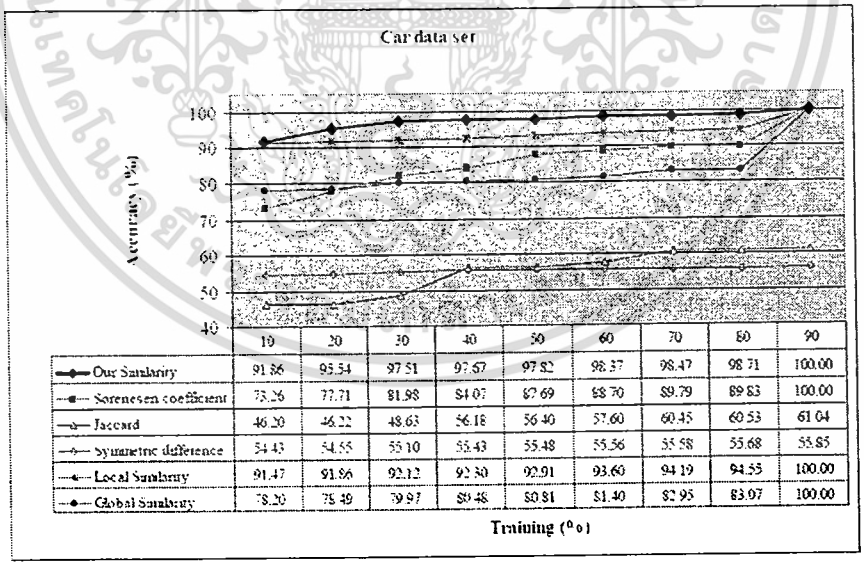
For Balance-Scale data set in Fig. 4.3, the classification using a frequency-based concept similarity measure is better than the existing similarity measures. However, over all using similarity measures can classify this data set with less classification accuracy. For this cause, FCA knowledge base supports hierarchical data structure because it analyzes data in hierarchical data structure form. This data set has a feature of non-hierarchical data. Thus, this feature leads to decrease of classification accuracy when it is compared with non-hierarchical data.



	10	20	30	40	50	60	70	80	90
Our Similarity	91.11	93.75	95.00	95.00	96.00	96.67	96.67	98.57	100.00
Sorenesen coefficient	85.56	90.00	93.75	96.67	97.50	98.33	98.57	100.00	100.00
Jaccard	41.33	42.20	42.30	42.70	45.82	49.92	50.13	50.24	76.50
Symmetric difference	19.79	22.60	25.10	25.35	26.53	26.85	28.28	28.52	40.50
Local Similarity	90.00	92.22	96.25	96.67	97.50	98.33	98.57	100.00	100.00
Global Similarity	80.00	87.50	90.00	92.50	93.33	95.00	95.71	98.00	100.00

Figure 4.4 A comparison of our method and others for Zoo data set

For Zoo data set in Fig. 4.4, the classification using our similarity measure is better than the existing similarity measures. Moreover, over all for every training data set using a frequency-based concept similarity measure satisfy with more classification accuracy because this data set has a hierarchical data feature.

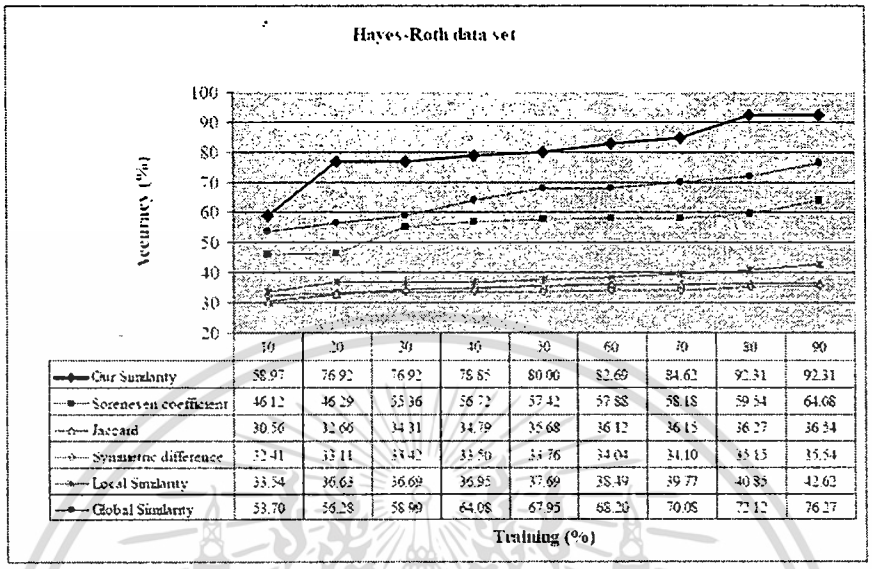


	10	20	30	40	50	60	70	80	90
Our Similarity	91.86	95.54	97.51	97.57	97.82	98.37	98.47	98.71	100.00
Sorenesen coefficient	75.26	77.71	81.98	84.07	87.69	88.70	89.79	89.83	100.00
Jaccard	46.20	46.22	48.63	56.18	56.40	57.60	60.45	60.53	61.04
Symmetric difference	54.43	54.55	55.10	55.43	55.48	55.56	55.58	55.68	55.85
Local Similarity	91.47	91.86	92.12	92.80	92.91	93.60	94.19	94.55	100.00
Global Similarity	78.20	75.49	79.87	80.48	80.81	81.40	82.95	83.07	100.00

Figure 4.5 A comparison of our method and others for Car data set

For Car data set in Fig. 4.5, the classification using our similarity measure is better than the existing similarity measures. Moreover, over all for every training data set using a frequency-

based concept similarity measure satisfy with more classification accuracy because this data set has a hierarchical data feature.



	10	20	30	40	50	60	70	80	90
Our Similarity	58.97	76.92	76.92	78.85	80.00	82.69	84.62	92.31	92.31
Sorenesen coefficient	46.12	46.29	55.36	56.72	57.42	57.88	58.18	59.54	64.08
Jaccard	30.56	32.66	34.31	34.79	35.68	36.12	36.15	36.27	36.34
Symmetric difference	32.41	33.11	33.42	33.50	33.76	34.04	34.10	35.15	35.54
Local Similarity	33.54	36.63	36.69	36.95	37.69	38.49	39.77	40.85	42.62
Global Similarity	53.70	56.28	58.99	64.08	67.95	68.20	70.05	72.12	76.27

Figure 4.6 A comparison of our method and others for Hayes-Roth data set

For Hayes-Roth data set in Fig. 4.6, the classification using a frequency-based concept similarity measure is better than the existing similarity measures. However, over all using similarity measures can classify this data set with less classification accuracy. For this cause, FCA knowledge base supports hierarchical data structure because it analyzes data in hierarchical data structure form. This data set has a feature of non-hierarchical data. Thus, this feature leads to decrease of classification accuracy when it is compared with non-hierarchical data. We compare a non-hierarchical data i.e., Balance-Scale data set. The classification accuracy of this data set is better than Balance-Scale data set. For this reason, Hayes-Roth data set has a small size when it is compared with Balance-Scale data set.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

This thesis proposes a construction of better knowledge base and new concept similarity measure.

Firstly, FCA is applied to build a better knowledge base in CBR system led into the simply solving problem. Namely, FCA can promote efficient retrieval method because each formal concept can identify set of sharing cases as same problem descriptions without calculating every case. For reusing and revising steps, FCA provides both explicit and implicit knowledge. Explicit knowledge can describe both problems with solution and cases of information represented in the hierarchical structure model. Implicit knowledge can elicit knowledge embedded in previous cases with its implication property. In the final step, FCA has incremental structure to facilitate dynamic knowledge base, which is still an incremental nature concept. For these reasons, FCA is a suitable knowledge construction. However, this structure specifically requires a complementary similarity measure because it consists of two sets in one node (graph representation).

Secondly, we propose a new similarity method based on vector model to retrieve previous cases. The proposed method considers weight of data content in knowledge instead of binary relation.

We experiment on several benchmark data sets to determine the performance in term of classification accuracy by implementing a part of classification CBR system. Afterwards, we compare our similarity measure and existing measures with these data sets. Our results indicate that (1) we obtain high improvement of classification accuracy for hierarchical data structure when comparing with non-hierarchical data structure, (2) our similarity measure can be applied successfully for both classification and problem-solving tasks, and (3) our similarity measure can classify data sets better than existing similarity measures.

5.2 Recommendation

In real-world applications, if a word frequency of problem descriptions in each case is high, user highly focuses on case similarly. Thus, in the future work, we should consider weight of words to solve binary relation in FCA by fulfilling this weight into the proposed similarity measure. Moreover, continuous data handling should be improved but this thesis is neglected. Complete CBR system should be developed to automatically retain for making it immediately available.

REFERENCES

[1] Aamodt, A. and Plaza, E. 1994. "Case-based reasoning: foundational issues, methodological variations, and system approaches". **AI Communication.** 7(1): 39-59.

[2] Aamodt, A. 2004. "Knowledge-intensive case-based reasoning in Creek". pp.1-15. in: **Proceedings of the 7th European Conference on Case-Based Reasoning.** Berlin: Springer.

[3] Alqadah, F. 2010. "Similarity measures in formal concept analysis". in: **Workshops of the eleventh International Symposium on Artificial Intelligence and Mathematics (ISIAM2010)**, Fort Lauderdale, Florida.

[4] Asuncion, A. and Newman, D. J. 2007. "UCI Machine Learning Repository" [http://www.ics.uci.edu/ mlearn/MLRepository.html], Irvine, CA: University of California, School of Information and Computer Science.

[5] Bagherjeiran, A. and Eick, C. F. 2008. "Distance function learning for supervised similarity assessment". **Studies in Computational Intelligence (SCI).** 7(3): 91-126.

[6] Belen, D. and Pedro, A. 2001. "Formal concept analysis as a support technique for CBR." **International Journal in Knowledge-based Systems.** 14(1): 163-171.

[7] Champin, P. A. and Solnon, C. 2003. "Measuring the similarity of labeled graphs". pp. 80-95. in: **Proceedings of the 5th International Conference on Case-Based Reasoning.** Berlin: Springer.

[8] Chen, Y. H. and Yao, Y. Y. 2005. "Formal concept analysis based on hierarchical class analysis". pp. 285-292. in: **Proceedings of the 4th IEEE International Conference on Cognitive Informatics.**

[9] Cunningham, P. 2008. "A taxonomy of similarity mechanisms for case-based reasoning". **IEEE transactions on knowledge and data engineering.** pp. 1-22.

[10] Dau, F., Ducrou, J. and Eklund, P. 2008. "Concept Similarity and Related Categories in SearchSleuth". **Lecture Notes Artificial Intelligent (LNAI-5113)**, Springer, pp.255-268.

[11] Dengfeng, L. and Chuntian, C. 2002. "New similarity measures of intuitionistic fuzzy sets and application to pattern recognitions". **Pattern Recognition Letters,** 23(1): 221-225.

[12] De Mantaras, R.L. et. al. 2005. "Retrieval, reuse, revision, and retention in case based reasoning". **Knowledge Engineering Review**. 20(1): 215-240.

[13] Formica, A. 2008. "Concept similarity in formal concept analysis: An information content approach". **Knowledge-Based Systems**. 21(1): 80-87.

[14] Formica, A. 2006. "Ontology-based concept similarity in formal concept analysis". **Information Sciences.** 176(18): 2624-2641.

[15] Ganter, B. and Wille, R. 1997. "Applied lattice theory: Formal concept analysis". Institute for Algebra. TU Dresden, Germany. pp.1-14.

[16] Ganter, B. and Wille, R. 1999. **Formal concept analysis: Mathematical foundation.** Springer, Heidelberg, New York.

[17] Jurisica, I. and Glasgow, J. 1996. "Case-based classification using similarity-based retrieval". pp. 1-10. in: **The 8th IEEE International Conference on Tools with Artificial Intelligence**.

[18] Kolodner, J. 1993. **Case-based reasoning.** USA: Morgan Kaufmann.

[19] Kotsiantis, S. B., Zaharakis, I. D. and Pintelas, P. E. 2007. "Machine learning: A review of classification and combining techniques". **International Journal of Artificial Intelligence Review**. 26(3): 159-190.

[20] Lengnink, K. 2001. "Ahnlichkeit als Distanz in Begrifisverbanden". In G Stumme, R.W., ed. Begrifiiche Wissensverarbeitung: Methoden und Anwendungen, Springer, pp. 57-71.

[21] Luke, K. M., Kalyan, M. G. and David, W. A. 2007. "Case-based collective classification". pp. 399-404. in: **The 20th International FLAIRS Conference**.

[22] McSherry, D. 2003. "Similarity and compromise". pp. 291-305. in: **Proceedings of the 5th International Conference on Case-Based Reasoning**. Berlin: Springer.

[23] Meng, D., Zhang, Z. and Xu, Y. 2005. "A case retrieval model based on factor-structure connection and λ-similarity in fuzzy case-Based Reasoning". **Lecture Notes Artificial Intelligent (LNAI-3613)**. Springer. pp.175-178.

[24] Mougouie, B. and Bergmann, R. 2002. "Similarity assessment for generalized cases by optimization methods". pp. 249-263. in: **Proceedings of the 6th European Conference on Case-Based Reasoning**. Berlin: Springer.

[25] Pattaraintakorn, P., Boonjing, V. and Tadrat, J. 2008. "A New case based classifier system using rough formal concept analysis". pp. 645-650. in: **Proceedings of the 2008 3rd**

International Conference on Convergence and Hybrid Information Technology (ICCIT 2008). Busan, Korea.

[26] Priss, U. 2006. "Formal concept analysis in information science". **Annual Review of Information Science and Technology**. 40(1): 521-543.

[27] Ralph, B., Kolodner, J. and Plaza, E. 2005. "Representation in case-based reasoning". **Knowledge Engineering Review**. 20(4): 209-213.

[28] Ricardo, B. Y. and Berthier, R. N. 1999. **Modern information retrieval.** USA: Addison Wesley.

[29] Rong, Q. B. 2000. "Case-based reasoning for course timetabling problems". PhD Thesis, University of Nottingham.

[30] Salamo, M. and Golobardes, E. 2000. "Weighting methods for a case-based classifier system". in: **Proceedings of the IEEE Learning'00.**

[31] Saquer, J. and Deogun, J. S. 2001. "Concept approximations based on rough sets and similarity measure". **International Journal of Applied Mathematics and Computer Science**. 11(3): 655-674.

[32] Smyth, B. and McClave, P. 2001. "Similarity vs. diversity". pp. 347-361. in: **Proceedings of the 4th International Conference on Case-Based Reasoning.** Berlin: Springer.

[33] Smyth, B., Keane, M. T. and Cunningham, P. 2001. "Hierarchical case-based reasoning integrating case-based and decompositional problem-solving techniques for plant control software design". **IEEE Trans. Knowledge and Data Engineering**. 13(1): 793-812.

[34] Stahl, A. and Gabel, T. 2003. "Using evolution programs to learn local similarity measures". pp. 537-551. in: **Proceedings ofthe 5th International Conference on Case-Based Reasoning**. Berlin: Springer.

[35] Tadrat, J., Boonjing, V. and Pattaraintakorn, P. 2009. "A new similarity measure in formal concept analysis for case-based reasoning". **Expert Systems With Applications.** (to appear).

[36] Tadrat, J., Boonjing, V. and Pattaraintakorn, P. 2008. "Building classification rules for case based classifier using fuzzy sets and formal concept analysis". pp.13-18. in: **Proceeding of the 5th International Conference on Soft Computing as Transdisciplinary Science and Technology (IEEE/ACM CSTST 2008).** Oct 27-31 2008, Paris, France.

[37] Tadrat, J., Boonjing, V. and Pattaraintakorn, P. 2007. "A hybrid case based reasoning system using fuzzy-rough sets and formal concept analysis". pp. 426-429. in: **The 4th International Conference on Fuzzy Systems and Knowledge Discovery**. Haikou, China.

[38] Tadrat, J., Boonjing, V. and Pattaraintakorn, P. 2007. "A framework for using rough sets and formal concept analysis in case based reasoning". pp. 227-232. in: **Proceeding of the 2007 IEEE International Conference on Information Reuse and Integration**. Hilton Hotel, Las Vegas.

[39] Tartakovski, A., Schaaf, M., Maximini, R. and Bergmann, R. 2004. "MINLP based retrieval of generalized cases". pp. 404-418. in: **Proceedings of the 7th European Conference on Case-Based Reasoning**. Berlin: Springer.

[40] Wang, Y. and Ming, L. 2007. "Classification rule acquisition based on extended concept lattice". **Lecture Notes Computer Science (LNCS- 4688)**. Springer. pp. 571-578.

[41] Weber, R.O., Ashley, K.D. and Bruninghaus, S. 2006. "Textual case-based reasoning". **Knowledge Engineering Review**. 20(3): 255-260.

[42] Wille, R. 2005. "Formal concept analysis as mathematical theory of concepts and concept hierarchies". **Lecture Notes Artificial Intelligent (LNAI-3626)**. Springer. pp.1-33.

APPENDIX

APPENDIX A

Publications

1. Jirapond Tadrat, Veera Boonjing, Puntip Pattaraintakorn: "A New Similarity Measure in Formal Concept Analysis for Case-Based Reasoning", in: *Expert Systems With Applications*, vol. 39(1). pp. 967-972. (2012).
2. Jirapond Tadrat, Veera Boonjing, Puntip Pattaraintakorn: "A Framework for Using Rough Sets and Formal Concept Analysis in Case Based Reasoning", in: *The 2007 IEEE International Conference on Information Reuse and Integration (IEEE IRI'07)*, Hilton Hotel, Las Vegas, Aug 13-15 (2007).
3. Jirapond Tadrat, Veera Boonjing, Puntip Pattaraintakorn: "A Hybrid Case Based Reasoning System Using Fuzzy-Rough Sets and Formal Concept Analysis", in: *The 4th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD'07)*, Haikou, China, Aug 24-27 (2007).
4. Puntip Pattaraintakorn, Veera Boonjing, Jirapond Tadrat: "A New Case Based Classifier System using Rough Formal Concept Analysis", in: *Proceedings of the 2008 Third International Conference on Convergence and Hybrid Information Technology (ICCIT 2008) Vol 2*, November 11st-13rd 2008, Busan, Korea, 645-650.
5. Jirapond Tadrat, Veera Boonjing, Puntip Pattaraintakorn: "Building Classification Rules for Case Based Classifier using Fuzzy Sets and Formal Concept Analysis", in: *The fifth International Conference on Soft Computing as Transdisciplinary Science and Technology (IEEE/ACM CSTS08)*, Oct 27-31 2008, Paris, France

Expert Systems with Applications 39 (2012) 967–972

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

A new similarity measure in formal concept analysis for case-based reasoning

Jirapond Tadrat [a,b,*], Veera Boonjing [a,b], Puntip Pattaraintakorn [c,d]

[a] Software Systems Engineering Laboratory, Department of Computer Science, Faculty of Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand
[b] National Centre of Excellence in Mathematics, PERDO, Bangkok 10400, Thailand
[c] Department of Mathematics, Faculty of Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand
[d] Faculty of Science and Engineering, York University, Canada

ARTICLE INFO

Keywords:
Case-based reasoning
Formal concept analysis
Knowledge representation
Concept similarity

ABSTRACT

In this work, we aim at developing a better knowledge base by using formal concept analysis (FCA) and propose its new similarity measure based on vector model for case-based reasoning (CBR). The features of our proposed approaches are illustrated using a part of CBR system for both classification and problem-solving. Concept lattice knowledge base provides more accuracy classification for hierarchical data structure when comparing with non-hierarchical data structure. Dependency induced from our concept lattice knowledge base can help to suggest informative solutions for problem-solving CBR. In addition, our similarity measure improves the accuracy of classification CBR significantly when we perform experiments on the UCI data sets with cross validation.

1. Introduction

A case-based reasoning (CBR) (Aamodt, 2004; Aamodt & Plaza, 1994; Kolodner, 1993; Ralph, Kolodner, & Plaza, 2005) is a method to problem solving that learns from prior experiences. The tasks of CBR system are often divided into; classification and problem-solving CBR (Kolodner, 1993). Classification CBR uses previous cases as reference points for new problem. In contrast, problem-solving CBR uses previous cases to suggest the most applicable solutions to new situation. Both tasks store a set of pairs problem descriptions and solution in their knowledge base for reusing in the future. Traditional CBR consists of four steps (Aamodt & Plaza, 1994; Kolodner, 1993) as follows: retrieve the most similar cases, reuse existing knowledge of previous cases to solve new problem, revise suggested solutions and retain useful parts of this experience for future problem solving as shown Fig. 1.

The structure of knowledge base that directly supports four steps above will make a great effect on efficiency and performance of CBR. Formal concept analysis (FCA) can elicit knowledge embedded in previous cases to solve new problems. FCA is especially well-suited to support CBR system when problem at hand involving hierarchical structure (Belen & Pedro, 2001). In addition, implication drawn from FCA can suggest solutions from dependency inside knowledge base (Pattaraintakorn, Boonjing, & Tadrat, 2008). Thus, we apply FCA to build a knowledge base for CBR. Nevertheless, the knowledge base obtained from FCA technique, called *concept lattice*, requires a specific retrieval process to solve new problem.

The retrieval process is usually regarded as the most important step in the CBR cycle. In essence, a good assessing similarity between cases is a key success of CBR. In the mean time, this retrieval process is directly related to the structure of knowledge base. Thus, both case retrieval process and knowledge base construction must be designed to accord. Hence, we propose a new similarity measure based on vector model that considers contents of data and support retrieval process from concept lattice.

This article is organized as follows. Section 2 provides basic notions of FCA. In Section 3, we briefly review related work for knowledge representation, FCA and similarity measures in CBR. In addition, we define our new concept similarity measure. Section 4 presents how to apply a new similarity measure to retrieve cases for concept lattice knowledge base. In Section 5, we report a case study for classification CBR. Section 6 concludes the article.

2. Formal concept analysis

Formal concept analysis (FCA), invented by Rudolf Wille, is not only a method for data analysis and knowledge representation, but also a formal formulation for concept formation and learning (Ganter & Wille, 1997; Priss, 2006; Wille, 2005). FCA provides relationship of generalization and specialization among concepts through concept lattice (Belen & Pedro, 2001; Chen & Yao, 2005). Practically, FCA starts with a *formal context* which contains values 0 or 1 in an information system. Below, we introduce basic definitions and idea of FCA taken from Ganter and Wille (1999).

* Corresponding author at: Department of Mathematics, King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand.
E-mail addresses: s9062904@kmitl.ac.th (J. Tadrat), kbveera@kmitl.ac.th (V. Boonjing), kppuntip@kmitl.ac.th (P. Pattaraintakorn).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.07.096

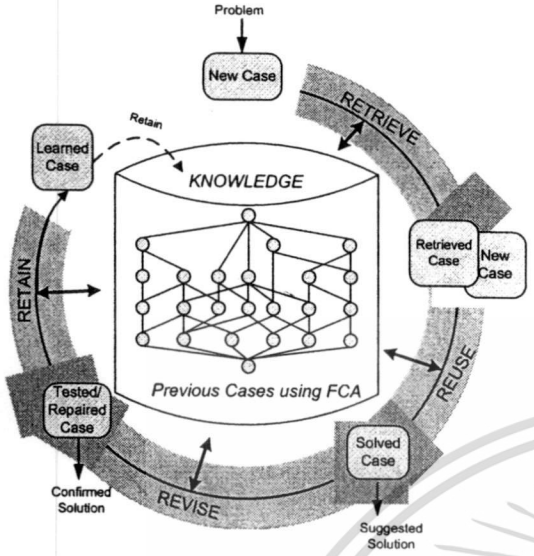


Fig. 1. System overview of CBR process (adapted from Aamodt, 2004).

Definition 1. A formal context $\mathbb{K} := (G, M, I)$ consists of two sets G and M and a relation I between G and M. The elements of G are called the objects and the elements of M are called the attributes of the context. In order to express that an object g is in a relation I with an attribute m, we write gIm or $(g,m) \in I$ and read it as "the object g has the attribute m".

In our CBR system, we represent an object as a case. Attributes are referred to as sets of problem descriptions and solutions. A formal context is considered as a case base. Below, we introduce definition of an association between object and its attributes.

Definition 2. For a set $A \subseteq G$ of objects we define

$A' := \{m \in M | gIm \text{ for all } g \in A\}$

(the set of attributes common to the objects in A). Correspondingly, for a set B of attributes we define

$B' := \{g \in G | gIm \text{ for all } m \in B\}$

(the set of objects which have all attributes in B).

Definition 3. A formal concept of the formal context (G,M,I) is a pair (A,B) with $A \subseteq G$, $B \subseteq M$, $A' = B$ and $B' = A$. We call A the extent and B the intent of the formal concept (A,B). $\mathfrak{B}(G,M,I)$ denotes the set of all formal concepts of the formal context (G,M,I).

Definition 4. If (A_1,B_1) and (A_2,B_2) are formal concepts of formal context, (A_1,B_1) is called a subconcept of (A_2,B_2), provided that $A_1 \subseteq A_2$ (which is equivalent to $B_2 \subseteq B_1$). In this case, (A_2,B_2) is a superconcept of (A_1,B_1), and we write $(A_1,B_1) \leqslant (A_2,B_2)$. The relation $\leqslant$ is called the hierarchical order (or simply order) of the formal concepts. The set of all formal concepts of (G,M,I) ordered in this way denoted by $\underline{\mathfrak{B}}(G,M,I)$ and is called the concept lattice of the formal context (G,M,I).

Concept lattice can be considered as a new structure for knowledge base in CBR. To extract knowledge for solving new problem, The Basic Theorem on Concept Lattices (see more detail in Ganter & Wille (1999)) is used. This theorem provides implications between attributes that are used to identify solution in our work.

Theorem 1 (*The Basic Theorem on Concept Lattice*). *Let T be an index set and, for every $t \in T$. The concept lattice $\underline{\mathfrak{B}}(G,M,I)$ is a complete lattice in which infimum and supremum are given by:*

$$\bigwedge_{t\in T}(A_t,B_t) = \left(\bigcap_{t\in T}A_t, \left(\bigcup_{t\in T}B_t\right)''\right),$$

$$\bigvee_{t\in T}(A_t,B_t) = \left(\left(\bigcup_{t\in T}A_t\right)'', \bigcap_{t\in T}B_t\right).$$

A complete lattice $\mathbf{V}$ is isomorphic to $\underline{\mathfrak{B}}(G,M,I)$ if and only if there are mappings $\tilde{\gamma}: G \to \mathbf{V}$ and $\tilde{\mu}: M \to \mathbf{V}$ such that $\tilde{\gamma}(G)$ is supremum-dense in $\mathbf{V}$, $\tilde{\mu}(M)$ is infimum-dense in $\mathbf{V}$ amd gIm is equivalent to $\tilde{\gamma}g \leqslant \tilde{\mu}m$ for all $g \in G$ and all $m \in M$.

The mappings $\tilde{\gamma}g$ and $\tilde{\mu}m$ in Theorem 1 indicate how formal context can be identified in the concept lattice. This is elaborated by the following definition.

Definition 5. For an object $g \in G$ we write g' instead of $\{g'\}$ for the object intent $\{m \in M | gIm\}$ of the object g. Correspondingly, $m' := \{g \in G | gIm\}$ is the attribute extent of the attribute m. Retaining the symbols used in Theorem 1, we write γg for the object concept (g'',g') and μm for the attribute concept (m',m'').

In CBR, an implication between any two attributes measures dependency by considering problem descriptions and solutions as subconcepts and superconcepts, respectively. Let C and D be sets of problem descriptions and solution where $C, D \subseteq M$, and $C \cap D = \emptyset$. An implication among attributes in M where $M = C \cup D$, is a pair of subsets of M, denoted by $C \to D$.

Proposition 1 (Ganter and Wille, 1999). *An implication $C \to D$ holds in (G,M,I) if and only if $D \subseteq C'$. It then automatically holds in the set of all concept intents as well.*

An implication $C \to D$ holds in (G,M,I) if and only if $C \to m$ holds for each $m \in D$. $C \to m$ holds if and only if $(m',m'') \geqslant (C',C'')$, i.e., if $\mu m \geqslant \wedge\{\mu n | n \in C\}$. This means that we have to check in a concept lattice whether the concept denoted by m is located above the infimum of all formal concepts denoted by n from C.

3. Knowledge representation, FCA and similarity measures in CBR

In this section, we briefly review interesting works of knowledge representation. Moreover, we review the state of the art of FCA and its similarity measures for CBR system. Finally, we define new similarity measure based on vector model which provide more accurate retrieval results.

3.1. Knowledge representation

Similarity-based retrieval in traditional CBR system is often grouped according to knowledge representation (case base structure): feature-vector representation and structure representation as surveyed in Ralph et al. (2005), Cunningham (2008) and De Mantaras et al. (2005).

Feature-vector structure represents every case as attribute-value pairs collected in a data table form. The similarity measure for this structure is based on distance concept between previous cases and new problems. They usually result in k most similar cases to new problem, referred to as *k-nearest neighbor* (*k*-NN). This similarity measure can be embedded as a basic retrieval step in other structures. For instance, Sun, Finnie, and Weber (2004) and Liu, Chen, and Hsc (2008) used their similarity measure to collect cases in the same group of problem descriptions and to retrieve exception cases in classification CBR.

Next, structure representation is a popular method to build case base structure in CBR system, for example, graph representation (Champin & Solnon, 2003; Rong, 2002), object-oriented representation, frame-based representation (Aamodt, 2004; Mougouie & Bergmann, 2002), hierarchical representation (Smyth, Keane, & Cunningham, 2001) and concept lattice (described in Section 3.2). For graph representation, Rong (2002) used undirected graph to represent previous cases in course timetabling problem domain. Next, Champin and Solnon (2003) employed graph representation and their similarity measure of directed graph to retrieve attributes. The advantage of their approach is a flexible model. Their qualitative similarity can be used to adapt solution in CBR system. To capture both specific and general knowledge, Aamodt (2004) invented object-oriented and frame-based representation CBR system. The author utilized similarity measurement in two steps. First, a set of potentially similar cases is retrieved. Next, this obtained set is used as a general domain knowledge to generate explanation for feature-to-feature matches.

Mougouie and Bergmann (2002) devised a similarity measurement for generalized cases. To do so, they applied nonlinear programming for continuous attributes and introduced an optimization-based retrieval method. For hierarchical representation problem descriptions are decomposed into subproblems (Smyth et al., 2001). A set of subproblems is separately solved and then recombined to produce a suitable solution. The advantages lie in the fact that it allows a whole case or its parts to be accessed and exploited by case-based reasoner, and constraints can be employed to guide adaptation.

3.2. FCA and similarity measures in CBR

FCA is successfully applied in several CBR systems (Belen & Pedro, 2001; Pattaraintakorn et al., 2008; Tadrat, Boonjing, & Pattaraintakorn, 2008). Belen and Pedro (2001) utilized FCA for problem solving CBR system. They report that FCA provided good facilitation and suggestion of solutions for new problem. In our initial study (Tadrat, Boonjing, & Pattaraintakorn, 2007), we proposed a framework to construct knowledge base in a CBR system based on rough sets and FCA. Recently, we improved our framework by using fuzzy sets (Tadrat et al., 2008). The result is that fuzzy sets support to build knowledge base by FCA technique successfully. Nevertheless, its similarity measure is computed separately by employing vector model idea. It will be more efficient to compute directly from FCA knowledge base. Thus, this paper fulfills this gap by using a new similarity measure invented for concept lattice structure.

Several researchers developed similarity measures to retrieve formal concept as surveyed in Formica (2008), Formica (2006), Dau, Ducrou, and Eklund (2008), Alqadah (2010), Saquer and Deogun (2001) and Lengnink (2001). Lengnink (2001) defined similarity measures to find similar and relevant concepts: local similarity and global similarity as follows. For any two concepts (A,B) and (C,D) in a formal concept, local similarity measure, s_l, and global similarity measure, s_g, respectively are defined as

$$s_l((A,B),(C,D)) = \frac{1}{2}\left(\frac{|A\cap C|}{|A\cup C|}+\frac{|B\cap D|}{|B\cup D|}\right), \tag{1}$$

$$s_g((A,B),(C,D)) = \frac{1}{2}\left(\frac{|A\cap C|}{|G|}+\frac{|B\cap D|}{|M|}\right). \tag{2}$$

Saquer and Deogun (2001) proposed a similarity measure for concept approximation by using local similarity. It is simple and can approximate extensions when their are only problem descriptions. To define a scope of retrieved results in semantic web, Dau et al. (2008) developed a new combined local and global similarity measure obtained from user. Formica (2006) proposed an adapted version of (1) for semantic web with weight of formal concept specified by user. In fact, this weight should be determined from the contents of data.

In addition, Alqadah (2010) improved existing similarity measures based on set theory which are described below. For any two sets of intension in formal concepts x and y, Jaccard index (s_{Jac}), Sorenesen coefficient (s_{Sor}) and Symmetric difference (s_{Xor}) are defined as

$$s_{Jac}(x,y) = \frac{|x\cap y|}{|x\cup y|}, \tag{3}$$

$$s_{Sor}(x,y) = \frac{2*|x\cap y|}{|x|+|y|}, \tag{4}$$

$$s_{Xor}(x,y) = 1-\frac{|(x\setminus y)\cup(y\setminus x)|}{|x\cup y|}. \tag{5}$$

From the above studies, weights are selected based on user's requirements or by matching user's query and previous cases in binary relation form. Alternatively, we should determine weight directly from data. Thus, we specifically propose a new similarity measure based on vector model that consider problem descriptions and solution with in a concept lattice.

Case retrieval in concept lattice can be done by two distinct ways: lattice traversal and similarity measure. Our target is to use the latter due to its accuracy and timely manners. To invent a new concept similarity measure, we exploit an idea of vector space model which is a classical model of information retrieval (Ricardo & Berthier, 1999) as described below.

Let C_P be a formal concept of formal context (G,M,I) represents a pair (E_P,I_P) of previous cases. Let $E_P \subseteq G$, $I_P \subseteq M$ where E_P is a set of previous cases that have similar problem description(s) and solution while I_P comprises of all problem descriptions and solution shared by all those cases. A new problem is defined as $C_N := (E_N,I_N)$, where E_N is a set of retrieved cases to achieve a solution, and I_N is a set of new problem descriptions provided by user. Initially E_N is calculated from C_P. We have that $E_N = E_P$ if its pair I_P gives $\max|I_N\cap I_P|$. Thus, a new concept similarity measure between a formal concept and new problem is defined as $Sim(C_N,C_P)$. The closer the value of $Sim(C_N,C_P)$ is to 1, the greater the similarity of C_N and C_P.

Definition 6. Given a formal concept of previous case $C_P = (E_P,I_P)$ and a formal concept of new problem $C_N = (E_N,I_N)$ in a formal context (G,M,I), concept similarity measure is defined as

$$Sim(C_P,C_N) = \frac{1}{2}\left(\frac{\sum_{u\in I_N\cap I_P}\left(\log\frac{N}{Fa_u}\right)^2}{\left[\sum_{i\in I_N}\left(\log\frac{N}{Fa_i}\right)^2\sum_{j\in I_P}\left(\log\frac{N}{Fa_j}\right)^2\right]^{1/2}}+\frac{\sum_{v\in E_N\cap E_P}\left(\log\frac{N}{Fc_v}\right)^2}{\left[\sum_{k\in E_N}\left(\log\frac{N}{Fc_k}\right)^2\sum_{l\in E_P}\left(\log\frac{N}{Fc_l}\right)^2\right]^{1/2}}\right),$$

where N is a total number of formal concepts, Fa_u, Fa_i and Fa_j are a frequency of attributes u, i and j, respectively, $\{u,i,j\}\in M$, and Fc_v, Fc_k and Fc_l are frequencies of cases v, k and l, respectively, $\{v,k,l\}\in G$.

Table 1
An example of zoo data set.

Case	Size	Leg	Nature	Action	Class
Case1	Small	2	Feathers	Fly	Poultry
Case2	Small	2	Feathers	–	Poultry
Case3	Small	2	Feathers	Swim	Poultry
Case4	Small	2	Feathers	Fly, hunt	Poultry
Case5	Medium	2	Feathers	Fly, hunt	Poultry
Case6	Medium	4	Hair	Hunt, run	Terrestrial Animal
Case7	Medium	4	Hair	Run	Terrestrial Animal
Case8	Medium	4	Hair	Hunt, run	Terrestrial Animal
Case9	Small	4	Hair	Hunt, run	Terrestrial Animal
Case10	Big	4	Hair	Hunt, run	Terrestrial Animal
Case11	Big	4	Hair	Run	Terrestrial Animal
Case12	Big	4	Hair	–	Terrestrial Animal

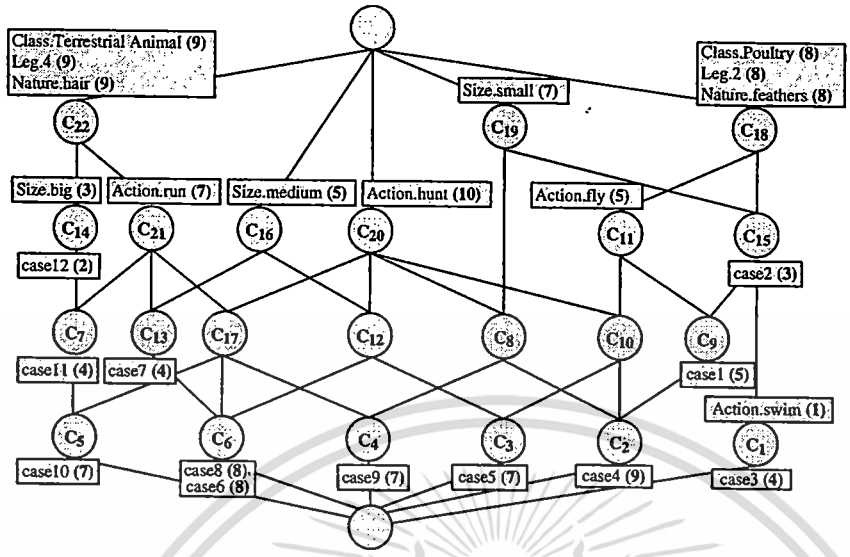


Fig. 2. Our knowledge base in the format of concept lattice.

Table 2
Concept similarity measure obtained from Fig. 2.

No.	$\sum_u \left(\log \frac{N}{Fc_u}\right)^2$	$\sum_v \left(\log \frac{N}{Fc_v}\right)^2$	$\sum_i \left(\log \frac{N}{Fc_i}\right)^2$	$\sum_l \left(\log \frac{N}{Fc_l}\right)^2$	$Sim(C_P, C_N)$
C_{22}	0.151	0.247	0.452	3.061	0.206
C_{21}	0.398	0.247	0.699	1.977	0.313
C_{20}	0.117	0.247	0.247	1.279	0.287
C_{19}	0.000	0.000	0.247	2.110	0.000
C_{18}	0.000	0.000	0.579	2.110	0.000
C_{17}	0.515	0.247	0.817	0.881	0.428
C_{16}	0.000	0.000	0.414	1.181	0.000
C_{15}	0.000	0.000	0.826	1.862	0.000
C_{14}	0.899	0.247	1.201	1.880	0.416
C_{13}	0.398	0.000	1.113	0.934	0.108
C_{12}	0.000	0.000	0.531	0.988	0.000
C_{11}	0.000	0.000	0.993	0.812	0.000
C_{10}	0.117	0.000	1.110	0.398	0.032
C_9	0.000	0.000	1.240	0.565	0.000
C_8	0.117	0.000	0.365	0.398	0.055
C_7	1.147	0.247	1.448	0.795	0.551
C_6	0.515	0.000	1.231	0.386	0.133
C_5	1.264	0.247	1.565	0.247	0.788
C_4	0.515	0.000	1.064	0.247	0.143
C_3	0.117	0.000	1.524	0.247	0.027
C_2	0.117	0.000	1.358	0.151	0.029
C_1	1.802	0.000	2.628	0.548	0.317

4. Knowledge base construction and case retrieval

In this section, we present a construction of better knowledge base by using FCA. Consequently, we describe how to apply a new similarity measure in Definition 6 to retrieve cases. This structure provides both general and specific knowledge, which support both classification and problem-solving CBR as described in Sections 4.1 and 4.2, respectively.

4.1. Classification CBR

In this section, we demonstrate retrieval process from concept lattice by using our proposed similarity measure for classification CBR. Let we given a zoo data set (Table 1) where each case is described by problem descriptions (columns 2–5) and its solution (column 6). This table involves classification of two types of animal: Poultry and Terrestrial Animal. We construct our knowledge base from Table 1 by using FCA as shown in Fig. 2. More detail of this concept lattice creation can be found in Tadrat et al. (2008), Tadrat et al. (2007) and Pattaraintakorn et al. (2008).

Concept lattice in Fig. 2 consists of formal concept, C_1–C_{22}, intensions (upper labels) and extensions (lower labels). The number in (symbol) refers to a frequency of cases (attributes). To retrieve, we compute similarity value between new problem and each formal concept. For example, given a new unseen problem C_N with

I_N = {Size.big, Leg.4, Nature.smooth, Action.run, Action.swim, Action.hunt}.

The aim of this classification CBR is to classify C_N to either classes Poultry or Terrestrial Animal. By Definition 6, we obtain similarity between C_N and C_P as shown in Table 2, where $\sum_{i\in} I_N \left(\log\left(\frac{N}{Fc_i}\right)\right)^2 = \left(\log\frac{22}{3}\right)^2 + \left(\log\frac{22}{9}\right)^2 + \left(\log\frac{22}{7}\right)^2 + \left(\log\frac{22}{1}\right)^2 + \left(\log\frac{22}{10}\right)^2 = 3.066$, and $\sum_{k\in E_N}\left(\log\left(\frac{N}{Fc_k}\right)\right)^2 = \left(\log\frac{22}{7}\right)^2 = 0.247$.

As one can see, a formal concept C_5 in Fig. 2 will be retrieved with $Sim(C_5, C_N) = 0.788$, where E_5 is {case10}, and I_5 is {Size.big, Leg.4, Nature.hair, Action.hunt, Action.run, Class.Terrestrial Animal}. Thus, a solution of new unseen problem, C_N, is Terrestrial Animal class. This new unseen problem can be retained to solve new problems in our CBR system. Please note that, if problem descriptions of new problem are exactly as same as existing previous case, then similarity value is 1 ($Sim(C_P, C_N) = 1$) and classification accuracy is 100%.

4.2. Problem-solving CBR

In this section, we illustrate problem-solving CBR by an example given in Belen and Pedro (2001). It describes travel agency domain where every case represents description of the journeys offered by a travel agency. Fig. 3 depicts a knowledge base in a concept lattice form of this data which contains 7 cases (see more details in Belen & Pedro (2001)). The objective of this problem solving example is to suggest journey from user's query: *Skiing*.

From Fig. 3, we compute similarity between C_N and C_P as shown in Table 3. $Sim(C_{13}, C_N) = 0.626$, where E_{13} is {case2}, and I_{13} is

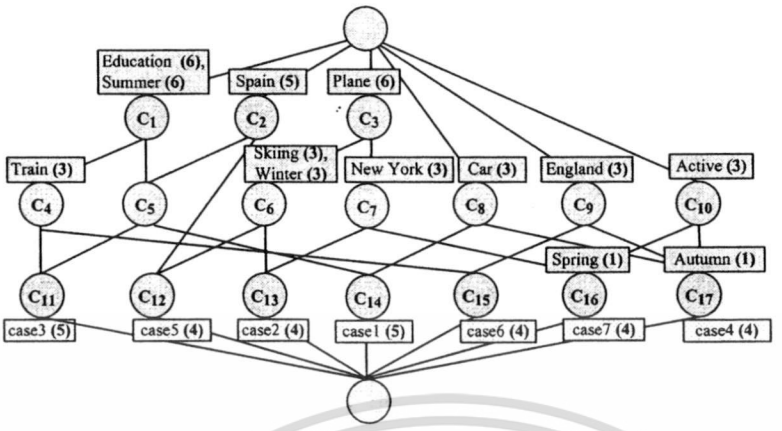
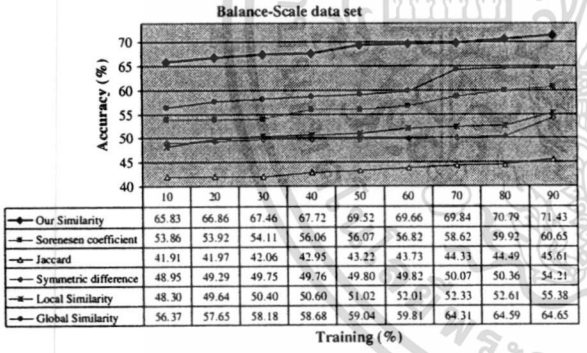


Fig. 3. Knowledge base in concept lattice form of travel agency domain (Belen & Pedro, 2001).

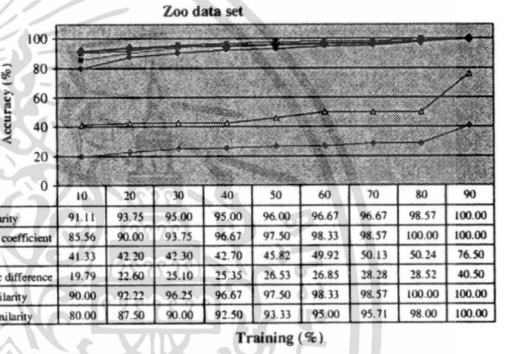
Table 3
Concept similarity measure obtained from Fig. 3.

No.	$\sum_u \left(\log \frac{N}{Fa_u}\right)^2$	$\sum_v \left(\log \frac{N}{Fc_v}\right)^2$	$\sum_j \left(\log \frac{N}{Fa_j}\right)^2$	$\sum_l \left(\log \frac{N}{Fc_l}\right)^2$	$Sim(C_P, C_N)$
C_1	0.000	0.000	0.409	0.960	0.000
⋮	⋮	⋮	⋮	⋮	⋮
C_{12}	0.568	0.000	1.622	0.395	0.296
C_{13}	0.568	0.395	1.907	0.395	**0.626**
⋮	⋮	⋮	⋮	⋮	⋮
C_{17}	0.000	0.000	3.217	0.395	0.000



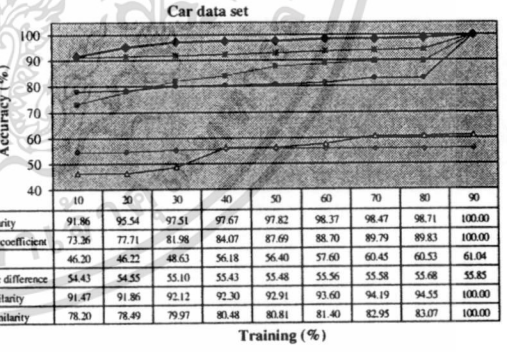
	10	20	30	40	50	60	70	80	90
Our Similarity	65.83	66.86	67.46	67.72	69.52	69.66	69.84	70.79	71.43
Sorenesen coefficient	53.86	53.92	54.11	56.06	56.07	56.82	58.62	59.92	60.65
Jaccard	41.91	41.97	42.06	42.95	43.22	43.73	44.33	44.49	45.61
Symmetric difference	48.95	49.29	49.75	49.76	49.80	49.82	50.07	50.36	54.21
Local Similarity	48.30	49.64	50.40	50.60	51.02	52.01	52.33	52.61	55.38
Global Similarity	56.37	57.65	58.18	58.68	59.04	59.81	64.31	64.59	64.65

Fig. 4. A comparison of our method and others for balance-scale data set.



	10	20	30	40	50	60	70	80	90
Our Similarity	91.11	93.75	95.00	95.00	96.00	96.67	96.67	98.57	100.00
Sorenesen coefficient	85.56	90.00	93.75	96.67	97.50	98.33	98.57	100.00	100.00
Jaccard	41.33	42.20	42.30	42.70	45.82	49.92	50.13	50.24	76.50
Symmetric difference	19.79	22.60	25.10	25.35	26.53	26.85	28.28	28.52	40.50
Local Similarity	90.00	92.22	96.25	96.67	97.50	98.33	98.57	100.00	100.00
Global Similarity	80.00	87.50	90.00	92.50	93.33	95.00	95.71	98.00	100.00

Fig. 5. A comparison of our method and others for Zoo data set.



	10	20	30	40	50	60	70	80	90
Our Similarity	91.86	95.54	97.51	97.67	97.82	98.37	98.47	98.71	100.00
Sorenesen coefficient	73.36	77.71	81.98	84.07	87.69	88.70	89.79	89.83	100.00
Jaccard	46.20	46.22	48.63	56.18	56.40	57.60	60.45	60.53	61.04
Symmetric difference	54.43	54.55	55.10	55.43	55.48	55.56	55.58	55.68	55.85
Local Similarity	91.47	91.86	92.12	92.30	92.91	93.60	94.19	94.55	100.00
Global Similarity	78.20	78.49	79.97	80.48	80.81	81.40	82.95	83.07	100.00

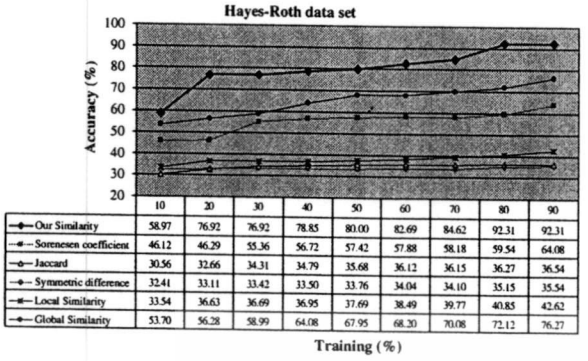
Fig. 6. A comparison of our method and others for Car data set.

{Plane, Skiing, Winter, New York}. We can read from Table 3 that C_{13} is retrieved so that we suggest case2. Now, implications between problem descriptions can be used to suggest solution from dependency inside this concept lattice. More detail of implications and its creation can be found in Pattaraintakorn et al. (2008). Our proposed concept similarity and concept lattice format can assist to identify an initial point to suggest informative solution as bottom-up search approach. After we retrieved C_{13}, we obtain *Skiing* ∧ *Winter* → *Plane*, *Skiing* → *Winter*, *New York* → *Plane*. We can interpret that if travelers want to go skiing, then they should travel during winter season and they should go to New York by plane.

5. A case study: classification CBR

In this section, we implement a part of classification CBR system based on concept lattice knowledge base. Afterwards, we use our similarity measure and other similarity measures (Eqs. (1)–(5) in Section 3) to retrieve previous experience in concept lattice for solving new problem. We use four benchmark data sets from the UCI repository Asuncion and Newman (2007): Balance-Scale, Zoo, Car and Hayes-Roth. These data sets are divided into two groups that are hierarchical data structure i.e., Zoo and Car, and non-hierarchical data structure i.e., Balance-Scale and Hayes–Roth. We randomly divide each data set into two sets: training and test sets. Experiments are done on different proportions of these sets e.g., 10% for training set and the rest (90%) for test set and so on. Then, 10-fold cross validation is performed to validate classification accuracy.

Figs. 4–7 show a comparison of obtained classification accuracy for four data sets. Unlike other 5 similarity measures depicted in



	10	20	30	40	50	60	70	80	90
Our Similarity	58.97	76.92	76.92	78.85	80.00	82.69	84.62	92.31	92.31
Sorensen coefficient	46.12	46.29	55.36	56.72	57.42	57.88	58.18	59.54	64.08
Jaccard	30.56	32.66	34.31	34.79	35.68	36.12	36.15	36.27	36.54
Symmetric difference	32.41	33.11	33.42	33.50	33.76	34.04	34.10	35.15	35.54
Local Similarity	33.54	36.63	36.69	36.95	37.69	38.49	39.77	40.85	42.62
Global Similarity	53.70	56.28	58.99	64.08	67.95	68.20	70.08	72.12	76.27

Fig. 7. A comparison of our method and others for Hayes–Roth data set.

Figs. 4–7, our similarity measure is outperformed the others. Other similarity measures are based on binary relation and deteriorate classification accuracy whereas our similarity measure is based on vector model and enhances it. Let us observe Figs. 5 and 6 (hierarchical data) that maximum accuracies are 100%. This result suggest that using our similarity measure to FCA knowledge base supports hierarchical data structure better than non-hierarchical data structure.

6. Conclusion

This paper proposes a construction of better knowledge base and new concept similarity measure. Firstly, FCA is applied to build a better knowledge base in CBR system. Secondly, we propose a new similarity method based on vector model to retrieve previous cases. We compare our similarity measure and existing measures with the UCI data sets by implementing a part of classification CBR system. Our results indicate that (1) we obtain high improvement of classification accuracy for hierarchical data structure when comparing with non-hierarchical data structure, (2) our similarity measure can be applied successfully for both classification and problem-solving tasks, and (3) our similarity measure can classify data sets better than existing similarity measures.

Acknowledgements

The authors are deeply grateful Commission on Higher Education, Thailand. We also would like to thank Centre of Excellence in Mathematics, Thailand. A part of this research has been supported by NSERC, Canada.

References

Aamodt, A. (2004). Knowledge-intensive case-based reasoning in Creek. In *Proceedings of the Seventh European Conference on Case-Based Reasoning* (pp. 1–15). Berlin: Springer.

Aamodt, A., & Plaza, E. (1994). Case-based reasoning: foundational issues, methodological variations, and system approaches. *AI Communication, 7*(1), 39–59.

Alqadah, F. (2010). Similarity measures in formal concept analysis. In *Workshops of the eleventh international symposium on artificial intelligence and mathematics (ISIAM2010), Fort Lauderdale, Florida.*

Asuncion, A., & Newman, D. J. (2007). UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine, CA. <http://www.ics.uci.edu/mlearn/MLRepository.html>.

Belen, D., & Pedro, A. (2001). Formal concept analysis as a support technique for CBR. *International Journal in Knowledge-based Systems, 14*(1), 163–171.

Champin, P. A., & Solnon, C. (2003). Measuring the similarity of labeled graphs. In *Proceedings of the fifth international conference on case-based reasoning* (pp. 80–95). Berlin: Springer.

Chen, Y. H., & Yao, Y. Y. (2005). Formal concept analysis based on hierarchical class analysis. In *Proceedings of the 4th IEEE international conference on cognitive informatics* (pp. 285–292).

Cunningham, P. (2008). A taxonomy of similarity mechanisms for case-based reasoning. *IEEE Transactions on Knowledge and Data Engineering.*

Dau, F., Ducrou, J., & Eklund, P. (2008). In *Concept Similarity and Related Categories in SearchSleuth. Lecture Notes Artificial Intelligent (LNAI)* (vol. 5113). Springer.

De Mantaras, R. L., Mcsherry, D., Bridge, D., Leake, D., Smyth, B., Craw, S., et al. (2005). Retrieval, reuse, revision, and retention in case based reasoning. *Knowledge Engineering Review, 20,* 215–240.

Formica, A. (2006). Ontology-based concept similarity in formal concept analysis. *Information Sciences, 176*(18), 2624–2641.

Formica, A. (2008). Concept similarity in formal concept analysis: An information content approach. *Knowledge-Based Systems, 21*(1), 80–87.

Ganter, B., & Wille, R. (1997). *Applied lattice theory: Formal concept analysis.* TU Dresden, Germany: Institute for Algebra.

Ganter, B., & Wille, R. (1999). *Formal concept analysis: Mathematical foundation.* Heidelberg, New York: Springer.

Kolodner, J. (1993). *Case-based reasoning.* USA: Morgan Kaufmann.

Lengnink, K. (2001). Ähnlichkeit als Distanz in Begriffsverbanden. In G. Stumme & R. Wille (Eds.), *Begriffliche Wissensverarbeitung: Methoden und Anwendungen* (pp. 57–71). Springer.

Liu, C. H., Chen, L. S., & Hsc, C. C. (2008). An association-based case reduction technique for case-based reasoning. *Information Sciences, 178*(17), 3347–3355.

Mougouie, B., & Bergmann, R. (2002). Similarity assessment for generalized cases by optimization methods. In *Proceedings of the Sixth European Conference on Case-Based Reasoning* (pp. 249–263). Berlin: Springer.

Pattaraintakorn, P., Boonjing, V., & Tadrat, J. (2008). A New case based classifier system using rough formal concept analysis. In *Proceedings of the 2008 third international conference on convergence and hybrid information technology, Busan, Korea* (pp. 645–650).

Priss, U. (2006). Formal concept analysis in information science. *Annual Review of Information Science and Technology, 40*(1), 521–543.

Ralph, B., Kolodner, J., & Plaza, E. (2005). Representation in case-based reasoning. *Knowledge Engineering Review, 20*(4), 209–213.

Ricardo, B. Y., & Berthier, R. N. (1999). *Modern information retrieval.* USA: Addison Wesley.

Rong, Q. B. (2002). *Case-based reasoning for course timetabling problems.* PhD Thesis, University of Nottingham.

Saquer, J., & Deogun, J. S. (2001). Concept approximations based on rough sets and similarity measure. *International Journal of Applied Mathematics and Computer Science, 11*(3), 655–674.

Smyth, B., Keane, M. T., & Cunningham, P. (2001). Hierarchical case-based reasoning integrating case-based and decompositional problem-solving techniques for plant-control software design. *IEEE Transactions on Knowledge and Data Engineering, 13,* 793–812.

Sun, Z., Finnie, G., & Weber, K. (2004). Case base building with similarity relations. *Information Sciences, 165*(1–2), 21–43.

Tadrat, J., Boonjing, V., & Pattaraintakorn, P. (2007). A framework for using rough sets and formal concept analysis in case based reasoning. In *Proceedings of the 2007 IEEE international conference on information reuse and integration, Hilton Hotel, Las Vegas* (pp. 227–232).

Tadrat, J., Boonjing, V., & Pattaraintakorn, P. (2008). Building classification rules for case based classifier using fuzzy sets and formal concept analysis. In *Proceedings of the fifth international conference on soft computing as transdisciplinary science and technology (IEEE/ACM CSTS08), Paris, France* (pp. 13–18).

Wille, R. (2005). Formal concept analysis as mathematical theory of concepts and concept hierarchies. *Lecture notes artificial intelligent (LNAI)* (Vol. 3626, pp. 1–33). Springer.

A Framework for using Rough Sets and Formal Concept Analysis in Case Based Reasoning

Jirapond Tadrat
Software Systems Engineering Laboratory, Department of Mathematics and Computer Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand 10520
s9062904@kmitl.ac.th

Veera Boonjing
Software Systems Engineering Laboratory, Department of Mathematics and Computer Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand 10520
kbveera@kmitl.ac.th

Puntip Pattaraintakorn
Department of Mathematics and Computer Science, Faculty of Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand 10520
kppuntip@kmitl.ac.th

Abstract

A significant open problem of case based reasoning system is a construction of better knowledge base. We propose a new framework for constructing alternative knowledge base in case based reasoning system based on rough sets and formal concept analysis. Our framework first applies rough set theory for discovering reduced cases required in a case based reasoning system. We then achieve further hierarchical structure of knowledge base using formal concept analysis. The result is the concept lattice knowledge base embedded to our proposed case based reasoning system. A part of case based reasoning system is developed with an example throughout. We also discuss how our proposed framework can be beneficial for a case based reasoning system.

1. Introduction

A case based reasoning (CBR) [1,8,9,12] is a method to problem solving that learns from prior experience, stored in the case form. A single *case* represents specific knowledge tie to a context. Several cases are stored in the *case base*. The case base (after several learning experiences) will be constructed as the *knowledge base*. A CBR system uses this knowledge base for the future problem solving. Traditional CBR processes are (i) retrieving from previous cases, (ii) reusing the information in that case, (iii) revising the solution and (iv) retaining a new experience into the knowledge base [1,12]. In order to develop a proper CBR system, we require knowledge base supporting the four processes above which can be achieved by suitable knowledge construction i.e., knowledge representation, knowledge acquisition and knowledge organization. First, single case can be represented by various forms in the knowledge base [4,12] depending on type of the data, learning experience and etc. Second, knowledge acquisition is obtained from the initial knowledge modeling and/or successive knowledge maintenance [3,12]. Finally, knowledge organization is a technique to rearrange knowledge base to be helpful for retrieving previous case process. Therefore, approaches to construct knowledge base are key successes for a CBR system.

The tasks of CBR system are often divided into; interpretive CBR and problem-solving CBR [3,8,9]. Interpretive CBR uses previous cases as reference points for new situation. In contrast, problem-solving CBR uses previous cases to only suggest the most applicable solutions to new situation. Both tasks store objects that are the properties as well as problem descriptions in knowledge base for reusing in the future. However, storing too many cases can result in very sizeable memory requirements and slow execution speed. In addition, knowledge base that comprises large number of objects causes oversensitivity to noise so called *overfitting*. Thus, we propose to alleviate this overfitting problem using rough set theory (RST) and formal concept analysis (FCA).

RST was introduced by Pawlak in 1982 [19,20]. It is a mathematical approach broadly used for classification and data analysis [20]. Several researchers applied RST approaches for knowledge acquisition in CBR systems [10,11,17]. RST can be applied to CBR systems for the problems involved hierarchical structure attributes [6], finding weighting of attributes and reducing objects technique [10]. The advantages of RST are finding minimal sets of data, evaluating significance of data and providing efficient algorithms for finding hidden patterns in data [19]. According to several strengths of RST, our framework uses RST for finding reduced cases and alleviating overfitting challenging CBR researchers.

1-4244-1500-4/07/$25.00 ©2007 IEEE

Nevertheless, stand alone RST cannot identify hierarchy without deploying some heuristics

FCA, invented by Wille [13], is a method for data analysis based on concept lattices. The problem of generating the set of all concept of concept lattice is extensively studied in the literature [14]. It is widely used for information science [15] in order to describe a natural attributes of information representation in hierarchical structure model. FCA is especially well suited to identify groups of objects with some common properties. Since FCA and RST offer complementary approaches for data analysis as "*FCA focuses on concepts that are definable by conjunctions of properties. Besides, RST focuses on concepts that are definable by disconjunctions of properties. They produce different types of rule summarizing knowledge embedded in data.*" [18]. Thus, we combine FCA to RST in the present paper. FCA based on *formal context* which is a binary relation between a set of object and a set of attributes. In general, case base contains several multi-value attributes. Therefore, the transformation from multi-value attribute to binary attributes is needed in FCA approach. Nevertheless, the transformation creates expanded and redundant cases in case base. RST attribute reduction is used once again to this case base. However, the expanded and redundant case base still cause problem in retrieving process. Thus, in our complete system, we will use latent semantic indexing (LSI) [2] for deriving solution from previous cases.

In this paper we focus on knowledge representation, knowledge acquisition and knowledge organization in CBR system. To reach the desired CBR system, we amalgamate RST and FCA to construct knowledge base. In Section 2, we describe related works. Section 3 provides preliminary of RST and FCA. Section 4 presents our proposed CBR framework. Section 5 shows an example of our propose CBR system. Section 6 concludes the article and presents future works.

2. Related works

Wang et al. applied RST to CBR system for knowledge representation [6]. Wierzbicki represented cases in hierarchical structure form and extracted dependency rules from knowledge base using the indiscernibility relation [7]. Ziarko used RST for derivation of predictive models from the data [16]. This work also proposed to use hierarchical structure of decision table for solving flat table structure. Computation of uncertain rules and reducing the boundary region by using rough set is studied as well. Salamo et al. developed the classifier system called BASTIAN based on RST [10,11]. They applied rough sets for weighting method and reducing instances in the case base.

Belen et al. [5] studied the usefulness of FCA for supporting CBR to discover knowledge embedded in the case base. The advantage of FCA is to automatically elicit the attribute dependency inside the case base. In addition, FCA is used to complete the knowledge already acquired by other techniques of domain modeling. Unfortunately, traditional FCA supports only binary relation in the data table and real situation often includes multi-value attributes. When FCA researchers faced this problem, they simply transformed the values of attributes to be new binary attributes. The result is loss of some information from original attributes. In addition, this transformation outputs large amount of new binary attributes and cases. Thus, our proposed system architecture designs for solving these problems using RST.

3. Primitives of FCA and RST

3.1. Rough sets

In this section, we provide some concepts of RST from [10,16,19,20]. A data table has columns that are labeled by attributes, rows are labeled by objects of interest and entries of the table that are attribute values. We have *universe* (U), which are non-empty finite set of N objects $\{x_1, x_2, x_3, ..., x_N\}$, called *cases* in the case base. Set of attributes (A) in data table consists of two disjoint classes of attributes, called *condition attributes* (C), described problem description, and *decision attributes* (D), described goal or solution. Associate set of values of attribute (V_a), called *domain*.

Definition 1 (Information system). An information *system* is a pair $S = (U,A)$, where U and A, are non-empty finite sets called the *universe*, and the set of attributes, respectively such that a: $U \rightarrow V_a$, where V_a is the set of all values of a called the *domain* of a.

If we distinguish in the information system to two disjoint classes of attributes, called *condition* and *decision attributes*, respectively, then the system will be called a *decision table* and will be denoted by $S = (U,C,D)$, where C and D are disjoint sets of condition and decision attributes, respectively and $C \cup D = A$.

Example1. From Table 1, U.= {C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8}, $A = C \cup D$ = {a_1, a_2, a_3, a_4, a_5} = {Destination, Type, Trans, Region, Season}, C = {Destination, Type, Trans, Region}, D = {Season}, $V_{a1} = V_{Destination}$ = {Spain, New York}, $V_{a2} = V_{Type}$ = {Education, Active}, $V_{a3} = V_{Trans}$ = {Car, Train, Plane}, $V_{a4} = V_{Region}$ = {Sweden, France}and $V_{a5} = V_{Season}$ = {Summer, Winter}.

Definition 2 (Indiscernibility relation). An equivalence relation, referred to as *indiscernibility relation*, denoted by *IND(P)* is associated with any subset of attribute $P \subseteq A$. This relation is defined as:

$$IND(P) = \{(x, y) \in U \times U : a(x) = a(y),\ \forall a \in P\}.$$

The elements of U satisfying relation *IND(P)* are indiscernible by attributes from P. *U/IND(P)* denotes the equivalence class (the partition) of *IND(P)*.

Definition 3 (Lower approximation). Let $S = (U, A)$ be an information system, $R \subseteq A$ and $X \subseteq U$. We can approximate X using only the information contained in R by constructing the *R-lower approximation* of X, denoted by $\underline{RX}$, we define as $\underline{RX} = \cup\{Y \in U/IND(R): Y \subseteq X\}$.

Definition 4 (Positive region). For attribute $A = C \cup D$, C and D are condition and decision attributes, respectively, and for a given set of condition attribute $R \subset C$, we can define the *P-positive region* ($POS_R(D)$) in the relation *IND(D)* as: $POS_R(D) = \cup\{\underline{RX} : X \in IND(D)\}$.

Definition 5 (Indispensable). Attribute $c \in R$ is a *dispensable* feature in attribute subset $R \subset C$, if $POS_{R-\{c\}}(D) = POS_R(D)$, otherwise attribute c is *indispensable* attribute.

Definition 6 (Reduct). Let $R' \subseteq R$. R' is a *reduct* of R, if $POS_{R'}(D) = POS_R(D)$, and every attribute c: $c \in R'$ is indispensable respect to R'.

Therefore, the *reduct* is minimal attribute subset of information system, which classify capability is equal to original attribute set of information system. It is obviously that there may be several reducts of an information system.

Definition 7 (Core). *Core* of information system S is the intersection of all reduct of S, i.e. $CORE(S) \cap RED(S)$ Where *RED(S)* is the reduct of information system S.

RST provides methods to choose the most important attributes based on core (Definition 6) and reduct (Definition 7). A reduct is essential part of information system, which suffices to define all objects occurring in the knowledgebase. An example of using the core and reduct is provided in Section 4.

3.2. Formal concept analysis

FCA [6,13,14] is a mathematical approach to data analysis based on the lattice theory. There is a binary relation between set of objects and a set of attributes. Form binary relation, one can construct hierarchical structure of concept; each concept is the unification of objects and attributes. It realizes the relationship of generalization and specialization among concepts through Hass diagram [6].

The *formal context* is defined as a triple *(G, M, I)*, (data table in RST) where there are two sets G (objects) and M (attributes), and a binary (incidence) relation $I \subseteq G \times M$. Attributes can be described each object i.e., $(g,m) \in I$ if the object g carries the attribute m (or m is a descriptor of the objects g). An example of formal context when it is depicted by a *cross table* is in Tables 3 and 4. With a general perspective, a concept represents a group of objects and is described by using attributes (its intent) and objects (its extent). The extend covers all objects belonging to the concept while the intent comprises all attributes shared by all those objects. With $A \subseteq G$ and $B \subseteq M$ the following operator (prime) is defined as:

$$A' = \{m \in M \mid (\forall g \in A): (g,m) \in I\},$$
$$B' = \{g \in G \mid (\forall m \in B): (g,m) \in I\}.$$

A pair (A,B) where $A \subseteq G$ and $B \subseteq M$ is said to be a *formal concept* of the context (G,M,I) if $A' = B$ and $B' = A$. A and B are called *extent* and *intent* of the concept, respectively. The set of all the formal concepts of a context (G,M,I) is denoted by $\beta(G,M,I)$. The most important structure on $\beta(G,M,I)$ is given by *subconcept-superconcept ordered relation* denoted by $\leq$ and is defined as follows: $(A_1,B_1) \leq (A_2,B_2)$ if $A_1 \subseteq A_2$ (which is equivalent to $B_2 \subseteq B_1$).

Let (G,M,I) be a context, $(\beta(G,M,I), \leq)$ is a complete lattice called *concept lattice* (Fig. 2) of the context (G,M,I), for which *infimum* and *supremum* are defined as:

$$\text{Inf}\beta(G,M,I) \equiv \bigwedge_{\alpha}(A_\alpha, B_\alpha) = [\bigcap_{\alpha}(A_\alpha, (\bigcup_{\alpha} B_\alpha)'')],$$

$$\text{Sup}\beta(G,M,I) \equiv \bigvee_{\alpha}(A_\alpha, B_\alpha) = [\bigcup_{\alpha}(A_\alpha, (\bigcap_{\alpha} B_\alpha)'')].$$

4. Our proposed CBR framework

Fig. 1 is an illustration of our proposed CBR framework. CBR processes usually are: retrieving, reusing, revising and retaining as mentioned in Section 1. In a CBR system, new problems are solved by retrieving (middle right box) previous case [1,8,12]. After that, previous case is reused and revised (bottom right) the solution in the case that acquires suggestion from previous learning. Finally, derived solution is tested and decided for retaining (top left) new case [12]. Our initial framework presented in this paper involves three main tasks of knowledge construction; knowledge representation, knowledge acquisition and knowledge organization as mentioned in Section 1 by using RST and FCA techniques. We split tasks in our framework into two main modules depending on the techniques used. The first module is *Reduced Case Base*. We use RST to this module to reduce size of case base. The second module is *Discovering Relationships* in case base. This module uses FCA in order to discover the relation in the case base. We

describe the first module in Section 4.1 and the second module in Section 4.2.

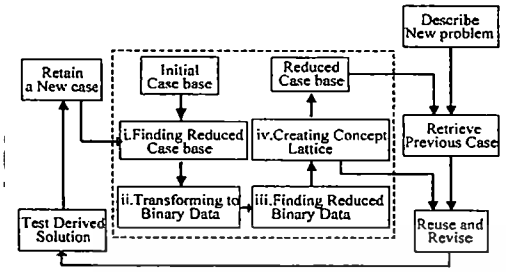


Figure 1. Our proposed framework CBR system based on RST and FCA techniques

4.1. Determining reduced cases

In a CBR system, knowledge starts from initial case base that still requires many experiences to create proper knowledge base. If the knowledge contains too many cases, CBR system faces the overfitting problem which affects the retrieval process. Therefore, we first use RST for reducing cases. We use *positive region* and *core attribute* (Definitions 4, 5) in RST for finding reduced and evaluating the attributes in the case base, respectively. Finding reduced cases in our framework is occurred twice. The first *reduced cases* ((i) in Fig. 1) occur in the initial case base process to reduce noisy cases and overfitting. The second *reduced cases* ((iii) in Fig. 1) occur when new case base is adjusted to binary form. The results are a large number of binary attributes. Therefore, we use rough sets for eliminating unnecessary attributes and/or cases. We obtain reduced cases from rough sets but we still cannot identify co-appearance of properties for each case in the case base. A technique for alleviating this problem is FCA as described in the next section.

4.2. Discovering relationships

To make FCA applicable, we have to transform the case base containing several multi-value attributes to binary attributes by *Transformed Binary Relation* ((ii) in Fig. 1). The output is the new case base in binary relation form. The resulting case base is expanded both attributes and objects that why we apply rough sets once again (as mentioned in Section 4.1). Finally ((iv) in Fig. 1), FCA can identify co-appearance in case base with *Create Concept Lattice*.

5. An example

Our framework in Fig. 1 includes four main processes: (i) finding reduced case base, (ii) transforming case base to binary data table, (iii) finding reduce binary data table, and (iv) creating concept lattice. Initially, we have the initial case base (data table that store several cases) as in Table 1.

We describe details of our methodology to the case base in Table 1 as follows:

(i) Finding reduced case base ((i) in Fig. 1). We can apply RST to the Table 1 with the following four sub-processes to obtain reduced set of cases.

a. Find indiscernibility relation (*IND(P)*) with respect to decision attribute: D = {Season} (refer to Definitions 1, 2). We obtain indiscernibility relations:
U/IND (Season) = $\{\{C_1, C_3, C_6\}, \{C_2, C_4, C_5, C_7, C_8\}\}$ where X_1 = *U/IND*[Season = Summer] = $\{C_1, C_3, C_6\}$, and X_2 = *U/IND*[Season=Winter] = $\{C_2, C_4, C_5, C_7, C_8\}$.

b. Find core attributes. We begin with consider condition attributes (C) and create lower approximation ($\underline{RX}$) of each condition attribute and then find core attributes (Definitions 3-5). We first consider {Destination} and assign C = R = {Destination, Type, Trans, Region}, and R_1 = {Type, Trans, Region}. Hence, $U/IND(R_1)$=$\{\{C_1,C_6\}, \{C_2,C_8\}, \{C_3\}, \{C_4\}, \{C_5\}, \{C_7\}\}$. Lower approximations: $\underline{R_1X_1}$ = $(\{C_1,C_6\} \subseteq \{C_1, C_3, C_6\}) \cup (\{C_2, C_8\} \subseteq \{C_1, C_3, C_6\}) \cup (\{C_3\} \subseteq \{C_1, C_3, C_6\}) \cup (\{C_4\} \subseteq \{C_1, C_3, C_6\}) \cup (\{C_5\} \subseteq \{C_1, C_3, C_6\}) \cup (\{C_7\} \subseteq \{C_1, C_3, C_6\})\}$ = $\{C_1, C_3, C_6\}$, $\underline{R_1X_2}$ = $\{C_2, C_4, C_5, C_7, C_8\}$. Positive regions: $POS_R(D)$ = $\underline{RX_1} \cup \underline{RX_2}$ = $\{C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8\}$, $POS_{R_1}(D)$ = $\underline{R_1X_1} \cup \underline{R_1X_2}$ = $\{C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8\}$

Because of $POS_R(D) = POS_{R_1}(D)$, thus {Destination} is not a core attribute. We repeat this b. step to the rest condition attributes. Only one core obtained from Table 1 is {Trans}.

c. Find reduct attributes. We refer to Definition 6 and explain this step using Table1, we set the reduct attribute from selected core and dispensable attributes as {Trans, Type}.

d. Eliminate dispensable attribute and redundant objects by using reduct attributes. The reduced of initial case base results are in Table 2.

(ii) Transforming to binary data table ((ii) in Fig. 1). We use FCA technique for transforming data in Table 2 to binary data as in Table 3 by transforming each value in Table 2 to new attributes in Table 3.

(iii) Finding reduced binary data table ((iii) in Fig.1). We use RST again to reduce binary data table in Table 3. We repeat processes (i)a-b to Table 3. We create sets of core and reduct attributes. Dispensable attributes as well as redundant objects are eliminated to retain reduced initial case base as shown in Table 4.

Table 1. An example of decision table or case base

	Destination	Type	Trans	Region	Season
C1	Spain	Education	Car	Sweden	Summer
C2	New York	Active	Car	France	Winter
C3	New York	Education	Train	Sweden	Summer
C4	Spain	Education	Plane	France	Winter
C5	New York	Education	Plane	Sweden	Winter
C6	New York	Education	Car	Sweden	Summer
C7	New York	Active	Plane	France	Winter
C8	Spain	Active	Car	France	Winter

Table 2. A Reduced case base

	Type	Trans	Season
C1	Education	Car	Summer
C2	Active	Car	Winter
C3	Education	Train	Summer
C4	Education	Plane	Winter
C7	Active	Plane	Winter

(iv) Creating concept lattice ((iv) in Fig 1.). After transformed (ii) and reduced case base (iii), we use FCA technique again to create formal concept as in Table 5. We create the relation with concept lattice for Table 4 as in Fig. 2. We can use this concept lattice, which is hierarchical structure, in order to reuse and revise solution for solving new problem.

Table 3. Transformed binary relation in new case base

	Education	Active	Car	Train	Plane	Summer	Winter
C1	1	0	1	0	0	1	0
C2	0	1	1	0	0	0	1
C3	1	0	0	1	0	1	0
C4	1	0	0	0	1	0	1
C7	0	1	0	0	1	0	1

Table 4. Reduced binary data table using RST

	Education	Active	Car	Train	Summer	Winter
C1	1	0	1	0	1	0
C2	0	1	1	0	0	1
C3	1	0	0	1	1	0
C4	1	0	0	0	0	1
C7	0	1	0	0	0	1

Figure 2 illustrates a structure of concept lattices associated to Table 4 by *Hasse diagrams* [5,6] that is a graphical representation of formal context. Each node represents a formal concept (Table 5) of formal context (Table 4). Each edge between nodes represents the subconcept-superconcept relation. The label of each node are intent (inside [ ]) and extent (inside {}). The intent is derived from union of the attributes in label [ ]. Similarly, the extent is derived from union of the cases in label {}. Besides, a structure of concept lattice provides dependence knowledge inside in case base. Namely, if label [ ] consists of several attributes, then there is a co-appearance of all these attributes for all the cases in case base. In addition, the lower node is dependent to the upper node for each edge.

This example illustrated two advantages of our proposed framework. The first advantage is our proposed system always retains the reduced (necessary) case base. Thus, reducing in space and time of computation is obtained (helpful for very large data). This reduced case also solved overfitting problem. The second advantage is the creation of hierarchical structure of case base that results in better knowledge base. We can summarize that Table 5 and Fig. 2 are new knowledge base that provide better support processes of retrieving, reusing, and revising for CBR system when comparing to Table 1. The attribute dependency knowledge is extracted e.g., the edges between *winter* and *active* or between *education* and *summer*. For example, from Fig. 2, we can suggest the new solution, if travel occurs in *summer* then holiday type is *education*.

Table 5. Formal concept from Table 4

($\beta(G,M,I)$: set of formal concept		
Formal concept	**Extent Case No.**	**Intent**
Top	1,2,3,4,7	∅
A	2,4,7	Winter
B	1,3,4	Education
C	2,7	Active, Winter
D	1,2	Car
E	1,3	Education, Summer
G	4	Education, Winter
H	3	Education, Train, Summer
I	2	Active, Car, Winter
J	1	Education, Car, Summer
Bottom	∅	Education, Active, Car Train, Summer, Winter

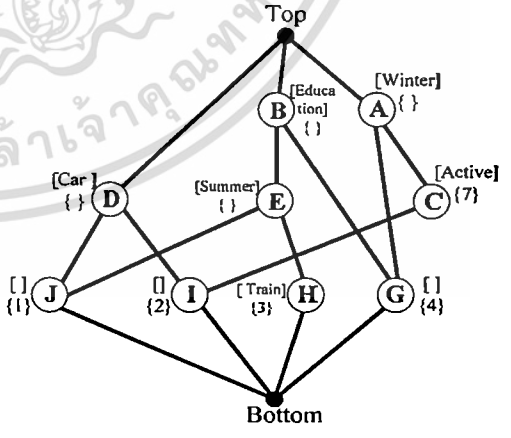


Figure 2. A structure of concept lattices by Hasse diagram

6. Conclusion and future work

We proposed novel framework based rough set theory and formal concept analysis for applying in a CBR system. We used RST for alleviating overfitting problem in case base. We also used FCA as an automatic technique to elicit the attribute dependency knowledge inside the case base. In addition, FCA can identify the relation in the hierarchy of concept in concept lattice and prevent the redundancy of information in the case base.

In the future, we will use fuzzy sets for knowledge representation and LSI for achieving the solution. LSI will use to map new problem description with previous case for retrieving process. The expected result of our complete framework with LSI is the most similar case retrieving from new large case base. An open problem is the knowledge maintenance for retaining process that remains complicated for inserting new case base.

Acknowledgments

The authors are deeply grateful Commission on Higher Education and King Mongkut's Institute of Technology Lardkrabang research grant, Thailand.

7. References

[1] A. Aamodt, E. Plaza, "Case-based reasoning: foundational issues, methodological variations, and system approaches", *AI Communication*, vol. 7, pp. 39-59, 1994.

[2] A. Kontostathis, W. Pottenger, "A framework for understanding latent semantic indexing (LSI) performance", *Information Processing and Management*, vol. 42, pp. 56-73, 2006.

[3] A. Aamodt, *A Knowledge-Intensive, Integrated Approach to Problem Solving and Sustained Learning*, PhD Thesis, University of Trondheim, Norway, May, 1991.

[4] B. Ralph, J. Kolodner and E. Plaza, "Representation in case-based reasoning," *Knowledge Engineering Review*, vol. 20, pp. 209-213, 2005.

[5] D. Belen, M.G. Antonio, P.G. Pablo and A. Pedro, "Formal concept analysis for knowledge refinement in case based reasoning", *In Proc. of the 25th International Conference on Innovative Techniques and Applications of Artificial Intelligence*, 2005, pp. 233-245.

[6] H. Wang, W. Zhang, "Relationships between concept lattice and rough set," *In Proc. of the 8th on Artificial Intelligence and Soft Computing*, 2006, pp. 538-547.

[7] J. Wierzbicki, "Rough set approach to CBR," *In Proc. of the 2nd of Rough Sets and Current Trends in Computing* (RSCTC 2000), Canada, Berlin: Springer, 2001, pp. 503-510.

[8] J. Kolodner, *Case-Based Reasoning*, Morgan Kaufmann, USA, 1993.

[9] B.D. Leaked, *Case-based Reasoning: Experiences, Lessons, and Future Directions*, AAAI press/MIT Press, Cambridge, MA, USA, 1996.

[10] M. Salamo, E. Golobardes, "Rough sets reduction techniques for case-based reasoning," *In Proc. of Case-Based Reasoning* (ICCBR), 2001, pp. 467-482.

[11] M. Salamo, E. Golobardes, "Weighting Methods for a Case-Based Classifier System," *In Proc. of the IEEE Learning*, 2000.

[12] R.L. De Mantaras et al., "Retrieval, reuse, revision, and retention in case based reasoning," *Knowledge Engineering Review*, vol. 20, pp. 215-240, 2005.

[13] R. Wille, "Formal concept analysis as mathematical theory of concepts and concept hierarchies," *Formal Concept Analysis: Foundations and Applications*, LNAI 3626, Berlin: Springer, 2005, pp. 1-33.

[14] S.O. Kuznetsov, S.A. Obiedkov, "Algorithm for the construction of concept lattices and their diagram raphs," *In Proc. of the 5th Principles of Data Mining and Knowledge Discovery: European Conference*, Freiburg, Germany, September 3-5, LNCS 2168, Berlin: Springer, 2001, pp. 289-300.

[15] U. Priss, "Formal concept analysis in information science, *Annual Review of Information Science and Technology*, vol. 40, pp. 521-543, 2006.

[16] W. Roman, "Rough set methods in feature reduction and classification," *Int. Appl. Math. Comput. Sci.*, vol. 11, pp. 565–582, 2001.

[17] W. Ziarko, "Acquisition of hierarchy-structured probabilistic decision tables and rules from data," *Knowledge Engineering, Expert Systems*, vol. 20, pp. 305-610, 2003.

[18] Y.Y. Yao, "A comparative study of formal concept analysis and rough set theory in data analysis", *In Proc. of the International* Conference on Rough Sets and Current Trends in Computing (RSCTC), 2004, pp. 59-68.

[19] Z. Pawlak, "Rough sets and data analysis," *In Proc. of the Fuzzy Systems Symposium, Soft Computing in Intelligent Systems and Information Processing*, 1996, pp. 1-6.

[20] Z. Pawlak, "Rough sets and intelligent data analysis," *Information Sciences Informatics and Computer Science*, vol. 147, pp. 1-12, 2002.

A Hybrid Case Based Reasoning System using Fuzzy-Rough Sets and Formal Concept Analysis

Jirapond Tadrat
Software Systems Engineering Laboratory, Department of Mathematics and Computer Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand 10520
s9062904@kmitl.ac.th

Veera Boonjing
Software Systems Engineering Laboratory, Department of Mathematics and Computer Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand 10520
kbveera@kmitl.ac.th

Puntip Pattaraintakorn
Department of Mathematics and Computer Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand 10520
kppuntip@kmitl.ac.th

Abstract

In this paper, we propose a new hybrid case based reasoning system based on rough set theory, formal concept analysis and fuzzy sets. This system applies rough set theory to assure minimally sufficient cases in its case base. It uses formal concept analysis to reveal knowledge of attribute dependencies in terms of concept lattices. Numeric attributes are transformed to be suitable for formal concept analysis using fuzzy sets.

1. Introduction

A case based reasoning (CBR) [7,11] is a method to problem solving using prior experience. Its process includes retrieving from previous cases, reusing the information in that case, revising the solution and retaining a new experience into the knowledge base. The process requires suitable knowledge construction [2,11] i.e., knowledge representation, acquisition and organization. Thus, construction of knowledge base is a key success for a CBR system. Since the CBR system solve a new problem using previous solved cases, it may store a number of redundant previous cases. This causes oversensitivity to noise so called *overfitting problem*. Therefore, rough set theory (RST) is proposed to solve the problem [5,10,16]. Nevertheless, RST alone cannot reveal hierarchical structures and co-appearance of case properties. However, formal concept analysis (FCA) is especially well suited to identify groups of cases sharing common properties [12,14]. Therefore, we combined FCA to fulfill RST for extracting hidden relations among values of attributes. Unfortunately, the FCA transformation for numeric attributes introduces substantial new attributes. Such new attributes lead to more space and time requirements for the CBR process mentioned above. These attributes also form less useful formal concept and concept lattice. Thus, we propose to transform numeric attributes using fuzzy sets. Since both FCA and fuzzy sets transformations might give an overwhelming number of case attributes, RST is used once again to assure minimally sufficient cases. Finally, we propose to form concept lattices from reduced transformed cases.

This article is organized as follows. Section 2 provides details of rough sets, FCA and fuzzy sets. We present our proposed CBR system as well as its illustrative example in Section 3. Section 4 concludes the article.

2. Primitives of FCA, RST and Fuzzy sets

2.1. Rough sets

In this section, we provide RST concepts from [15,17]. An information system is a data table, whose columns are labeled by attributes, rows are labeled by objects and entries of the table are attribute values. We define universe (U), which are finite (not null) set of N objects $\{x_1, x_2, x_3, ..., x_N\}$, called cases in the case base. Set of attributes (A) in data table consists of two disjoint classes of attributes: condition attributes (C) describing problem description and decision attribute (D) describing goal or solution. Associated set of values of attribute (V) is called domain of V. The function mapped from objects to attribute values is called *information function f*.

Information system is a data table. A data table is $T = (U, A, V, f)$, where $A = C \cup D$ ($C \cup D = \varnothing$), $V = \cup_{a \in A} V_a$, then f: $U \times A \rightarrow V$ is an information function such that $f(x,a) \in V, \forall x \in U, a \in A$.

Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007)
0-7695-2874-0/07 $25.00 © 2007 IEEE

IEEE COMPUTER SOCIETY

Example1. From Table 1, $U = \{C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8\}$, $A = C \cup D = \{a_1, a_2, a_3, a_4, a_5, a_6, a_7\}$ = {Destination, Type, Trans, Region, Age, Price, Season}, C = {Destination, Type, Trans, Region, Age, Price }, D = {Season}, $V_{a1} = V_{Destination}$ = {Spain, New York}, $V_{a2} = V_{Type}$ = {Educstion, Active}, $V_{a3} = V_{Trans}$ = {Car, Train, Plane}, $V_{a4} = V_{Season}$ = {Summer, Winter}, $f(C_1, a_1) = f(C_1, Destination)$ = {Spain}.

The *indiscernibility relation* is an equivalence relation over U. $\forall P \subseteq A$ determines an indiscernibility relation $IND(P)=\{(x_i, x_j) : (x_i, x_j) \in U \times U, a \in P, f(x_i, a) = f(x_j, a)\}$, where x_i, x_j are objects i and j where $i, j \in \{1, 2, ..., N\}$, $i \neq j$, respectively. All indiscernibility relation in P represented by $U/IND(P)$.

In an information system T, for each subset $X \subseteq U$ and an equivalence relation $R \subseteq A$, *lower approximation* is defined as follows:
$\underline{RX} = \cup\{Y \in U/IND(R): Y \subseteq X\}$.

For attribute $A = C \cup D$, C and D are condition and decision attributes, respectively, and for a given set of condition attribute $R \subset C$, we can define the *positive region* ($POS_R(D)$) in the relation $IND(D)$ as:
$POS_R(D) = \cup \{\underline{RX} : X \in IND(R)\}$.

An attribute $C_j \in C$ is a *dispensable* attribute in C with respect to D if $POS_C(D) = POS_{C-C_j}(D)$.

An attribute $C_j \in C$ is a *core* attribute in C with respect to D if $POS_C(D) \neq POS_{C-C_j}(D)$.

An attribute $C_j \in C$ is a *reduct* attribute if C_j is part of a dispensable and/or core attributes (a reduced set of attributes) where $POS_R(D) \neq POS_{C_j}(D)$.

2.2. Formal concept analysis

FCA [1,3,5,12] is a mathematical approach to data analysis based on the lattice theory. The *formal context* is defined as a triple (G, M, I), (data table) where there are two sets G (objects) and M (attributes), and a binary (incidence) relation $I \subseteq G \times M$. Attributes can describe each object i.e., $(g,m) \in I$ if the object g carries the attribute m. Concept represents a group of objects and is described by using attributes (its intent) and objects (its extent). The extent covers all objects belonging to the concept while the intent comprises all attributes shared by all those objects. With $A \subseteq G$ and $B \subseteq M$, following operator (prime) is defined as: $A' = \{m \in M \mid (\forall g \in A): (g,m) \in I\}$, $B' = \{g \in G \mid (\forall m \in B): (g,m) \in I\}$. A pair (A,B) where $A \subseteq G$ and $B \subseteq M$ is said to be a *formal concept* of the context (G,M,I) if $A' = B$ and $B' = A$ where A, B are *extent* and *intent* of concept, respectively. The set of all the formal concepts of a context (G,M,I) is denoted by $\beta(G,M,I)$. The most important structure on $\beta(G,M,I)$ is given by *subconcept-superconcept ordered relation* ($\leq$) and is defined as: $(A_1,B_1) \leq (A_2,B_2)$ if $A_1 \subseteq A_2$. Let (G,M,I) be a context, $(\beta(G,M,I), \leq)$ is a complete lattice called *concept lattice* of the context (G,M,I), for which *infimum* and *supremum* are defined as: $\mathrm{Inf}\beta(G,M,I) = [\bigcap_{\alpha}(A_\alpha, (\bigcup_{\alpha} B_\alpha)'']$, $\mathrm{Sup}\beta(G,M,I) = [\bigcup_{\alpha}(A_\alpha, (\bigcap_{\alpha} B_\alpha)'']$.

Real world data table consists of multi-value attributes. Thus, FCA researchers developed formal context that can describe multi-value attributes [1,14]. In [1,3], formal context (for multi-value attributes) is defined as $(G, M, (W_m)_{m \in M}, I)$ where G is a set of object, M is a set of attribute, each W_m is a set of possible values for attribute $m \in M$, and $I \subseteq G \times \{(m,w) \mid m \in M, w \in W_m\}$ is a relation that $(g,m,w_1) \in I, (g,m,w_2) \in I \Rightarrow w_1 = w_2$. $(g,m,w) \in I$.

2.3 Fuzzy sets

Given a universe set, X, and a membership function, $\mu : X \rightarrow [0,1]$, a fuzzy set [8,13] is a collection of pairs: $\{(x, \mu(x)): x \text{ in } X\}$. The example of membership functions are piecewise-linear membership function [4], π-membership function [6,9], and etc. Figure 1 is the triangular membership functions defined over a universe set of real numbers. Therefore, there are 5 fuzzy sets defined: MIN, LMD, MDM, HMD, and MAX.

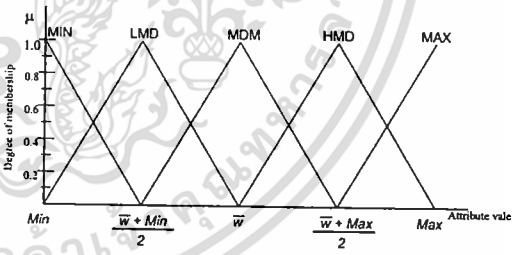


Figure 1. An example of triangular membership function.

3. Our proposed hybrid CBR system

Our proposed hybrid CBR system assures minimally sufficient cases and provides attribute dependency knowledge in term of concept lattices in four steps. The first step it uses RST to reduce size of case base. Next, it transforms the case base containing several multi-value attributes to binary attributes. Descriptive and numeric attributes are transformed using FCA and fuzzy sets, respectively. The output is new transformation case base that expand both attributes and objects. Thus, in the third we apply RST once again to obtain reduced case base. In

urth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007)
'695-2874-0/07 $25.00 © 2007 IEEE

the final step, FCA is applied to create concept lattice. The following example illustrates our proposed system.

Suppose the initial case base is as shown in Table 1. For simplicity, we use {E, A} for *Education* and *Active* of {Trans}, {C, T, P} for *Car, Train* and *Plane* of {Type}, and {S, W} for *Summer* and *Winter* of {Season}.

Table 1. An example of decision table or case base

	Destination	Type	Trans	Region	Age	Price	Season
C1	Spain	Education	Car	Sweden	20	120	Summer
C2	New York	Active	Car	France	45	350	Winter
C3	New York	Education	Train	Sweden	50	280	Summer
C4	Spain	Education	Plane	France	15	200	Winter
C5	New York	Education	Plane	Sweden	25	450	Winter
C6	New York	Education	Car	Sweden	30	550	Summer
C7	New York	Active	Plane	France	35	330	Winter
C8	Spain	Active	Car	France	20	120	Winter

Table 2. A reduced case base

	Type	Trans	Age	Price	Season
C1	Education	Car	20	120	Summer
C2	Active	Car	45	350	Winter
C3	Education	Train	50	280	Summer
C4	Education	Plane	15	200	Winter
C5	Education	Plane	25	450	Summer
C6	Education	Car	30	550	Winter
C7	Active	Plane	35	330	Winter

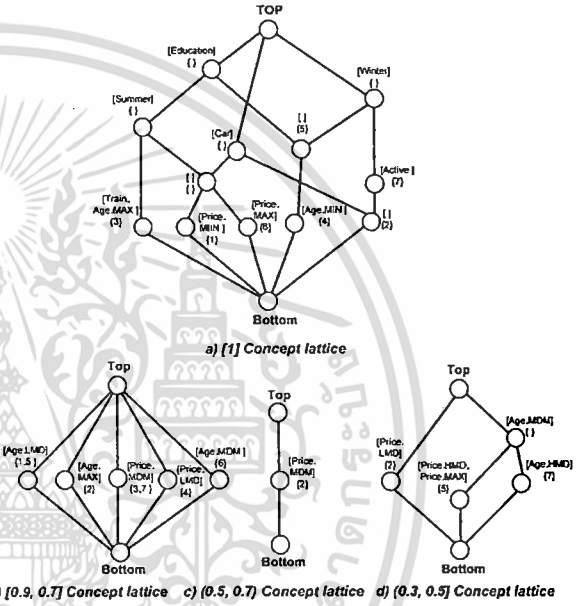
Table 3. Transformed relation in new case base

	Trans		Type			Age					Price					Season	
	E	A	C	T	P	MIN	LMD	MDM	HMD	MAX	MIN	LMD	MDM	HMD	MAX	S	W
C1	1	0	1	0	0	0.2	0.8	0	0	0	1	0	0	0	0	1	0
C2	0	1	1	0	0	0	0	0	0.2	0.8	0	0.4	0.6	0	0	0	1
C3	1	0	0	1	0	0	0	0	0	1	0	0.2	0.8	0	0	1	0
C4	1	0	0	0	1	1	0	0	0	0	0.1	0.9	0	0	0	0	1
C5	1	0	0	0	1	0	0.7	0.3	0	0	0	0	0	0.5	0.5	0	1
C6	1	0	1	0	0	0	0.1	0.9	0	0	0	0	0	0	1	1	0
C7	0	1	0	0	1	0	0	0.3	0.4	0	0	0	0.8	0.2	0	0	1

Table 4. Reduced data table using RST

	Trans		Type		Age					Price					Season	
	E	A	C	T	MIN	LMD	MDM	HMD	MAX	MIN	LMD	MDM	HMD	MAX	S	W
C1	1	0	1	0	0.2	0.8	0	0	0	1	0	0	0	0	1	0
C2	0	1	1	0	0	0	0	0.2	0.8	0	0.4	0.6	0	0	0	1
C3	1	0	0	1	0	0	0	0	1	0	0.2	0.8	0	0	1	0
C4	1	0	0	0	1	0	0	0	0	0.1	0.9	0	0	0	0	1
C5	1	0	0	0	0	0.7	0.3	0	0	0	0	0	0.5	0.5	0	1
C6	1	0	1	0	0	0.1	0.9	0	0	0	0	0	0	1	1	0
C7	0	1	0	0	0	0	0.3	0.4	0	0	0	0.8	0.2	0	0	1

Using RST to determine sets of core attributes and reduct attributes, we obtain core = {Trans} and reduct = {Trans, Type, Age, Price}. Dispensable attributes as well as redundant objects are eliminated to retain reduced initial case base as shown in Table 2. Next we use FCA to transform descriptive attributes: Trans, Type, and Season and fuzzy sets to transform numeric attributes: Age and Price. The later transformation uses triangular membership function defined in Figure 1. Table 3 shows the result of these transformations. Table 4 is reduced version of Table 3 using RST. The last step, we use four predefined relationship intervals ([0.3, 0.5], (0.5, 0.7), [0.7,0.9], and [1]) over the linguistic variables {Age, Price} in Table 4 to create four concept lattices as shown in Figure 2. Each node represents a formal concept of formal context (Table 4). Each edge represents subconcept-superconcept relation. Labels of each node are intent (inside [ ]) and extent (inside {}). Intent is derived from union of the attributes in label [ ]. Similarly, extent is derived from union of the cases in label {}. Structure of concept lattice provides dependence knowledge in case base. If label [ ] consists of several attributes, then there is a co-appearance of such attributes for all the cases in case base. In addition, lower node is dependent on the upper node for each edge.



a) [1] Concept lattice

b) [0.9, 0.7] Concept lattice c) (0.5, 0.7) Concept lattice d) (0.3, 0.5] Concept lattice

Figure 2. Concept lattices generated from Table 4

4. Conclusion

We propose a hybrid CBR system based on RST, FCA and fuzzy sets to achieve minimally sufficient case base while well supporting CBR process. RST is used to solve overfitting problem. FCA is used as an automatic technique to discover attribute dependency knowledge from case base based on predefined intervals of relationship values. Numeric attributes are transformed into predefined fuzzy sets to be suitable for FCA. This way this CBR system assures minimally sufficient cases and provides attribute dependency knowledge in terms of concept lattices. However, effectiveness of this new CBR system relies on predefined fuzzy sets associated with numeric attributes and predefined intervals of relationship values.

Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007)
0-7695-2874-0/07 $25.00 © 2007 IEEE

5. References

[1] B. Ganter and R. Wille, "Applied Lattice Theory: Formal Concept: Analysis", Institute for Algebra, TU Dresden, Germany, 1997.

[2] B. Ralph, J. Kolodner and E. Plaza, "Representation in Case-Based Reasoning", *Knowledge Engineering Review*, Vol. 20, 2005, pp. 209-213.

[3] D. Belen, M. G. Antonio, P. G. Pablo and A. Pedro, "Formal Concept Analysis for Knowledge Refinement in Case Based Reasoning", *In Proc. of the 25th SGAI, Int. Conf. Innovative Techniques and Applications of Artificial Intelligence* (2005) 233-245.

[4] G. Coa, S. Shiu and X. Wang, "A Fuzzy-Rough Approach for Case Base Maintenance", *In Proc. of the 4th Int. Conf. on Case-Based Reasoning: Case-Based Reasoning Research and Development, Lecture Notes In Computer Science*; Vol. 2080, 2001, pp. 118-130.

[5] H. Wang and W. Zhang, "Relationships Between Concept Lattice and Rough Set", *Int. Conf., the 8th on Artificial Intelligence and Soft Computing* (ICAISC), 2006, pp. 538-547.

[6] K. P. Sankar and P. Mitra, "Case Generation Using Rough Sets with Fuzzy Representation", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 16, 2004, pp. 292-300.

[7] Kolodner, J, *Case-Based Reasoning*, Morgan Kaufmann, USA, 1993.

[8] L.A Zadeh, "Fuzzy sets", *Information control 8*, 1965, pp. 338-353.

[9] M. Banerjee, S. Mitra and K. P. Sankar, "Rough Fuzzy MLP: Knowledge Encoding and Classification", *IEEE Transactions on Neural networks*, Vol. 9, 1998, pp. 1203-1215.

[10] M. Salamo and E. Golobardes, "Rough Sets Reduction Techniques for Case-Based Reasoning", *Int. Conf., Case-Based Reasoning* (ICCBR), 2001, pp. 467-482.

[11] R. L. De Mantaras, D. McSherry, and et.cl, "Retrieval, Reuse, Revision, and Retention in Case Based Reasoning", *Knowledge Engineering Review*, Vol. 20, 2005, pp. 215-240.

[12] R. Wille, "Formal Concept Analysis as Mathematical Theory of Concepts and Concept Hierarchies", *Formal Concept Analysis: Foundations and Applications*, LNAI 3626, Berlin: Springer, 2005, pp. 1-33.

[13] T. Terano, K. Asai and M. Sugeno, *Fuzzy Systems Theory and its Applications*, Academic Press Limited, USA, 1992.

[14] U. Priss, "Formal Concept Analysis in Information Science", In: Cronin, Blaise (ed.), *Annual Review of Information Science and Technology*, Vol. 40, 2006, pp. 521-543.

[15] W. Roman, "Rough Set Methods in Feature Reduction and Classification", *Int. Appl. Math. Comput. Sci.*, Vol. 11, 2001, pp. 565 – 582.

[16] W. Ziarko, "Acquisition of Hierarchy-Structured Probabilistic Decision Tables and Rules from Data", *Knowledge Engineering, Expert Systems*, Vol. 20, 2003, pp. 305-610.

[17] Z. Pawlak, J. Peters and A. Skowron, "Approximating Functions using Rough Sets, Fuzzy Information", *IEEE Annual Meeting, Proc.* 2004 (NAFIPS '04), Vol. 2, 2004, pp. 785–790.

Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007)
0-7695-2874-0/07 $25.00 © 2007 IEEE

Third 2008 International Conference on Convergence and Hybrid Information Technology

A New Case-Based Classifier System Using Rough Formal Concept Analysis

Puntip Pattaraintakorn
Department of Mathematics and Computer Science, Faculty of Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand 10520
kppuntip@kmitl.ac.th

Veera Boonjing
Software Systems Engineering Laboratory, Department of Mathematics and Computer Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand 10520
kbveera@kmitl.ac.th

Jirapond Tadrat
Software Systems Engineering Laboratory, Department of Mathematics and Computer Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand 10520
s9062904@kmitl.ac.th

Abstract

Rough set theory and formal concept analysis were invented by Pawlak and Wille in the 1980s and have been applied successfully in several domains. In this paper, we propose a new case-based classifier system based on an integrated rough set theory and formal concept analysis technique.

We focus on the construction of a better knowledge base to produce the classification rules. Our system employs rough set theory to discover reduced cases. We then formulate a knowledge base with hierarchical structure by using formal concept analysis. The result is a concept lattice knowledge base embedded in our case-based classifier. We can generate classification rules from implications and subconcept-superconcept relations inside the obtained concept lattice. An illustrative example and a case study are provided to demonstrate the feasibility and applicability of our system. The advantages of our system are thus a better knowledge base for new problem classification and the flexibility to learn new rules.

1. Introduction

Case-based reasoning (CBR) is a method for problem solving that learns from prior experience and uses a knowledge base for future problem solving. When considering the tasks of a CBR system, they are often divided into classification and problem-solving CBR [11,15]. In this initial study, we focus on classification CBR. Classification CBR (case-based classifier) uses the previous cases as a reference point for new situations. Performance of CBR depends on the competence of learning and size of the case base. Traditional case base classifiers focus on matching process by using similarity and difference measures between the new situation and the previous ones [2,9,10]. The output of such matching process in the classification system provides a solution to the input problem, and thus does not require additional reasoning [10]. However, previous experiences should be added and considered for all cases to achieve knowledge in terms of classification rules. In order to hybridize these rule-based techniques and cases in CBR, the supporting structure of the knowledge base is required. Hence, we propose a new case-based classifier with more supportive knowledge base construction.

Our previous studies applied rough set theory (RST) in a CBR framework to obtain reduced cases [6,7]. We use RST in the present paper to enhance performance of knowledge construction. Nevertheless, stand alone RST cannot identify hierarchical structures and co-appearance of attributes for each case in the case base unless some heuristics are applied. Hence, we include formal concept analysis (FCA) in our proposed system. FCA can elicit the attribute dependency inside the case base in concept lattice form. FCA identifies the subconcept-superconcept relation in the concept lattice and prevents redundancy [6,7]. In this study, we apply the obtained concept lattice to derive classification rules for our case-based classifier system. These rules are then used to classify new problems into an appropriate class.

This article is organized as follows. In Section 2, we describe related works. Section 3 provides mathematical details of RST and FCA with discussion. Section 4 presents our proposed case-based classifier system followed by an illustrative example and case study in the subsequent section. Finally, we conclude the paper and list our future works.

2. Related works

In this section, we briefly review interesting previous works of FCA and RST in CBR systems.

978-0-7695-3407-7/08 $25.00 © 2008 IEEE
DOI 10.1109/ICCIT.2008.343

Practically, FCA and RST offer related and complementary approaches for data analysis.

Rule-based classifier systems were developed by several researchers [2,10,12,14]. Its classification accuracy depends significantly on coverage of the case base. Usually, the system requires a very large case base to obtain desirable accuracy. This large case base degrades the speed of case retrieval. Thus, the goal of case-based classifier systems is to build a minimal and sufficient knowledge base. There are two approaches to case-based classification problems: a similarity-based approach and a rule-based approach. In this study, we focus on the latter. A rule based classifier uses extracted IF-THEN rules (decision rule) in classification. Gupta et al. proposed an approach to integrate association and classification rules based on a concept lattice [2]. In [14], Xiong et al. built a fuzzy classification system and used genetic algorithms to achieve general rules from all possible rules. Current issues of rule-based classifiers are (i) develop a system with a sufficient number of rules and (ii) generate more suitable knowledge bases.

Pawlak introduced RST and developed several RST approaches in order to support a wide range of applications [19]. RST is a promising method to CBR systems because CBR involves large scale data that RST can be applied to efficiently. In [8], Wierzbicki derived the dependency rules by using the indiscernibility relation in RST to solve complex problems efficiently. Salamo et al. developed the case-based classifier system called BASTIAN based on RST [13]. They applied RST to weight and reduce cases in the case base and can retrieve the nearest similar case. RST was used to reduce the initial case base by several researchers. Sankar et al. [9] used fuzzy sets to represent cases to support similarity measure methods after they applied RST to reduce the case base. The advantage was that it supported numeric data very well. Following these successes of RST in CBR, we apply RST for case reduction in our case base classifier.

Belen et al. [4,5] used FCA to discover knowledge embedded in the case base. It is used to complement the knowledge already acquired by other techniques [5]. However, traditional FCA supports only binary data tables and real situations often include multi-value attributes. When FCA researchers faced this problem, they simply transformed the values of attributes to new binary attributes. This new formal context can be used to build rules with its implications. However, this process was found to be NP-hard [17]. To alleviate this problem, we apply RST to reduce the new formal context. Concept lattices generated by FCA were applied very successfully in several studies (cf. [17]). We thus use FCA to generate concept lattices as a new form of knowledge base.

3. Primitives of FCA and RST

3.1 Rough set theory

Mathematical rough set theory (RST) was introduced by Zdzislaw Pawlak [19]. Its major function is to automatically transform data into knowledge [6,7,13]. This section provides some theoretical concepts of RST taken from [19].

A data table contains columns, rows and entries that represent attributes, objects and attribute values, respectively. This data table can be viewed as an *information system* which is a pair $S = (U, A)$ where U is a finite non-empty set of N *cases* $\{x_1, x_2, \ldots, x_N\}$, called the *universe*, and A is the set of attributes in the data table such that a: $U \rightarrow V_a$, where V_a is the set of all values of a called the *domain* of a. We distinguish the information system to two disjoint classes of attributes. *Condition attributes* (C), describe properties of the problem, and *decision attribute* (D), describe the goal or solution The system will be called a *decision table* and will be denoted by $S = (U, C, D)$, where $C \cup D = A$. An equivalence relation, referred to as *indiscernibility relation*, $IND(P)$, is associated with any subset $P \subseteq A$. This relation is defined as:

$$IND(P) = \{(x, y) \in U \times U: a(x) = a(y), \forall a \in P\}.$$

It is the most important relation considered in rough set theory. The elements of U satisfying relation $IND(P)$ are called *indiscernible* by attributes from P. $U/IND(P)$ denotes the equivalence classes (partition) of $IND(P)$. In most of the problems considered in the literature, data contains vagueness and uncertainty. RST is very efficient to deal with this type of data by approximating the given data set. We can approximate X, using the information contained in R by constructing the *R-lower approximation* of X:

$$\underline{RX} = \bigcup \{Y \in U/\, IND(R) : Y \subseteq X\},$$

where $R \subseteq A$ and $X \subseteq U$.

We can define the *positive region* $POS_R(D)$, in the relation $IND(D)$ as $POS_R(D) = \bigcup \{\underline{RX} : X \in IND(D)\}$. Attribute $c \in R$ is a *dispensable attribute* in attribute subset $R \subset C$, if $POS_{R-\{c\}}(D) = POS_R(D)$, otherwise c is an *indispensable attribute*. Let $R' \subseteq R$. R' is a *reduct* R (*RED*), if $POS_{R'}(D) = POS_R(D)$. Intuitively, a reduct can be understood as the minimal attribute subset of the information system in which its classification ability is as same as the original attribute set. Obviously, there may be several reducts for an information system.

Next, the *core* of the information system S is defined as $CORE(S) = \bigcap RED(S)$ where $RED(S)$ is the reduct of information system S. From the above principles, RST is able to extract necessary condition attributes based on core and reduct.

3.2 Formal concept analysis

Formal concept analysis (FCA), invented by Wille [3,16,18], is a method for data analysis based on concept lattices. It is widely used for information science [18] to describe attributes of information represented in the hierarchical structure model. We are able to formulate the relationship of generalization and specialization among concepts through a data representation in FCA called *Hass diagram* [5].

The *formal context* is defined as a triple (G, M, I), (data table in RST) where there are two sets G (*objects*) and M (*attributes*), and a binary *incidence relation* $I \subseteq G \times M$. Attributes describe each object, $(g, m) \in I$ if the object g carries the attribute m (or m is a descriptor of the objects g). A representation of a formal context is called a *cross table*. With a general perspective, a concept represents a group of objects and is described by attributes and objects. The *extend* covers all objects belonging to the concept while the *intent* comprises all attributes shared by all those objects. With a set of objects $A \subseteq G$ and a set of attributes $B \subseteq M$, the derivation operators are defined as:

$$A' = \{m \in M \mid (\forall g \in A): (g, m) \in I\},$$
$$B' = \{g \in G \mid (\forall m \in B): (g, m) \in I\}.$$

In other words, a pair (A, B) where $A \subseteq G$ and $B \subseteq M$ is said to be a *formal concept* of (G, M, I) if $A' = B$ and $B' = A$. A and B are called *extent* (extension) and *intent* (intension) of the concept, respectively. The set of all formal concepts of a formal context (G,M,I) is denoted by $\beta(G,M,I)$. The most important structure on $\beta(G,M,I)$ is given by the *subconcept-superconcept ordered relation* denoted by $\leq$ defined as follows:

$$(A_1,B_1) \leq (A_2,B_2) \Leftrightarrow A_1 \subseteq A_2 \ (\Leftrightarrow B_2 \subseteq B_1).$$

$(\beta(G,M,I), \leq)$ is called the *concept lattice* of the context (G,M,I), for which the *infimum* and *supremum* are defined as

$$\mathrm{Inf}\beta(G,M,I) \equiv \bigwedge_{i \in I}(A_i,B_i) = [\bigcap_{i \in I}(A_i, (\bigcup_{i \in I} B_i)'')],$$
$$\mathrm{Sup}\beta(G,M,I) \equiv \bigvee_{i \in I}(A_i,B_i) = [\bigcup_{i \in I}(A_i, (\bigcap_{i \in I} B_i)'')],$$

where $\{(A_i, B_i) \mid i \in I\} \subseteq \beta(G, M, I)$.

In our proposed system, we build concept lattices from the case base in reduced and binary form. This concept lattice can provide classification rules by its implications. An implication between a set of condition attributes C and a decision attribute D is a pair of subsets of all attributes in an information system, denoted by $C \rightarrow D$. The set C is the premise of the implication and D is conclusion. $C \rightarrow D$ holds in a context (G,M,I) if each case that has all the attributes in C also has all the attributes in D. Thus, new classification rules can be generated more efficiently by this implication.

4. Our proposed case-based classifier

Our proposed case-based classifier system is divided into three main phases: *knowledge base* (Figure 1), *rule learning* (Figure 2), and *classification* (Figure 3). Phase 1 involves construction of a knowledge base to generate initial rules. Phase 2 performs rule learning to initial rules that can improve the performance of classification. Finally, Phase 3 classifies new problems using all of rules from Phases 1 and 2.

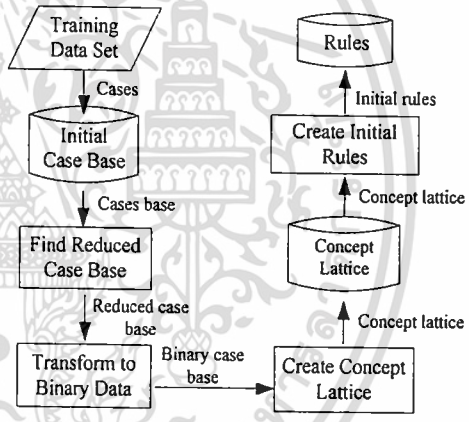


Figure 1. Phase 1: Knowledge base

In the knowledge base phase, we begin by using an initial case base to build a new structure of knowledge base. High dimensional data can cause inefficient time and space complexities, thus to obtain a reduced case base, we first apply RST to reduce irrelevant attributes and redundant cases. Next, we transform each attribute value in the reduced case base into new attributes. These new attribute values are represented in the binary relation form (e.g., Table 3, for more information the reader is referred to [6,7]). Next, we use RST once again to assure the sufficiently reduced number of cases. Afterwards, we use the FCA technique to construct a concept lattice. We then consider implications in the formulated concept lattice to build initial rules. In this phase we finally obtain a concept lattice and initial rules.

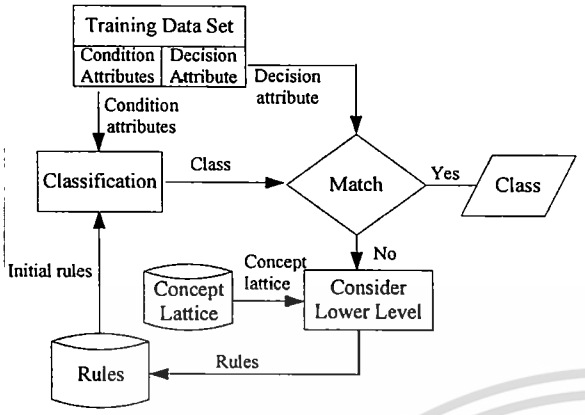


Figure 2. Phase 2: Rule learning

In Phase 2, to enhance the covering capability of rules, we evaluate initial rules with the training data set. If they can classify correctly, we output the obtained class. Otherwise, if these initial rules cannot classify the training data set, we generate more rules from the subconcept–superconcept relation stored in the concept lattice. We end this phase by updating new rules and repeat the processes. These rules are used to classify new problem more accurately.

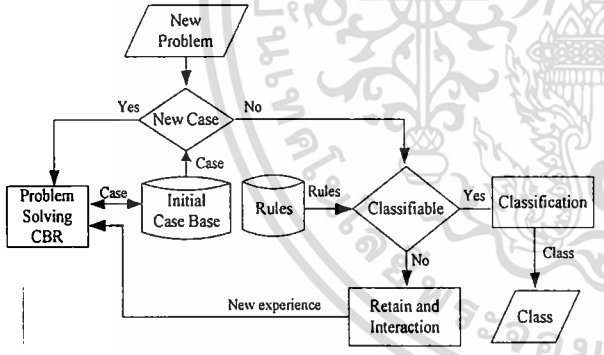


Figure 3. Phase 3: Classification

The last phase is classification, depicted in Figure 3. We use the acquired rules from the Phases 1 and 2 for classifying new problems. If it is not a new case, we will consider its classification ability by our existing rules. Its proper class will be provided as a result for classifiable problems. Otherwise, our system requires interaction with users in the retain and interaction processes. The new experience will then be added to the problem solving CBR module. On the other hand, if a new case is input, we will send it to the problem solving CBR. Phases 1 and 2 will be completed in our future works.

5. Illustrative example and case study

Table 1. An example of initial case base

ID	Weight	Door	Size	Cylinder	Class
1	Low	2	Compact	4	High
2	Low	4	Sub	6	Low
3	Med	4	Compact	4	High
4	High	2	Compact	6	Low
5	High	4	Compact	4	Low
6	Low	4	Compact	4	High
7	High	4	Sub	6	Low
8	Low	2	Sub	6	Low

Initially, we demonstrate details of our method in Phase 1 to the given case base in Table 1. Condition attributes are *Weight, Door, Size,* and *Cylinder*. The final column, decision attribute, is *Class*. We apply RST attribute reduction to this initial case base to reduce the size of the case base. We obtained the core attributes and reducts from our previous studies [6,7] as

core attribute = {*Weight*},
reducts = {*Weight, Size*}.

Table 2. A reduced case bases

ID	Weight	Size	Class
1	Low	Compact	High
2	Low	Sub	Low
3	Med	Compact	High
4	High	Compact	Low
7	High	Sub	Low

The new case base in Table 1 is reduced according to these reducts as shown in Table 2. Afterwards, we transform this reduced case base to have binary attributes by using FCA as shown in Table 3. Obviously, this new case base is expanded in both attributes and objects. Thus, we apply RST once again to reduce this new case base. Please note that Table 3 is in the reduced form already.

Table 3. A formal context

ID	Weight			Size		Class	
	Low	Med	High	Compact	Sub	Low	High
1	1	0	0	1	0	0	1
2	1	0	0	0	1	1	0
3	0	1	0	1	0	0	1
4	0	0	1	1	0	1	0
7	0	0	1	0	1	1	0

We use the FCA technique again here to create a formal concept (Table 4). From this formal concept, we extract the new relation in concept lattice form as in Figure 4. It depicts the knowledge base in hierarchical structure which is better than traditional case base (Tables 1, 4) in both retrieval and further rule learning processes. Each node represents a formal concept, each edge is the subconcept–superconcept relation.

Table 4. Formal concept from Table 3

$\beta(G,M,I)$: set of formal concept		
Formal concept	Extent Case No.	Intent
Top	1,2,3,4,7	∅
A	2,4,7	Class.Low
B	1,3,4	Size.Compact
C	2,1	Weight.Low
D	2,7	Class.Low,Size.Sub
E	4,7	Class.Low,Weight.High
F	3,1	Class.High, Size.Compact
G	2	Class.Low, Weight.Low, Size.Sub
H	1	Class.High, Size.Compact, Weight. Low
I	4	Class.Low,Weight.High,Size.Compact
J	3	Class.High, Size.Compact, Weight.Med
K	7	Class.Low,Weight.High,Size.Sub
Bottom	∅	Class.Low, Class.High, Weight.High, Weight.Med, Weight. Low, Size.Compact, Size.Sub

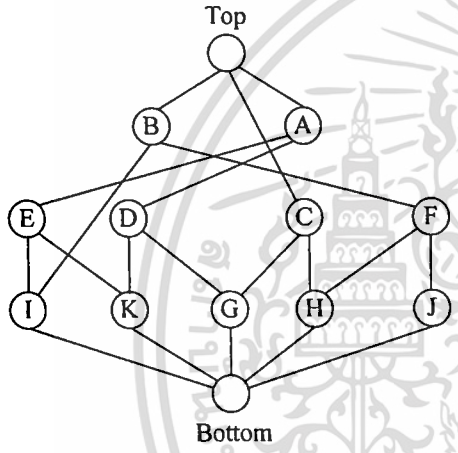


Figure 4. Knowledge base in the structure of concept lattice

The representation in Figure 4 gives another view of the knowledge base since it visualizes inherent structure existing in the given data clearly without expert knowledge.

We consider class nodes from top to bottom of this concept lattice and use implications to create rules. Then, we build the initial rules as shown in Table 5. The classification accuracy of the initial rules is 100%. Thus, we not only obtained the decision rules but also the concept lattice knowledge base.

Table 5. The initial rules

Rule order	Antecedent	Consequent
1	If (Size=Sub)	Then Class = Low
2	If (Weight=High)	Then Class = Low
3	If (Weight= Low and Size=Sub)	Then Class = Low
4	If (Weight= Low and Size= Compact)	Then Class = High
5	If (Weight= High and Size= Compact)	Then Class = Low
6	If (Weight= Med and Size= Compact)	Then Class = High
7	If (Weight= High and Size= Sub)	Then Class = Low

We also ran an experiment on a real-world machine learning problem from UCI repository: LENSES [1]. The goal is to predict whether people require soft, hard or no contact lenses. There are 4 condition attributes (1 tertiary) and 24 examples cover all cases. The documentation for this data set concluded that the correct number of rules to cover all examples is 9 (cf. [1]). The following are the portions of initial case base, formal context and a full obtained rule set, respectively.

Table 6. A case study: LENSES data set

age	spectacle	astigm-atic	tearProduc-tion	Class
young	myope	no	reduced	noContact
young	myope	no	normal	softContact
...	....	...	...	...
presbyopic	hypermetrope	yes	normal	noContact

Table 7. A formal context for LENSES data set

age. young	age.presbyopic	age. presbyopic	spect...acle. myope	...	Class. noContact	Class. softContact	Class. Hard Contact
1	0	0	1	...	1	0	0
1	0	0	1	...	0	1	0
...	...	...	...	...	...	...	...
0	0	1	0	...	0	0	1

Table 8. A full obtained rule set

Rule order	Rule
1	If (tearProduction=reduced) Then (Class= noContact)
2	If (astigmatic=no and spectacle=myope and age=presbyopic) Then (Class= noContact)
3	If (astigmatic=yes and spectacle=hypermetrope and age=pre-presbyopic) Then (Class= noContact)
4	If (astigmatic=yes and spectacle=hypermetrope and age=presbyopic) Then (Class= noContact)
5	If (tearProduction=normal and astigmatic=no and spectacle=hypermetrope) Then (Class= softContact)
6	If (tearProduction=normal and astigmatic=no and age=young) Then (Class= softContact)
7	If (tearProduction=normal and astigmatic=no and age=pre-presbyopic) Then (Class= softContact)
8	If (tearProduction=normal and astigmatic=yes and spectacle=myope) Then (Class= hardContact)
9	If (tearProduction=normal and astigmatic=yes and age=young) Then (Class= hardContact)

Due to space limitation, we do not depict the concept lattice knowledge base (used to derive the rules). Essentially, we create 9 initial rules successfully for this real-world data. The accuracy is 100% as in our example. However, the accuracy rate of the rules could be reduced proportionally to the size of the data set. Thus, the rule learning phase should be added to learn these obtained rules. The problem solving CBR should also be attached to our system to be able to perform both classification and problem solving from the knowledge base.

6. Conclusion

We propose a new CBR classifier framework based on FCA and RST. This integrated technique leads us to satisfactory knowledge base construction. The formulated knowledge base is in the structure of a concept lattice. Its implications provide classification rules efficiently where the time and space complexities are reduced.

Our system constructs a better knowledge base which best suits the inherent data structure for new problem classification and has the flexibility to learn new rules. In our complete research, a rule learning phase will be added to improve the classification ability. The problem solving CBR module will also be constructed based on our proposed knowledge base.

Acknowledgments

The authors are deeply grateful Commission on Higher Education, Thailand. We also would like to thank Faculty of Science grant and King Mongkut's Institute of Technology Ladkrabang research grant.

7. References

[1] A. Asuncion, D.J. Newman, UCI Machine Learning Repository [http://www.ics.uci.edu/~mlearn/MLRepository.html], Irvine, CA: University of California, School of Information and Computer Science, 2007.

[2] A. Gupta, N. Kumar, V. Bhatnager, "Incremental Classification Rules Based on Association Rules Using Formal Concept Analysis", *LNAI 3587*, Berlin: Springer, 2005, pp. 11-20.

[3] B. Ganter, R. Wille, "Applied Lattice Theory: Formal Concept Analysis", *Institute fur Algebra, TU Dresden*, Germany, 1997.

[4] D. Belen, A. Pedro, "Formal Concept Analysis as a support technique for CBR". *Int. J. in Knowledge-based Systems*, 2001, pp. 163-172.

[5] D. Belen, M.G. Antonio, P.G. Pablo, A. Pedro, "Formal Concept Analysis for Knowledge Refinement in Case Based Reasoning", *The 25th Int. Conf. on Innovative Techniques and Applications of Artificial Intelligence*, 2005, pp. 233-245.

[6] J. Tadrat, V. Boonjing, P. Pattaraintakorn, "A Framework for Using Rough Sets and Formal Concept Analysis in Case Based Reasoning", *The 2007 IEEE Int. Conf. on Information Reuse and Integration*, 2007, pp. 227-232.

[7] J. Tadrat, V. Boonjing, P. Pattaraintakorn, "A Hybrid Case Based Reasoning System Using Fuzzy-Rough Sets and Formal Concept Analysis", *The 4th Int. Conf. on Fuzzy Systems and Knowledge Discovery*, 2007, pp. 426-429.

[8] J. Wierzbicki, "Rough Set Approach to CBR", *Int. Conf. the 2nd of Rough Sets and Current Trends in Computing*, Canada, Berlin: Springer, 2001, pp. 503-510.

[9] K. Sankar Pal, P. Mitra, "Case Generation Using Rough Sets with Fuzzy Representation", *IEEE Transactions on Knowledge and Data Engineering, Vol. 16*, 2004, pp. 292-300.

[10] K.M. Luke, M.G. Kalyan, W.A. David, "Case-Based Collective Classification", *20th International FLAIRS Conference*, 2007, pp. 399-404.

[11] Kolodner, J., Case-Based Reasoning, 1st edn. Morgan Kaufmann, 1993, USA.

[12] L. Yan, S.S. Chi-Keung, K.P. Sankar, J.L. Nga-Kwok, "Rough Learning Vector Quantization Case Generation for CBR Classifiers", *The 10th Int. Conf. on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing*, 2005, pp. 128-138.

[13] M. Salamo, E. Golobardes, "Rough Sets Reduction Techniques for Case-Based Reasoning", *Int. Conf. on Case-Based Reasoning*, 2001, pp. 467-482.

[14] N. Xiong, L. Litz, H. Ressom, "Learning Premises of Fuzzy Rules for Knowledge Acquisition in Classification Problems", *Knowledge and Information Systems Vol. 4 Issue 1*, Springer-Verlag New York, 2002, pp. 96 – 111.

[15] R. Lopez De Mantaras, D. McSherry, D. Bridge, et al., "Retrieval, Reuse, Revision, and Retention in Case Based Reasoning", *J. of Knowledge Engineering Review, Vol. 20*, 2005, pp. 215-240.

[16] R. Wille, "Formal Concept Analysis as Mathematical Theory of Concepts and Concept Hierarchies", *Formal Concept Analysis: Foundations and Applications, LNAI 3626*, Berlin: Springer, 2005, pp. 1-33.

[17] S.O. Kuznetsov, "On Complexity of Computing the Duquenne-Guigues Basis", *J. Universal Computer Science Vol. 10(8)*, 2004, pp. 927-933.

[18] U. Priss, "Formal Concept Analysis in Information Science", *Annual Review of Information Science and Technology Vol. 40*, 2006, pp. 521-543.

[19] Z. Pawlak, "Rough Sets and Intelligent Data Analysis", *Int. J. Information Sciences Informatics and Computer Science Vol. 147*, 2002, pp. 1-12.

Building Classification Rules for Case-Based Classifier Using Fuzzy Sets and Formal Concept Analysis

Jirapond Tadrat
Software Systems Engineering Laboratory, Department of Mathematics and Computer Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand 10520
s9062904@kmitl.ac.th

Veera Boonjing
Software Systems Engineering Laboratory, Department of Mathematics and Computer Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand 10520
kbveera@kmitl.ac.th

Puntip Pattaraintakorn
Department of Mathematics and Computer Science, Faculty of Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand 10520
kppuntip@kmitl.ac.th

ABSTRACT
The focus of this paper is a construction of better knowledge base in case-based classifier system. Our knowledge base structure is based on concept lattice where rules are built from its subconcept-superconcept relation. Since the lattice can only be constructed from inputs with binary attributes, descriptive and numeric attributes must be transformed to binary attributes. In this paper, we propose the transformation of numeric attributes to descriptive attributes using fuzzy set theory. We experiment on benchmark data sets, Car and Iris, to determine the performance in term of number of rules used and classification precision. The results show that trend of accuracy is proportional to the size of learning inputs. The number of rules used is relatively small compared with size of training data. Our case-based classifier produces very promising results in practice and can classify the new problem more accurate than traditional classifiers.

Categories and Subject Descriptors
I.5.2 [**Design Methodology**]: Classifier design and evaluation.

General Terms
Algorithms, Design, Experimentation.

Keywords
Formal concept analysis, Concept lattice, Fuzzy sets, Case-based classifier, Knowledge acquisition.

1. INTRODUCTION
Case based reasoning (CBR) [2,10] is a method to problem solving that learns from prior experience and uses knowledge base for the future problem solving. A single case represents specific obtained knowledge, stored in the case base. CBR system that intends to classify new problems is referred to as a case-based classifier. In practice, case-based classifier uses the knowledge base from case base to determine an appropriate class for a new problem. To identify class of the new problem, the tradition case-based classifier focuses on similarity retrieval of previous cases to solve the new problem [7,9,13]. Nevertheless, the previous experiences should also be processed for all cases to achieve the knowledge. One of such techniques to achieve knowledge is the usage of rule-based. In order to hybridize the rule-based technique and cases in CBR, the supporting structure of the knowledge base is required.

Intuitively, rules represent general knowledge of domain whereas cases specify knowledge. This general knowledge gives more accurate results than specific knowledge for problem solving task. Thus, we propose to build new case-based classifier using formal concept analysis (FCA) in the general rules acquisition.

FCA [4,17,19] is a method for data analysis based on concept lattice. It is widely used for information science [19] to describe attributes and objects of information that can be represented in hierarchical structure. FCA provides relationship of generalization and specialization among concepts in concept lattice form. Representing a case base in concept lattice form is very useful. From a practical point of view, we can directly elicit the attribute dependency inside the case base. Furthermore, we can identifies the relation in concept lattice which gives redundancy prevention in the case base [4,17]. Thus, FCA will be used to construct knowledge base. We apply the obtained concept lattice from FCA to build rules for our case-based classifier. Next, these rules are used to classify new problem into an appropriate class.

Unfortunately, traditional FCA supports only binary relation and real situations often include multi-value attributes. The numeric attributes transformation in FCA technique also generates a great number of new attributes. Such new attributes lead to more space

CSTST 2008, October 27-31, 2008, Cergy-Pontoise, France.

and time requirements. They also affect qualities of formal concept and concept lattice. To avoid these troubles, we transform numeric attributes using fuzzy set theory (FST). FST is a mathematical concept proposed by Zadeh [11,15]. The usage of FST can change numeric attributes to more flexible degree and thus reduce size of the case base. The salient advantage of FST is to describe an elastic relation between attributes and objects. The result is new case base in which its values are in the interval [0,1]. We use this interval to predefine layer to support the FCA in knowledge base structure.

This article is organized as follows. In Section 2, we describe related works. Section 3 provides mathematical details of FCA and FST. Section 4 presents our proposed case-based classifier system follows by the experimental results in the next section. Finally, section 6 gives conclusion.

2. RELATED WORKS

In this section, we brief interesting previous works of traditional classifier systems follows by traditional classifier systems based on rule-based and case-based classifier system.

FST was applied successfully to support numeric attributes on classification problems. Xiong et al. [16] built fuzzy classification system and used genetic algorithms (GA) to achieve the general rules. These rules had average accuracy 96.70 % for cross-validation using training data 96.00% for Iris data set. For Cancer data set, the accuracy is 91.00 % for cross-validation using training data 85.40%. Their system required many training data set for achieving correctly classification. Fakhrahmad et al. [18] proposed a method for rule generation and rule weighting from numeric input data by using fuzzy sets. They used a learning mechanism to find weight of specific and general rules which can prevent overfitting. Their system can deal with high dimension data sets. Its performance is more effective in reducing the error rate of the classifier than those systems with traditional weighting metrics and C4.5. Pach et al. [6] used fuzzy association rules to enhance efficiency of classification. The associate rules often cause a superfluous number of rules. From these studies, FST support the numeric attributes, which provide its interpretability. In other word, the performance of classification model is not only considerable its interpretability, but extend to its accuracy. The addition of association rules to classification rules is an interpretation. Gupta et al. [3] proposed an approach to integrate association and classification rules based on concept lattice. The advantage of concept lattice in this system was its ability to increase the incoming input data.

Furthermore, the lattice is successfully used in classifier systems. In [12], Sahami used training data to build Galois lattice. This lattice was used to induce classification rules and support descriptive attributes or binary data. In addition, users are able to specify and configure the system to induce classification rules. Wang et al. [20] proposed Classification Rule Acquisition Based on Extended Concept Lattice (CAECL) that improved their previous work called LACS. The authors employed rule novelty to prune rule sets. The comparisons of CAECL and LACS for Iris and Car data sets were given as shown in Table 1.

Next, we consider interesting case-based classifiers. Traditional similar-based classifier systems were developed by several researchers. The goal of similar-based classifier is to retrieve previous cases to solve the new problem. Jurisica et al. [7] proposed the case-based classifier system called *TA3*. They used context-based similarity to retrieve relevant cases and then used them for the classification task. Salamo et al. [13,14] developed the case-based classifier system called BASTIAN with three methods: the original method (without weight), the weighting rough set method, and the sample correlation method. They applied RST for weighting method and reducing attributes in the case base. In addition, their system used sample correlation to compute weights. Their system represents previous case with attribute-value description and its solution. The authors retrieved solution using different similarity metrics such as *Minkowsky's metric*, *Clark's distance* and the *Cosine distance*. Their experiment results show that performance of weight methods is better than original method as shown in Table 2.

Table 1. The results for Iris and Car data sets[1] from [20].

Algorithm	Car		Iris	
	Accuracy (%)	Number of rules	Accuracy (%)	Number of rules
LACS	64.00	9.00	90.70	10.00
CAECL	64.00	6.00	89.3	12.00

Table 2. The accuracy mean for the Iris data set[2] from [13,14]

Training (%)	Methods		
	Without Weight	Rough sets and Weight	Sample Correlation
40	96.22	96.00	96.22
60	95.33	95.50	96.16
70	95.11	95.33	95.77
80	97.00	97.00	97.33
90	96.66	96.66	97.33

FCA also was applied to support case-based classifier. Belen et al. [5] used FCA as a complementary technique to enrich the domain taxonomy which provided an alternative organization of case base. Luke et al. [9] introduced case-based collective classification. Collective classification is a methodology that simultaneously classifies cases which may be interrelated. The author used k-nearest neighbor (k-NN) rule for case-based classification. Moreover, our initial study [8] proposed a

[1] They used 50% data for training and 50% data for testing.

[2] They used 10-fold cross-validation.

framework for constructing knowledge base in CBR system based on rough sets, FST and FCA. The advantages are alleviation of overfitting problem, reduce cases and elicit attribute dependency knowledge base. However, usage of rough sets in our previous work leads to slowly speed of our system. Thus, in this paper we drop the rough sets technique to achieve the faster speed.

3. PRELIMINARIES

3.1 Formal Concept Analysis

Definition 1. [17] The *formal context* is a triple *(G, M, I)*, where *G* denotes *cases*, *M* denotes *attributes*, and $I \subseteq G \times M$ denotes a binary *incidence relation* where $(g,m) \in I$ if *m* is a descriptor of the objects *g*.

Definition 2. [4] A pair (A,B) is a *formal concept* of *(G, M, I)* if and only if $A \subseteq G$, $B \subseteq M$, $A' = B$ and $A = B'$

when $A' = \{m \in M \mid (\forall g \in A): (g,m) \in I\}$,
$B' = \{g \in G \mid (\forall m \in B): (g,m) \in I\}$.

The set *A* is called the *extent* and the set *B* is called the *intent* of concept.

Each formal concept includes a pair of the extent and the intent. The extent consists of all cases that have intersection of attributes and the intent consists of all attributes corresponding to the extent.

Definition 3. [4] Let $\beta(G,M,I)$ be the set of all formal concept in formal context *(G,M,I)*. The concepts of given context are naturally ordered by the *subconcept-superconcept ordered relation* denoted by $\leq$ and is defined as follows:

$$(A_1,B_1) \leq (A_2,B_2) \Leftrightarrow A_1 \subseteq A_2 \ (\Leftrightarrow B_2 \subseteq B_1).$$

Definition 4. [17] Let *(G,M,I)* be a formal context, then $(\beta(G,M,I), \leq)$ is a complete lattice called *concept lattice* of the context *(G,M,I)*, for which infimum and supremum are defined as:

$$\text{Inf}\beta(G,M,I) \equiv \bigwedge_{\alpha}(A_\alpha, B_\alpha) = [\bigcap_{\alpha}(A_\alpha, (\bigcup_{\alpha} B_\alpha)'')],$$

$$\text{Sup}\beta(G,M,I) \equiv \bigvee_{\alpha}(A_\alpha, B_\alpha) = [\bigcup_{\alpha}(A_\alpha, (\bigcap_{\alpha} B_\alpha)'')].$$

In our proposed system we build concept lattice from initial case base. This concept lattice can provide classification rules from subconcept-superconcept ordered relations.

3.2 Fuzzy Set Theory

Fuzzy set theory is a mathematical knowledge representation based on degrees of membership (degree of truth) rather than crisp membership of binary logic [7,15]. It uses the continuum of logic value between 0 (completely false) and 1 (completely true). We use fuzzy sets to transform numeric attributes into continuum of logic values.

Definition 5. [15] Let *F* be numeric attributes which defined by $F = \{A_1, A_2, ..., A_k\}$ where A_k is *k*-th numeric attributes in a case base. The *membership function* of set A_i is defined by $\mu_{A_i}(v): V_i \rightarrow [0,1]$, where value of $\mu_{A_i}(v)$ is 1 if *v* is totally in A_i, 0 if *v* is not in A_i and $0 < \mu_{A_i}(v) < 1$ if *v* is partly in A_i.

In this paper, we use the triangular membership function in Figure 1 to transform numeric attribute to be new linguistic variables. We set linguistic values to be {MIN, MDM, MAX} which refer to minimum attribute value, medium attribute value, and maximum attribute value, respectively.

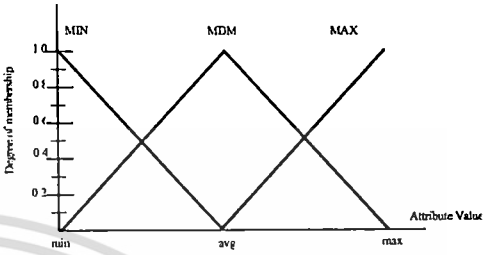


Figure 1. A triangular membership function.

4. OUR PROPOSED CASE-BASED CLASSIFIER SYSTEM

Figure 2 illustrates our proposed case-based classifier system architecture. It is divided to three main phases: *knowledge base* (Figure 3(a)), *rule learning* (Figure 3(b)), and *classification* (Figure 3(c)). Phase 1 produces new structure of knowledge base for achieving initial rules. Phase 2 uses initial rules from Phase 1 to enhance performance of classification with rule learning. Finally, Phase 3 classifies new problem using all of rules from Phases 1 and 2. In addition, this phase also suggests the solution for the new problem and reorganization for retaining by user.

Phase 1: Knowledge base.

The detail of this phase is shown in Figure 3(a). We begin with using an initial case base to build a new structure of knowledge base. FCA and FST are used to transform descriptive and numeric attributes, respectively. The result is that the new attributes consist of value in the interval [0,1]. However, traditional lattice can only be constructed from input with binary attributes. Thus, we require the formal context like to binary. Predefined layers are provided to transform new input to traditional formal context to support concept lattice construction. We predefine 4 layers[3] : Layer 1 for the value 1, Layer 2 for the values in [0.7,1), Layer 3 for the values in (0.5,0.7) and Layer 4 for the values in (0,0.5]. These predefined layers are defined like to traditional formal context to transform formal concept (Definition 2) in concept lattices creation. Afterwards, we create 4 concept lattices (Definition 4) then we use subconcept-superconcept relation referring to (Definition 3) to achieve initial rules from its relation inside each concept lattice.

[3] These layers can be defined with other intervals.

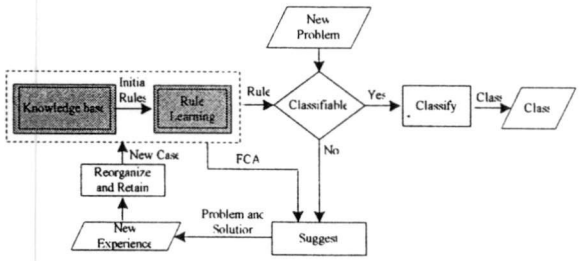


Figure 2. Our proposed case-based classifier system.

Table 3. An example of initial case base

Case	Type	Trans	Age	Price	Class
C1	Education	Car	20	120	Summer
C2	Active	Car	45	350	Winter
C3	Education	Train	50	280	Summer
C4	Education	Plane	15	200	Winter
C5	Education	Plane	25	450	Summer
C6	Education	Car	30	550	Winter
C7	Active	Plane	35	330	Winter

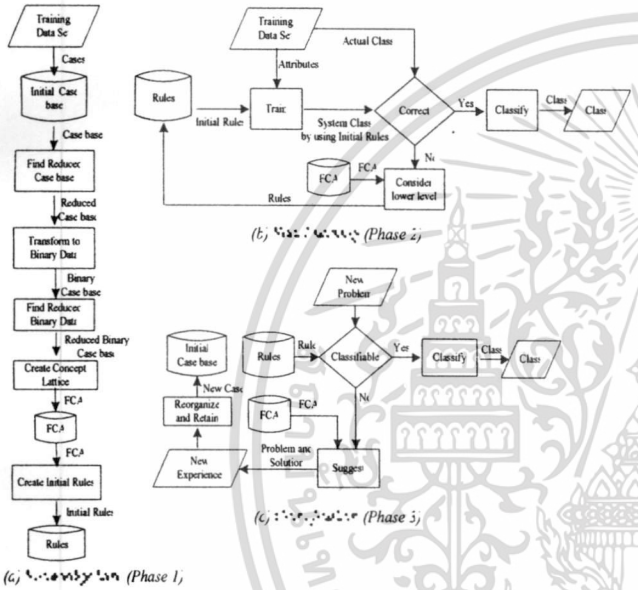
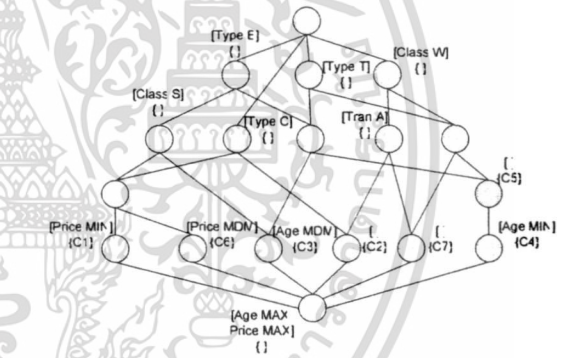


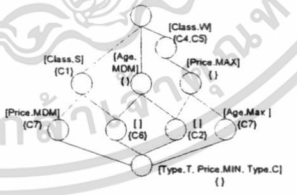
Figure 3. Three phases of case-based classifier system

Table 4. Transformed relation in new case base

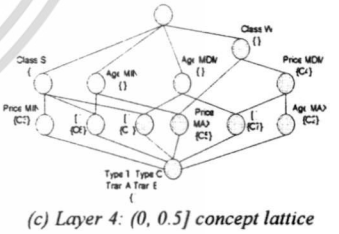
Case	Type		Tran			Age			Price			Class	
	E	A	C	T	P	MIN	MDM	MAX	MIN	MDM	MAX	S	W
C1	1	0	1	0	0	0.70	0.30	0	1	0	0	1	0
C2	0	1	1	0	0	0	0.73	0.27	0	0.11	0.89	0	1
C3	1	0	0	1	0	0	1	0	0.22	0.78	0	1	0
C4	1	0	0	0	1	1	0	0	0.61	0.39	0	0	1
C5	1	0	0	0	1	0.39	0.61	0	0	0.55	0.45	0	1
C6	1	0	1	0	0	0.09	0.91	0	0	1	0	1	0
C7	0	1	0	0	1	0	0.19	0.81	0	0.02	0.98	0	1



(a) Layer 1: [1] concept lattice



(b) Layer 2: [0.7,1) concept lattice



(c) Layer 4: (0, 0.5] concept lattice

Figure 4. Concept lattices generated from Table 4

For example, Table 3 is an initial case base. We use {E,A} for *Education* and *Active* of {Type}, {C,T,P} for *Car*, *Train* and *Plane* of {Trans}, and {S,W} for *Summer* and *Winter* of {Class}. FCA is used to transform descriptive attributes {Trans, Type, Class} whereas FST is used to transform numeric attributes {Age, Price} where triangular membership function in Figure 1 is used to obtain membership values. The result of attributes transformation is shown in Table 4. We use four predefined relationship intervals ((0, 0.5], (0.5, 0.7), [0.7,1), and [1]) over the linguistic variables {Age, Price} in Table 2 to create four concept lattices as shown in Figure 4.

A portion of our generated lattice as in Figure 4. Each node represents a formal concept of like to formal context (Table 4). Each edge represents subconcept-superconcept relation. The more detail of concept lattice creation can see in [8]. We consider class nodes from top to bottom of concept lattice. For instance, the rules acquisition from Figure 4(a) derive from subconcept-superconcept relation such as If *Tran=A* then *Class=W*, If *Age=MDM* then *Class=S*, etc. This phase provide the set of rules that derive from each class of concept lattice.

Phase 2: Rule learning.

We use training data set to achieve general rules to classify unknown class data in this phase as shown in Figure 3(b). The goal of this phase is to acquire general rules. Intuitively, the *general rules* are the reduced number of rules that can use to classify the biggest number of incoming cases in case based. The summarily process of rule learning is shown as follow:

Step1: Using training data set for building rules from each concept lattice. This step uses the set of rules from knowledge base to training data set for classification testing.

Step2: Evaluating set of rules. The co-appearances and subconcept–superconcept relation of concept lattice are considered for evaluating general rules acquisition. If the rules from *phase 1* cannot decide the actual class of training data (or misclassification), we then consider lower level of concept lattice and next rules. We will obtain the general rules that correctly classify for training test.

Step3: Collecting general rules. The rules obtained from above evaluating set of rules are collected for classifying new cases.

For example, Figure 4 represents each layer of concept lattice. Firstly, we find the set of rules considering Figure 4(a). This new set of rules is derived from subconcept–superconcept relation of considered class node in Phase 1. We use the set of rules from Phase 1 to evaluate training data in order to correctly classify 100%. Otherwise, we consider the lower nodes to build the new rules such as If *Price=MIN* then *Class =S* or If *Price=MDM* then *Class =S*, which will be added to general rules.

Phase 3: Classification.
We use general rules obtained from Phase 2 to classify new problems as shown in Figure 3(c). If a new problem is an unseen case and cannot decide its class, it will be stored into initial case base for using in the future. Thus, incoming cases interact with users to obtain their classes in initial case base. They are then added to case base as rule update. In contrast, if a new problem is a seen case, the general rules are applied to classify an appropriate class. Thus, we achieve the solution for new problem. However, if the new problem cannot use the general rule for classifying class label, the new problem will be suggested and retained into initial case base. These new problems, new experience, shall be retained to use in the future while the general rules are updated (when reorganizing knowledge of an existing case base).

5. EXPERIMENT RESULTS

To illustrate the applicability of our processed system, we use our system to experiment on the benchmark data sets: Car and Iris from the UCI repository [1]. The Car data set contains 1,728 samples described by 6 attributes (descriptive attributes). The Iris data set contains 150 samples and described by 4 attributes (numeric attributes). We randomly divide each data set into two sets: training and test sets. We experiment on different proportion of these sets e.g., 10% of the data set for the training set and the rest (90%) for the test set, 20 % for the training set and the rest (80%) for the test set and so on.

The results of our experiments on each data set in terms of classification accuracy and the percentage of rules used on test sets are shown in Table 5. The average accuracy of Car data is 91.89 and the average accuracy of Iris data is 94.66. We obtained the highest accuracy for Car and Iris data sets as 99.13 and 98.00, respectively. These results are comparable to the results reported in literature (cf. [13].[14],[20]).

Figure 5 shows classification accuracy of our classifier on Car and Iris data sets with different proportions of training sets. We found that Car data set (with descriptive attributes) has the trend of accuracy increase better than Iris data set (with numeric attributes). The reason is that the main structure of our system is constructed from FCA, thus the descriptive attributes will be supported by this structure better than numeric attributes. Furthermore, the percentage of rules used by our proposed system is relatively small compared with the size of training data. From the results of our experiments, when we use all cases (or previous experience) to achieve the knowledge or general rules, our system can correctly solve the new problem corresponding to Table 5 and Figure 5.

Table 5. The results of our proposed case-based classifier on Car and Iris data sets

Training (%)	Car		Iris	
	Accuracy (%)	Percentage of rules	Accuracy (%)	Percentage of rules
10	82.75	5.43	86.66	14.00
20	84.37	8.68	91.33	17.33
30	87.73	9.31	94.00	22.00
40	90.68	12.03	94.66	20.66
50	92.93	13.83	95.33	20.66
60	94.79	15.21	96.66	19.33
70	96.64	16.20	97.33	24.66
80	97.97	16.95	98.00	25.33
90	99.13	15.74	98.00	24.00

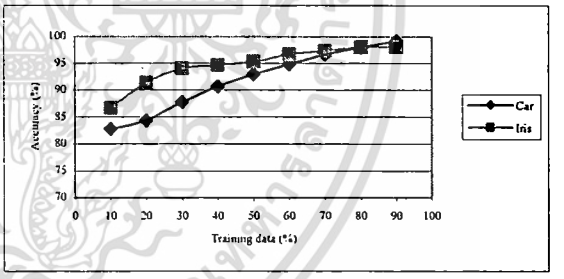


Figure 5. The trend of the result for our classification

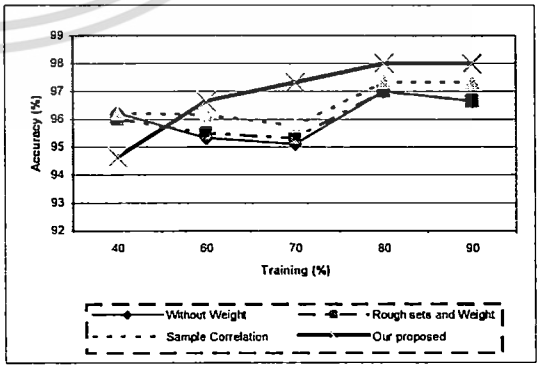


Figure 6. The comparison of our system and other methods

We further compare accuracy of our results in Table 5 with the other techniques (cf. [13,14] in Table 2) as shown in Figure 6. This figure shows that our proposed system outperforms other methods in classification accuracy at the same percentage of training data. Furthermore, accuracy of our results with 50% training data in Table 5 is better than of other methods (cf. [20]) shown in Table 1 with the same percentage of training data.

6. CONCLUSION

We propose a new case-based classifier system to support suitable knowledge base construction using formal concept analysis. The suitable knowledge structure leads into the simply rule acquisition. We use FCA as an automatic technique to elicit attributes dependency knowledge and create hierarchical structure of a case. The transformation of multi-value attributes lead to new large size attributes in case base. We solved this problem and changed numeric attributes to more flexible degree by FST. Transformation results were the reduced case base which better than stand alone FCA. In addition, we set predefined layer formal concept that provide degree of relationships to support FCA technique. For rule acquisition, we consider the class attributes from the subconcept–superconcept relation inside concept lattice. We performed our system with the benchmark data sets, successfully. The results show that trend of accuracy increased when increasing learning inputs and number of rules used is relatively small compared with the size of training data. Our case-based classifier produce very promising results in practice and can classify the new problem more accurate than traditional classifier systems.

ACKNOWLEDGMENTS

The authors are deeply grateful Commission on Higher Education, Thailand. We also would like to thank National Research Council of Thailand.

7. REFERENCES

[1] A. Asuncion, D.J. Newman, UCI Machine Learning Repository http://www.ics.uci.edu/~mlearn/MLRepository.html], Irvine, CA: University of California, School of Information and Computer Science, 2007.

[2] A. Aamodt, E. Plaza, "Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches", *J. AI Communication*, Vol. 7, 1994, pp.39-59.

[3] A. Gupta, N. Kumar, and V. Bhatnager, "Incremental classification rules based on association rules using formal concept analysis", *LNAI 3587*, Berlin: Springer, 2005, pp.11-20.

[4] B. Ganter and R. Wille, "Applied Lattice Theory: Formal Concept: Analysis", *Institute for Algebra, TU Dresden, Germany*, 1997.

[5] D. Belen and A. Pedro, "Formal Concept Analysis as a support technique for CBR". *Int. J. in Knowledge-based Systems*, 2001, pp. 163-172.

[6] F.P. Pach, A. Gyenesei and I. Abonyi, "Compact fuzzy association rule-based classifier", *Expert systems with Application*, Vol.34, 2008, pp.2406-2416.

[7] I. Jurisica and J. Glasgow, "Case-based classification using similarity-based retrieval", *8th IEEE International Conference on Tools with Artificial Intelligence*, Toulouse, France, 1996, pp.1-10.

[8] J. Tadrat, V. Boonjing, P. Pattaraintakorn, "A Hybrid Case Based Reasoning System Using Fuzzy-Rough Sets and Formal Concept Analysis". *The 4th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD'07)*, Haikou, China, Aug 24-27, 2007, pp.426-429.

[9] K.M. Luke, M.G. Kalyan, and W.A. David, "Case-based collective classification", *20th International FLAIRS Conference (FLAIRS-20)*, 2007, pp.399-404.

[10] Kolodner, J., Case-Based Reasoning, 1st edn. Morgan Kaufmann, 1993, USA.

[11] L.A Zadeh, "Fuzzy sets", *Information control 8*, 1965, pp.338-353.

[12] M. Sahami, "Learning classification rules using lattices", *Proceedings of the Eighth European Conference on Machine Learning (ECML-95)*, Springer-Verlag, Berlin, Germany, 1995, pp 343-346.

[13] M. Salamo, and E. Golobardes, "BASTIAN : incorporating the rough sets theory into a case-based classifier system", I*n III Congres Catala d'Intelligéncia Artificial (CCIA'00)*, Barcelona, Spain, 2000, pp. 1-10.

[14] M. Salamo, E Golobardes, "Weighting Methods for a Case-Based Classifier System", *Proc. of the IEEE Learning'00*, 2000.

[15] Negnevitsky, M.: Artificial Intelligence a Guide to Intelligent Systems, 2nd end., Addison-Wesley (2005).

[16] N. Xiong, L. Litz and H. Ressom, "Learning premises of fuzzy rules for knowledge acquisition in classification problems", *Knowledge and Information Systems*, Vol.4(1), Springer-Verlag New York, NY, USA, 2002, pp.96 – 111.

[17] R.Wille, "Formal Concept Analysis as Mathematical Theory of Concepts and Concept Hierarchies", *Formal Concept Analysis: Foundations and Applications, LNAI 3626, Berlin: Springer*, 2005, pp.1-33.

[18] S.M. Fakhrahmd, A. Zare, and M.Z. Jahromi, "Constructing accurate fuzzy rule-based classification system using Apriori principles and rule-weighting", *LNCS 4881*, Berlin: Springer, 2007, pp. 547-556.

[19] U Priss, "Formal concept analysis in information science", *Annual Review of Information Science and Technology*, Vol.40, 2006, pp. 521-543.

[20] Y. Wang and L. Ming, "Classification rule acquisition based on extended concept lattice", *LNCS 4688*, Berlin: Springer, 2007, pp. 571-578.

BIOGRAPHY

PERSONNEL INFORMATION

Thai Name: นางสาวจิราภรณ์ ทัดรัตน์

English Name: Miss. Jirapond Tadrat

Date of Birth: 3 February 1979

Permanent Address: 5 Moo 5, Nanglhong Sub-District, Chauat District, Nakornsithammarat, 80180, Thailand.

Telephone: (+66) 085-5789139

E-mail: s4145205@hotmail.com, jirapond03@gmail.com

EDUCATION

2005 – 2011 **Doctor of Philosophy in Computer Science.**
Department of Computer Science, Faculty of Science,
King Mongkut's Institute of Technology Ladkrabang,
Bangkok, Thailand 10520.

2002 – 2005 **Master Degree in Computer Science.**
Department of Computer Science, Faculty of Science,
King Mongkut's Institute of Technology Ladkrabang,
Bangkok, Thailand 10520.

1998 – 2002 **Bachelor Degree in Applied Mathematic.**
Department of Mathematics and Computer Science,
Faculty of Science and Tecnology,
Prince of Songkla University, Pattani Campus,
Pattani, 94000, Thailand.

1998 – 1992 **Hischool in Science-Math Programme.**
Chauat Hischool, Chauat District,
Nakornsithammarat, 80180, Thailand.