An Ontology-based Similarity Measurement for Problem-based Case Reasoning

Adela Lau[a,*], Eric.Tsui[b], WB Lee[c]

[a,*]School of Nursing[1]
[b,c]Department of Industrial and Systems Engineering

The Hong Kong Polytechnic University,
Hung Hom, Kowloon, Hong Kong, China

[a,*]Phone: 852+27664026 Fax: 852+23649665 E-mail: Adela.Lau@inet.polyu.edu.hk
[b]Phone: 852+27666609 Fax: 852+23625267 E-mail: Eric.Tsui@polyu.edu.hk
[c]Phone: 852+27666594 Fax: 852+23625267 E-mail: wb.lee@polyu.edu.hk

Keywords: knowledge retrieval, ontology based similarity measurement, problem-driven case

Abstract.

Traditional case-based reasoning uses a table/frame or scenario to represent a case. It assumed that similar input/event results in similar output/event state. However, similar cases may not have similar output/event states since problem solver may have different way to break down the problem. Thus, authors previously proposed problem-based case reasoning to overcome the limitation of the traditional approaches and used clustered ontology to represent the problem spaces of a case. However, synonym problem causes the mismatch of similar sub-problems of historical cases for new case. Thus, this paper proposed ontology-based similarity measurement to retrieve the similar sub-problems that overcomes the synonym problems on case retrieval. The recall and precise of ontology-based similarity measurement were higher than that of the traditional similarity measurement.

Introduction

Traditional knowledge representation methods of case-based reasoning represent the case by basing it on database tables, frames or scenarios (Fong et al., 1999; Meacham et al., 1989; Bo et al., 2003; Bergmann et al., 2006). The problem with a database table or frame approach is its limited storage and the fact that it is unable to represent events characterized by path dependence and context sensitivity (Bo et al., 2003). Scenario-based case representation, that represents the event states and the links between them, has solved this problem (Dutta et al., 1997). However, the knowledge derived from the retrieved cases can only be applied to event-driven cases but not to problem-driven cases. The problem solving steps cannot be triggered. Most importantly, each problem solver has their own mental model for breaking problems into sub-problems whereby they identify the situation, information and techniques to be used for solving the problems. Therefore, authors previously proposed the clustered ontology approach to represent the

[1]Part of the research work of this paper was done when the first author was working at Industrial and Systems Engineering Department of the Hong Kong Polytechnic University.

semantic meaning of a case (Lau et al., 2006). The problem was broken down into sub-problems and was used the ontology to represent the sub-problems and its corresponding situations, information and techniques, and these ontology representation units was clustered to represent the whole case. Problem solver retrieved and combined the knowledge of the historical cases by co-relating ontology representation units from different similar cases, which had similar sub-problem and situation.

The Cognitive Processes in Problem Solving

Problem solving varies in its external factors, including problem type and problem representation, and internal factors such as internal characteristics of the problem solver (Smith, 1991). Well-structured and simple problems can be solved with regular rules and principles. They have knowable and comprehensible solutions where the relationship between decision choices and all problem states is known or probabilistic (Avramenko 2006; Han , 2005; Meacham et al., 1989). Ill-structured and complex problems possess multiple solutions, solution paths, or no solution at all. An ill structured problem possesses multiple criteria for evaluating solutions so it is uncertain which concepts, rules, and principles are necessary for its solution and how they should be organized. It is often necessary for problem solvers to make judgments and express personal opinions or beliefs about the problem; so ill-structured problems are uniquely human and interpersonal activities (Wood, 1983). Therefore, the frame-based or scenario-based case representation is only suitable for well-structured problem solving since the rules and principle of problem solving are well-defined that means the similar cases that retrieved based on it input or states can be applied to new problem.

Solving ill-structured problems depends on the personal judgment of the problem solver. Problem solver goes through the cognitive process of encoding sub-problems in working memory; searching long term memory for algorithms and heuristics; executing the appropriate algorithms or heuristics and the new sub-problem state with its goal; identifying differences between the current state and the goal state; and selecting operations that reduce these differences (Fukumoto, 2007; Sohn, 2005; Zhang, 1994) for problem solving. The acquisition process of problem solving skill depends on how the similar problem solving skill are represented to and perceived by the problem solver. Since frame-based or scenario-based case representation is input or state-driven, it can only retrieve the cases with similar inputs or states and with static problem solving steps. It is not suitable for ill-structured problem representation since the problem solving steps varies in different cases and it depends on how problem solver solves the problem.

Thus, previously, authors proposed the problem-driven case representation to capture the problem solving steps of problem solvers. Problem-driven case representation breaks the problem into a series of well-defined sub-problems and captures the problem solving skill in terms of situation, information and techniques. It represents the problem solving processes characterized by the cognitive processes of human brain and helps problem solver learning to solve new problem. Problem solver can base on its own judgment to break down the sub-problems and search the similar sub-problems solutions from different cases.

Ontology for Semantic Meaning Representation and Retrieval

As mentioned before, the cognitive processes in problem solving are remembering, understanding, analyzing, applying, evaluating, and creating. To computerize the cognitive processes, problem solving steps are captured and stored in knowledge base and then be retrieved based on cognitive model of problem solver. However, how does the computer understand the semantic meaning of the document in order to analyze, evaluate, and apply the problem solving skills of the cases to new problem? Since the same concept terms may have different meaning in different document or the same concept may use different terms to represent, this synonym problem may cause the mismatching of similar cases.

Ontology provides a formal semantic representation of the objects for case representation. The ontology notions or specification can be classified as controlled vocabulary (a finite list of terms), glossary (a list of terms and meanings) and a thesaurus (a list of synonym relations). The design of ontology can be based on classificatory knowledge or generic knowledge (Noh, 2000). Classificatory knowledge is used to organize words into groups that share many properties. Generic knowledge consists of features about each group. It allows user to construct concepts explicitly, build a hierarchical organization of them, and define the relations between the concepts. Hence, ontology can better represent the semantic meaning of a case and overcomes the synonym problem.

Previous Work Done by Authors

Authors developed an ontology taxonomy database (Lau et al., 2006), which acts as communication media between people and systems by providing a shared and common understanding of knowledge that improves mapping of the cases, adopts both techniques to group the semantic meaning of the terms or objects using controlled vocabulary, term relationships, and hierarchical subclass relationships between classes, properties and constraint specifications in classes. A new approach of clustered ontology was used to capture and represent the semantic meaning of a case (Lau et al., 2006). The problem was broken down into sub-problems and ontology was used to represent the sub-problems and their corresponding situations, information and techniques, then these ontologically representative units were clustered to represent the whole case. Thus, problem solvers can use its mental model of problem solving to retrieve and combine the knowledge of the historical cases by co-relating the ontologically representative units from various similar cases that have similar sub-problems and situations.

For example (see fig. 2), user will identify the context, problem domain, information to be collected, techniques to be used (see Level 0 of the figure) when solving the problem of a case. When processing task 1 in level 0, user will identify the people, business nature and place involved in this case. That means the user will decompose the task into sub-tasks and then sub-sub-tasks and so on during the problem solving process.

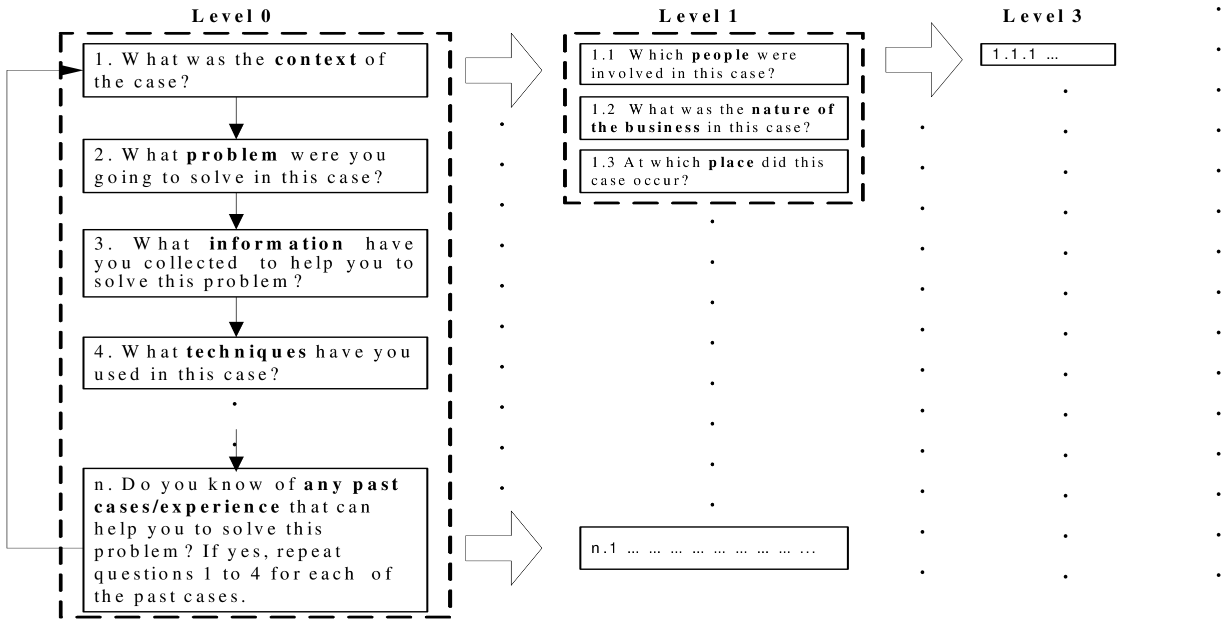

Fig. 2: Task decomposition for Problem Solving.

After the human mental process of problem solving was captured, the next step was to decide how to index and represent the knowledge dynamically and structurally so that it can be applied to other problem domains in other cases. The ontological approach was used to represent and relate the objects in the case in a hierarchically structured manner so that the semantic meaning of the case can be represented (see fig. 3). For example,

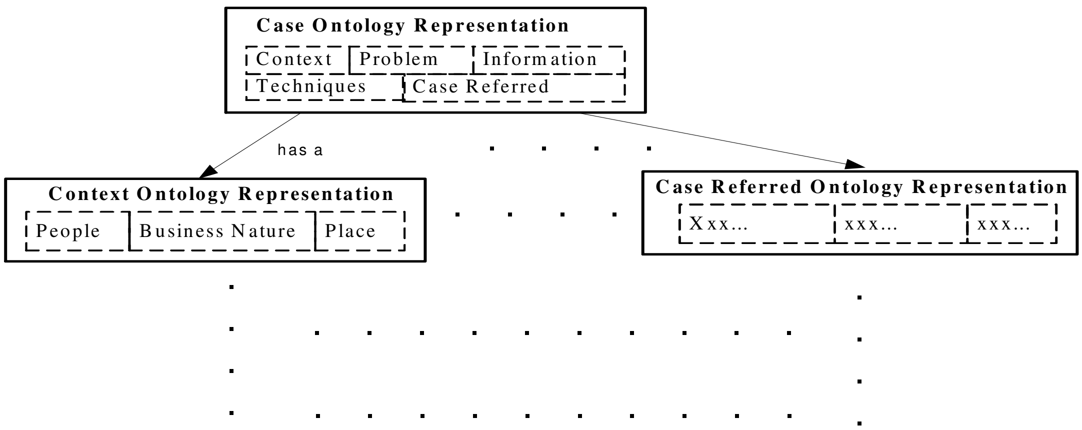

Fig. 3 Semantic Meaning of the Case using Ontological Representation

Ontological taxonomies were built in order to classify the context, problems, information, and techniques in different subject domains. See example for the e-business development ontology taxonomy (see fig. 4). In this way, each case can be indexed by context type, problem type, information type and techniques type.

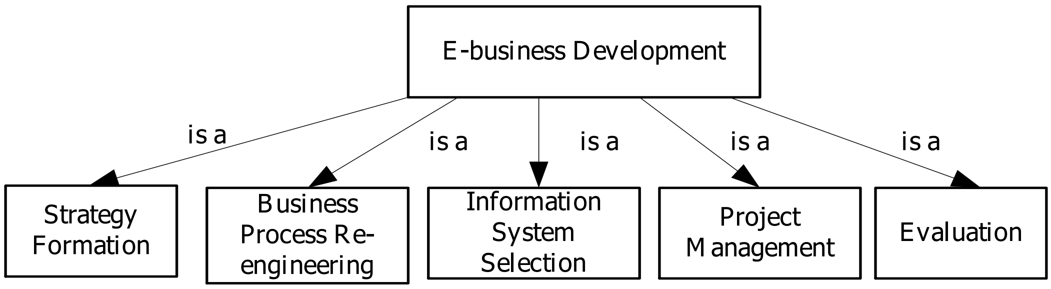

Fig. 4 E-business Development Ontology Taxonomy

However, in a complicated case, a problem may contain several sub-problems and the context of the sub-problems may vary. In other words, different techniques, information, and reference cases may be required in the sub-problems. Thus, another ontological representation of a case is required to represent the new case. Eventually these ontological representations will be clustered to represent the whole case (see fig. 5).

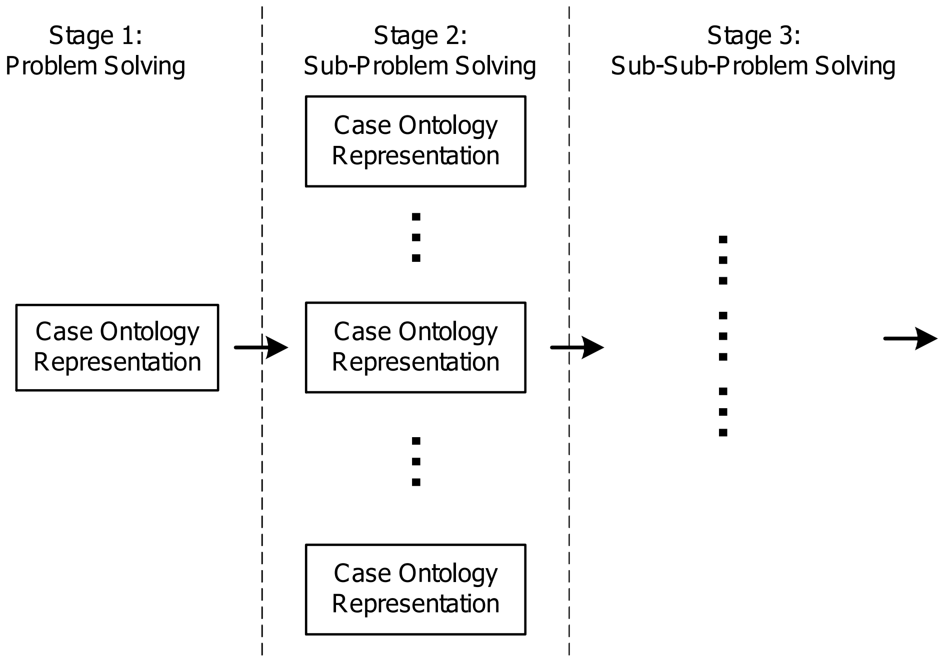

Fig. 5 Different Stages of Problem Solving

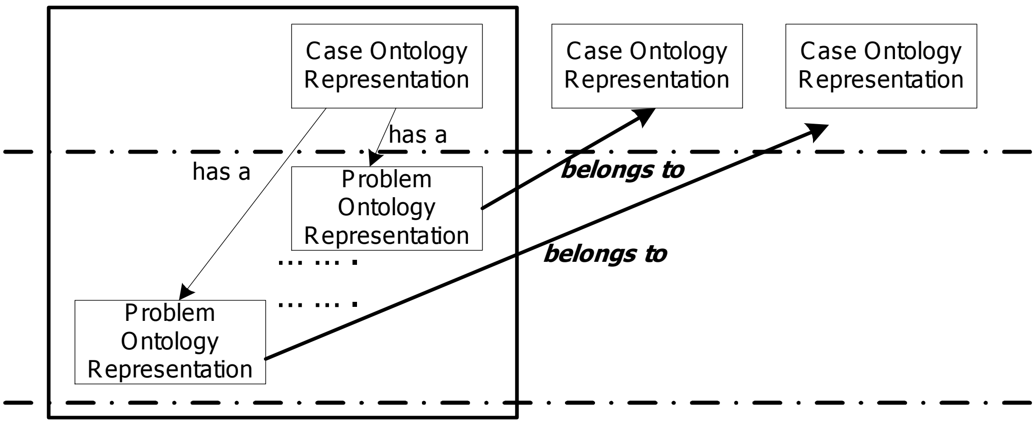

Fig. 6 Clustered Knowledge Map to Represent the Whole Case

To represent a case, we define seven classes that are called “Case”, “Problem”, “Context”, “Information”, “Technique”, “Event” and “Result” (see fig. 6). “Problem” is subclass of “Case”. The object properties of the “Case” class are “hasProblem”, “hasContext”, “hasInformation”, “usedTechnique”, “hasEvent”, “hasResult” and “referredCase”. These represent the semantic meaning of the case.

Case Retrieval using Ontology and Similarity Measurement

To retrieve the similar problem spaces from the historical cases, similarity measurement is commonly used in case retrieval (Montani et al., 2006; Passone et al., 2006; Cheng, 2005; Moczulski et al., 2004; Pal et al., 2004; Finnie et al., 2002; Nomoto et al., 2002; Liao et al., 1998). As mentioned before, synonym problems cause the mismatch of similar cases. Thus, a novel approach of ontology-based similarity measurement on problem space (see fig. 7) is proposed in this paper to overcome the synonym problem of case retrieval.

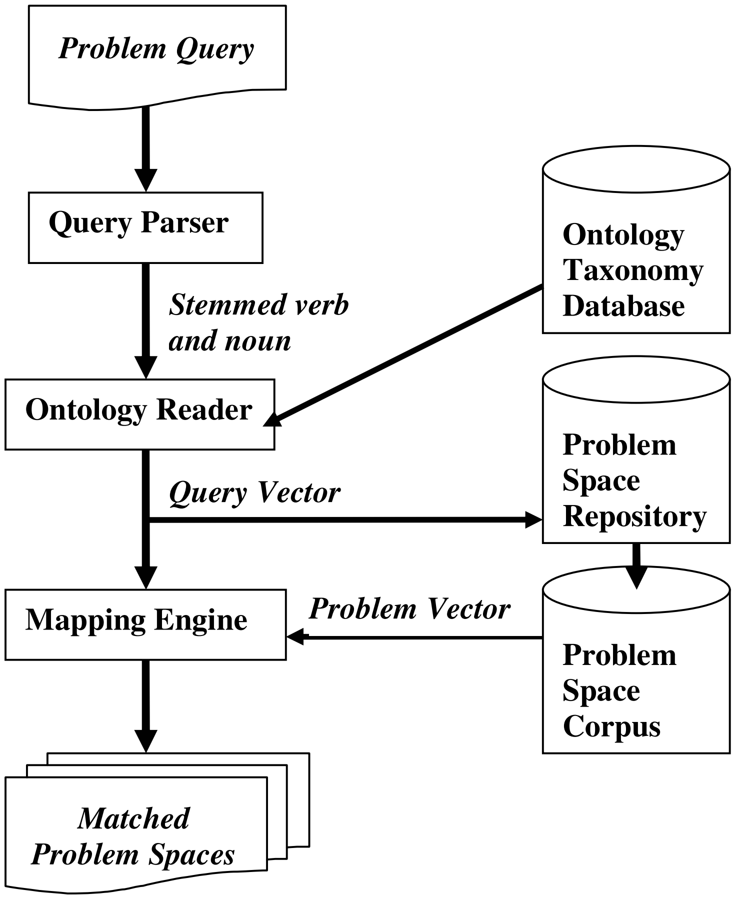

Fig. 7 An overview of the mechanism of Ontology-based Similarity Measurement

When a new case comes, problem solver can break down the new problem into sub-problems and these sub-problem query strings (e.g. How to do business process reengineering?) will be passed to the query parser. The query parser will then stem the query strings of the new sub-problems into verb and noun (i.e. do, business process reengineering) for matching the similar historical problem objects. Due to the synonym problem, the stemmed query terms may not match with the sub-problems of the similar cases. Thus, the noun term (i.e. "business process reengineering) will be passed to e-business ontology reader (see example in fig. 8) to extract all its child terms from the ontology tree to construct query vector $P_q(\vec{V}, \vec{N})$ for similarity measurement. In which, $\vec{V}$ is the verb vector (i.e. do, has, redesign, reengineer, design, evaluate,….. in this example), $\vec{N}$ is the noun vector (i.e. business process reengineering, business process, logistic management, production management, IT infrastructure, information system, ….. in this example).

…

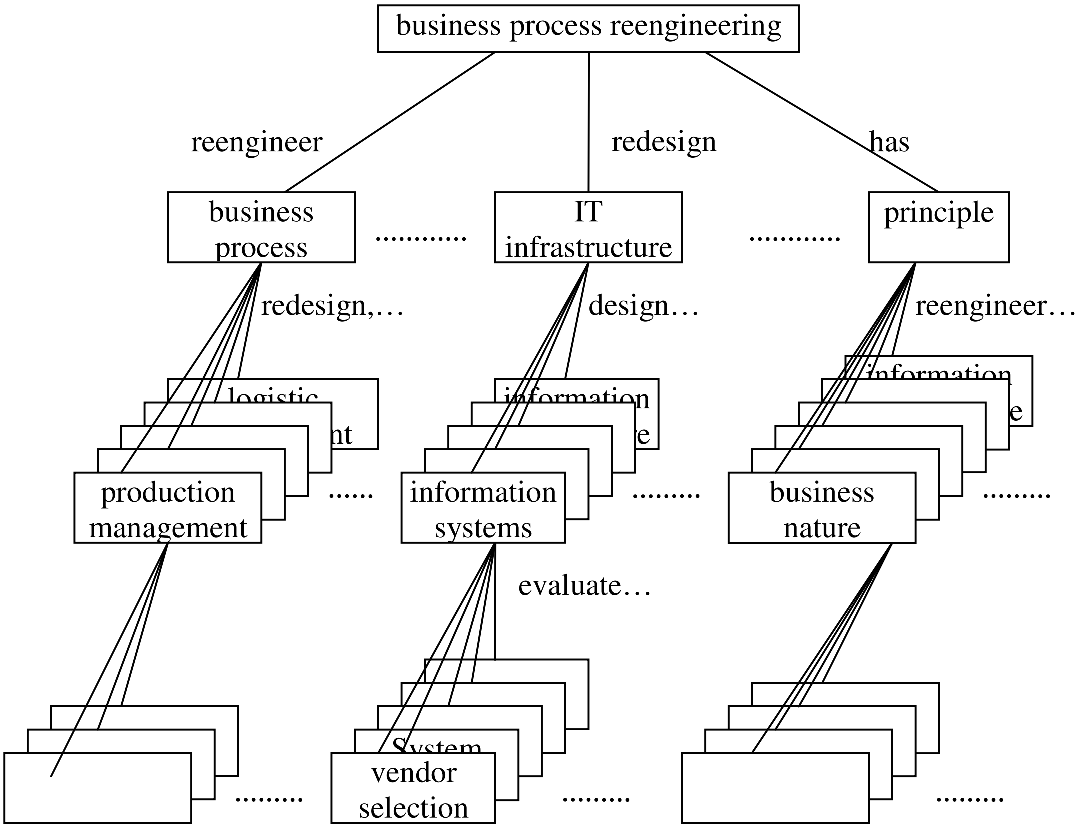

Fig. 8 Example of the sub-branch of "business process re-engineering" in the e-business ontology tree

The problem spaces with any term(s) of vector $\vec{V}, \vec{N}$ were collected as the problem space corpus and problem space *i* is represent by $P_i(\vec{V}, \vec{N})$. To measure the degree of similarity of the problem space in the corpus and the problem query, P_q, similarity coefficient (Grossman et al., 2004) is used because the problem query vector and problem space vector are similar in length. The calculation of the similarity coefficient is described as below.

The weighting factor, w_{ij}, for a term t_j in a problem space P_i is defined as $tf_{ij} \times ipf_j$, where

tf_{ij} is the number of occurrences of the term t_j in problem space P_i.

$ipf_j = \log \frac{p}{pf_j}$ is the inverse problem space frequency, where p is the total number of problem spaces and pf_j is the number of problem spaces that contain the term t_j.

The similarity coefficient (Grossman et al., 2004) between a query vector P_q and a problem space P_i is defined by the dot product of two vectors. For a collection of problem space with t distinct collection-wide terms, the similarity coefficient measurement is

$SC(P_q, P_i) = (\sum_{j=1}^{t} w_{qj} \times w_{ij})$, where t is the number of distinct terms in problem space collection (i.e. $\vec{V} + \vec{N}$).

If the problem space with higher similarity coefficient value, it implies that it has higher degree of similarity to the problem query.

System Evaluation

In the experiment, “How to do business process reengineering?" was selected as the target problem space. There were 50 problem spaces that contain the terms of $P_i(\vec{V}, \vec{N})$ was archived from our supply chain management case repository and was used as the problem space corpus for this experiment. From these 50 problem spaces, 25 problem spaces were asking how to do business process reengineering. We measured the performance of the case retriever by using recall that is the fraction of the relevant problem spaces which has been retrieved; and precision, that is the fraction of the retrieved problem spaces which is relevant. The recall and precision of case retriever R on problem space set P were defined as follows.

$$recall(R, p, P) = \frac{|X \cap Y|}{|X|}$$

$$precision(R, p, P) = \frac{|X \cap Y|}{|Y|}$$

where P is a set of problem spaces and p is a target problem space

X is the subset of P that is similar to p, and Y is the subset of P that is determined similar to p by our case retriever R.

The threshold of similarity coefficient ranged from 0.1 to 1.5 was used to determine its precise and recall (See fig. 9). From the experiment, the optimized threshold value is 0.09.

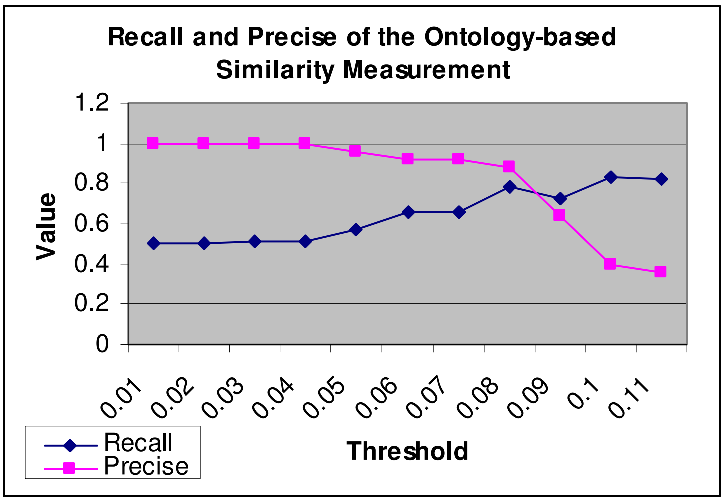

Fig. 9 Recall and Precise of the Ontology-based Similarity Measurement

Without using the ontology approach, the similar case was retrieved using the term "re-engineering" and "business process" only. The recall and precise were found to be 0.12 (i.e. 3 out of 25) and 0.6 (i.e. 3 out of 5) respectively. Thus, the recall and precision of the ontology-based similarity measurement performed better than that of the traditional similarity measurement.

Summary

In summary, the advantage of using ontology-based similarity measurement can overcome the synonym problem of similarity measurement. Since all the cases' problem solving steps have been decomposed and stored into the problem spaces. After the query parser stemmed the problem query, the ontology reader finds out all the related concept terms from the ontology taxonomy database for similarity measurement. The retrieved problem objects can be clustered, learnt and analyzed by problem solver to find a solution for the new problem.

Acknowledgement

This research was funded by UGC departmental research fund in the year of 2006/2007.

References

[1] Avramenko, Y., & Kraslawski A., (2006). Similarity concept for case-based design in process engineering. *Computers and Chemical Engineering, 30(3)*, 548-557

[2] Bergmann, R., Kolodner, J. & Plaza, E., (2005). Representation in case-based reasoning. *Knowledge Engineering Review, 20(3)*, 209-213.

[3] Bo, S.; Li, D.X.; Xuemin, P. & Li, H., (2003). Scenario-based knowledge representation in case-based reasoning systems. *Expert Systems, 20(2)*, 92 - 99.

[4] Cheng, Y., Melhem, H.G., (2005). Monitoring bridge health using fuzzy case-based reasoning. *Advanced Engineering Informatics, 19(4)*, 299-315.

[5] Dutta, S., Wierenga, B. & Dalebout A., (1997). Case-based reasoning systems: from automation to decision-aiding and stimulation. *IEEE Transactions on Knowledge and Data Engineering, 9 (6)*, 911 – 922.

[6] Finnie, G. & Sun, Z., (2002). Similarity and metrics in case-based reasoning. *International Journal of Intelligent Systems, 17(3)*, 273-287.

[7] Fong, J. & Huang, S.M., (1999). Architecture of a Universal Database: A Frame Model approach. *International Journal of Cooperative Information Systems, 8(1)*, 47-82.

[8] Fukumoto, T., Kishi, Y., (2007). Proposal of thinking process model based on putting a question to oneself for problem solving by skilled engineers. *Journal of Nuclear Science and Technology, 44(7)*, 997-1006.

[9] Grossman, D.A., & Frieder, O., (2004). *Information Retrieval: Algorithms and Heuristics*, third edition, Springer, 11-18.

[10] Han, S.G.; Lee, S.G., Jo, G.S., (2005). Case-based tutoring systems for procedural problem solving on the www. *Expert Systems with Applications, 29(3)*, 573-582.

[11] Liao, T.W., Zhang, Z.M., Mount, C.R., (1998). Similarity measures for retrieval in case-based reasoning systems. *Applied Artificial Intelligence, 12(4)*, 267-288.

[12] Meacham, J.A., & Emont, N.C., (1989). The interpersonal basis of everyday problem solving. In J.D. Sinnott (Ed.), *Everyday problem solving: Theory and applications*, 7-23.

[13] Montani, S., Portinale, L., (2006). Accounting for the temporal dimension in case-based retrieval: A framework for medical application. *Computational Intelligence, 22(3-4)*, 208-223.

[14] Moczulski, W., Szulim, R., (2004). On case-based control of dynamic industrial processes with the use of fuzzy representation. *Engineering Applications of Artificial Intelligence, 17(4)*, 371-381.

[15] Noh, J.B., Lee, K.C.; Kim, J.K.; Lee, J.K., Kim, S.H., (2000). Case-based reasoning approach to cognitive map-driven tacit knowledge management. *Expert Systems with Applications, 19(4)*, 249-259.

[16] Nomoto, K., Kise, W., & Kosuge, Y., (2002). Definition of a similarity measure between cases based on auto/cross-fuzzy thesauri. *Systems and Computers in Japan, 33(9)*, 99-108.

[17] Pal, S.K., & Mitra, P., (2004). Case generation using rough sets with fuzzy representation. *IEEE Transactions on Knowledge and Data Engineering, 16(3)*, 292-300.

[18] Passone, S., Chung, P.W.H., Nassehi, V., (2006). Incorporating domain-specific knowledge into a genetic algorithm to implement case-based reasoning adaptation. *Knowledge-Based Systems, 19(3)*, 192-201.

[19] Smith, M.U. (1991). A view from biology. In M.U. Smith (ed.), *Toward a unified theory of problem solving*. Hillsdale, NJ: Lawrence Erlbaum Associates.

[20] Sohn, M.H., Douglass, S.A., Chen, M.C., Anderson, J.R., (2005). Characteristics of fluent skills in a complex, dynamic problem-solving task. *Human Factors, 47(4)*, 742-752

[21] Wood, P.K., (1983). Inquiring systems and problem structures: Implications for cognitive development. *Human Development, 26*, 249-265.