Introducing Experiential Knowledge Platform: A Smart Decision Supporter for Field Experts

Keejun Han, Eunkyung G. Lee, Hyunwoo Je, Mun Y. Yi
Dept. of Knowledge Service Engineering
KAIST
Daejeon, South Korea
{keejun.han, grace.lee, jhw4824, munyi}@kaist.ac.kr

***Abstract*—Experiential knowledge is knowledge obtained through reflection on experience. In case of experiential knowledge within a specialized domain, this knowledge is strengthened over time as a field expert accumulates more experience in the chosen field. However, it is unfortunate that the knowledge is often confined within each individual in implicit form and it is hardly well-managed by an organization. Although there are several systems designed to acquire and exploit experiential knowledge, escalating maintenance costs pose serious challenges to their adoption and continuous use. In this paper, we propose a new knowledge-based system that acquires experiential knowledge through natural interactions with domain experts and keeps it growing by adding specialization rules, thereby reducing maintenance costs substantially. We also present the overall flow of how the acquired knowledge is processed and applied to decision supporting process, particularly in diagnosing potential diseases from blood tests.**

***Keywords*—experiential knowledge; knowledge-based system; procedural knowledge; descriptive knowledge**

I. Introduction

Experiential knowledge is knowledge obtained via experience. In many industrial fields, experiential knowledge plays a crucial role in solving sophisticated problems and making important decisions. This knowledge is expanded and strengthened over time as a field professoional accumulates more experience in his or her chosen field. However, it is unfortunate that the experiential knowledge exists mostly in implicit form. As such, it is hardly well managed in an organization even though it is imperative that the knowledge is transferred from senior experts to the newbies in the field. It takes usually years to be a field expert with a high level of understanding and proficiency in a professional field. This problem inevitably results in a situation where a limited number of senior experts have to handle overwhelming amount of tasks.

Needs for replicating and digitalizing the experiential knowledge have fueled the development of various knowledge-based systems, which have been commonly implemented utilizing heuristic rules [3, 5, 6], case-based reasoning [7, 8] and machine learning [4, 9, 10]. A knowledge-based system stores expert knowledge internally in a digitalized format and applies it to practical problem-solving with a high level of reasoning. In Section II, each of these techniques is visited and investigated by analyzing their strengths and weaknesses. In spite of the potential usefulness of every technique, these techniques commonly demand an excessive amount of management costs because they are unable to replace knowledge engineers during the transfer process from implicit knowledge to explicit knowledge in a digitalized format. In other words, knowledge engineers still must be involved in the system maintenance phase. Thus, organizations often cast doubt on Return on Investment (RoI) in managing the conventional knowledge-based systems.

To overcome the aforementioned limitations of the current knowledge-based systems, we propose a novel knowledge acquisition and management technique that not only learns the experiential knowledge from the field experts while interacting with them in the course of their normal system use, but also validates newly obtained knowledge without the prolonged involvement of knowledge engineers after system implementation. In short, we aim at developing a self-growing knowledge-based system for medical experts who work in clinical pathology examination. This field can be characterized with the following two salient attributes, among others: First, this field strictly requires a high level of experience to handle sophisticated tasks, consistently forcing workers under intense workload. Second, senior experts who are capable of making reliable decisions consistently are limited in number, consequently demanding a hefty paycheck. Expert decisions on pathological cases are directly linked to patient lives, which means a great amount of stress and responsibility, as well. Our proposed system is designed to benefit these field experts by providing consistent and reliable suggestions based on heuristic rules, derived from expert decisions on similar or identical cases.

II. Related Work

In this section, we first briefly introduce commercial knowledge-based systems with decision supporting functions using experiential knowledge. We then move on to review some state-of-the-art methods used to implement knowledge-based systems and platforms.

A. Commercial Platforms

Several attempts have been made to build knowledge-based systems and platforms in various fields. A knowledge platform

ARISAM [1], developed by Samsung SDS, incorporated a reward system to encourage knowledge sharing among its users at Samsung. WiseStar [2], developed by Incheon International Airport, is a similar knowledge sharing system that rewards experts who willingly post their own working experience in the system in order to raise the overall quality of services at the airport. In addition to the rewarding system, it also considers their working activities as procedural knowledge and facilitates the management of the activities by providing procedure maps. Similarly, K-Windows and K-Pert [3], introduced by Mahindra Satyam, are search and blog-alike systems for knowledge management along with user-friendly interfaces. In consequence, having the user-friendly interface led experts to actively post and search the experiential knowledge that satisfies their various needs.

In recent years, more intelligent knowledge-based systems not only to manage the knowledge, but also to support experts decision making have been launched. In a medical domain, Symptomchecker [4], introduced by WebMD, asks patients about their symptoms, shows similar diagnosis cases, and allows them to search for further information. In the domain of business, Sparkling Logic [5] has been in active use to support business decision making through machine-learning based analytic methods. A knowledge-based system was also developed for agriculture. A system named Rice Doctor [6] predicts possible consequences about the soil and climates of the surrounding areas on the basis of the location of experts. With that information, experts, farmers in this case, can minimize possible threats and risks.

In sum, there have been multiple attempts to exploit experiential knowledge for improved productivity with gradual advancement in recent years. In fact, the number of patents about knowledge acquisition and management has been steadily increased each year. In 2010, the number of the registered patents was 517 and it went up to 826 in 2014 , implying that decision-supporting technologies utilizing experience knowledge is still under development.

B. Methods for Decision Supporting

To our best understanding, in order to build such knowledge-based systems that provide decision supporting, there have been mainly 3 methods: Heuristic Rule-based System (HRS), Case-based Reasoning (CBR), and Machine Learning-based System (MLS).

HRS makes a decision in accordance with the pre-defined rules specified by knowledge engineers in advance. Its implementation is straightforward and reliable as the rules are pre-generated and those rules are heavily refined by the experts. However, it is almost impossible to expand new rules without constant involvement of knowledge engineers, resulting in updating of the system hard and often delayed.

CBR has been one of the common methods to implement such systems for a long time. Given a new case, a system looks for similar cases by calculating similarities among existing cases and suggests solutions that are derived from the most similar cases. Despite the established reliability of this method, however, it also holds some limitations: a lack of elaboration on calculating similarity values and no possible solutions for distinct cases.

Lastly, MLS has received an increasing amount of attention recently with the rise of Deep Learning. By letting machines learn rules from the training data, it performs decision making with much less involvement of knowledge engineers. Compared with the HRS and CBR, it can more effectively discover new rules by analyzing new cases after the initial rule set is generated, suggesting that MLS is mostly superior in maintaining and keeping the system updated than the other two methods.

However, MLS is not yet considered as the best choice for implementing knowledge-based systems. As new rules are automatically generated without proper validation from the experts, the rules are error-prone. Even if functions for validation can be provided, it is almost impossible for the experts to find out reasons of incorrect rules from inferencing as the systems are too complicated for domain experts to understand, indicating that knowledge engineers should still be needed for system management and maintenance even with the employment of MLS. Another prevailing disadvantage of MLS is so-called overfitting, which means that generated rules are too sensitive for a training dataset, consequently producing a non-versatility problem when the system is applied in another domain.

To overcome the limitations of the existing methods for the knowledge-based systems, we adopt a more practical method, called Ripple Down Rules (RDR), which conducts inferences based on initial rules and revises the rules by having the field experts review them directly without the involvement of knowledge engineers. For revising, they are not required to understand the background of the inferencing engine but they are still able to spot and correct wrong rules by just indicating which cases are the most similar or why the current rules are not applicable via controlled language interface (CLI). Underneath the whole rule addition and revision procedure is the representation of rules in RDF, which naturally accommodates rule expansions and rule updates by adding new rules to the existing rule set by treating them as specialization cases. In fact, RDR treats all new rules added as specialization cases and it naturally supports rule addition and revision by adding a specialization branch to the relevant path rather than rewrite the related rules. Unlike MLS, RDR uses human-validated rules as inputs. Thus, the rules are much more reliable. Furthermore, compared with HRS, costs for rule addition and validation are much lower because knowledge engineers do not need to be involved during the process to rule revision. Due to the limited length of this paper, detailed explanations about RDR are not given in this paper. However, an interested reader is referred to [12] for more technical details.

Table 1 shows a brief summary of the commercial knowledge-based systems for each method aforementioned. In Table 1, Y

[1]The figure was obtained from Korean Online Patent Search System, called KIPRIS (www.kipris.or.kr).

Table I. A summary of commercial knowledge-based systems. For each column, NKE indicates No Need for Knowledge Engineer, RU indicates Rule Update, RV indicates Rule Validation, and A/I indicates whether the service is currently Active or Inactive.

Name	Type	NKE	RU	RV	A/I
Rice Doctor[6]	HRS	N	N	N	A
CaDet [7]	HRS	N	N	N	I
Symtomchecker [4]	HRS	N	N	N	A
Athena [8]	CBR	Y	N	N	I
LPA Toolkit [9]	CBR	Y	N	N	I
Azure ML [10]	MLS	Y	Y	N	A
Knowledge Studio [11]	MLS	Y	Y	N	A
Sparking Logic [5]	MLS	Y	Y	N	A
Pacific Knowledge [12]	RDR	Y	Y	Y	A

indicates yes or positive for the corresponding column, otherwise, we assign N. We address that Y indicates that the service is more practical knowledge-based systems. Furthermore, A/I stands for active or inactive that indicates whether the service is currently in operation or not. If the service is not in operation though all of the systems were launched after 2010, it could indicate that the utility of the underlying methods is not so promising from a system maintenance perspective. Note that the bottommost service is developed based on RDR and it seems the most stable knowledge-based system for decision supporting as it has the longest history of active use among those in operation.

III. Proposed System

In this section, we describe the proposed decision making system, particularly designed for field experts who diagnose potential diseases, with given measurements from blood tests. Thus, we developed the proposed system to help field experts make better decisions (diagnoses in this case).

The proposed system makes use of two types of experiential knowledge: descriptive knowledge [12] and procedural knowledge [13]. Descriptive knowledge is the background knowledge used for decision-making, commonly expressed in declarative sentences or indicative propositions [12]. Meanwhile, procedural knowledge is the knowledge used to solve problems in a specific domain, usually expressed as implicit procedures to complete given tasks [13]. To the best of our knowledge, the proposed system is the first attempt to utilize the two types of knowledge simultaneously in a knowledge-based system. Although the majority of conventional knowledge-based systems use only a single type of knowledge for inferencing, it should be noted that the two types of knowledge are complementary. In the field of clinical pathology, for instance, doctors often diagnose using procedural knowledge given that most cases are repetitive. However, for some cases which are turned out to be new (e.g., newly emerging influenza or virus), the procedural knowledge is not enough to make exact diagnoses. In this case, the descriptive knowledge is also required to understand the relatively uncommon attributes of new influenza as well as the attributes of existing similar influenza.

Table 2 shows several cases that are partial results of clinical pathology examination for better understanding of the scenario. For an anonymized patient, we assume that his or her results for each test are given; then the system returns automatically generated comments by conducting inferences based upon the rules represented in RDR. It is noted that these rules are primitive, implying that rules should be confirmed by the doctor. Below we first explain how the rules are updated by utilizing the relevant procedural knowledge. We then describe a situation when the doctor wonders some test results, then the system automatically returns a list of descriptive knowledge that is related to the current cases.

Table II. An example of input case. Patient ID indicates anonymized patent ID. Code indicates test ID. Name indicates blood test name. L/H indicates if the value is Low or High.

Patient ID	Code	Name	Value	L/H
20150418-56422	00018	T. Bill	3.3	H
20150418-56422	00019	D. Bill	3.3	H
20150418-56422	00021	AST	286	H
20150418-56422	00022	ALT	839	H
20150418-56422	00023	r-GTP	532	H
20150418-56422	00025	ALP	210	H
20150418-56422	00029	LDH	311	H
Comments				
Exceptionally high for AST, ALT, r-GTP, ASP, and LDH with a high level of Bilirubin, possible disorder of biliary.				

A. Rules Strengthened by Using Procedural Knowledge

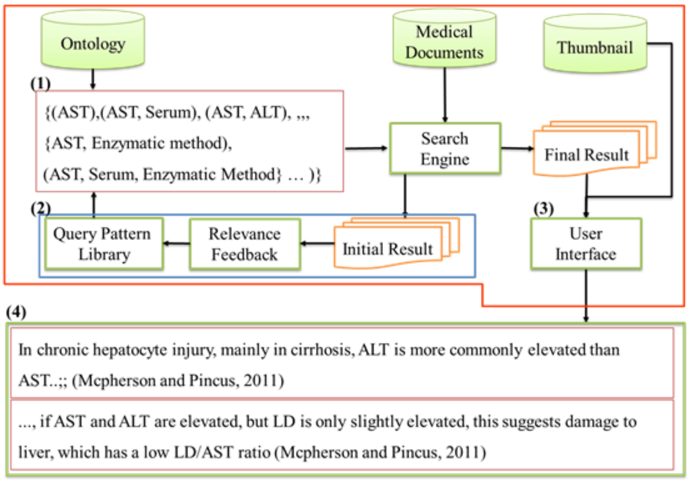
Let us assume that a doctor is not satisfied with comments returned by the system. He wants to address that test results also imply possible disorder of liver as well as biliary. Via event interactions, the system catches a specific part of the current rules to be strengthened and returns a set of rules that corresponds to the section automatically.

Table 3 shows our abstract visualization of a set of rules that contains the section where the doctor makes modifications. Note that the doctor can make changes on any part of conditions and consequences in the rules. If a condition is not satisfied within a single step, it can be shown as nested-if in the condition section. Meanwhile, a complete sentence is usually used for the consequence section. In this case, the doctor decides to make a change in the consequence section. His actual change is given in bold in Table 3. It should be noted that fundamental theories about this rule strengthening are originated from RDR [12]. However, the procedural knowledge alone is not sufficient to verify whether the change is correct or not. This vulnerability is relieved by exploiting descriptive knowledge, as described below.

Table III. Display of a set of rules that an end-user makes a change

Condition Section
[1] IF (GTP = hc) THEN [1.1] IF (AST = hc) & (ALT = hc) THEN [1.1.1] IF (LDH = hc) & (ALP = nc) THEN [1.1.1.1] IF (T.Bill = hc) & (D.Bill = hc) THEN
Consequence Section
Exceptionally high for AST, ALT, r-GTP, ASP, and LDH with a high level of Bilirubin, possible disorder of biliary (and liver).

Figure 1. A overall flow of returning relevant descriptive knowledge



B. Rule Validation by Using Desriptive Knowledge

Based upon the changes, the system searches all indexed documents from the Internet and reachable handbooks with a set of queries that indicate those tests related with the changes made. In this scenario, AST, ALT, r-GTP, ASP, and LDH should be re-visited altogether due to the fact that the consequence caused by combination of the tests was changed. The search system is simply implemented using Lucene for now, however, it will also cover Language Modeling [14] for better search performance by using Indri [15] in the future.

To formulate a query set, we mainly use two sources for now: Ontology and Relevance Feedback [16]. The ontology we created for this system covers terms and relations about all existing blood tests included in clinical pathology examinations. It also covers some medical terms and their relations from existing successful medical ontologies such as UMLS [17] and KOSTOM [18]. Other than the ontology, more terms retrieved for each test can be obtained from the set of documents through Relevance Feedback by extracting potentially meaningful terms from the documents that are regarded as relevant by the experts.

Due to the limited length of this paper, fundamental theories of determining relevant documents and extracting terms from the documents are referred in [14, 15, 16]. Instead, we only focus on describing the overall flow of how the relevant descriptive knowledge is returned, shown in Figure 1 as follows: (1) Formulating various combinations of queries for AST and ALT tests, (2) Extracting more terms for the tests through relevance feedback, (3) Returning final retrieved descriptive knowledge with corresponding images to the doctor for decision confirmation, (4) Retrieving descriptive knowledge about AST and ALT tests. The doctor can confirm whether his/her changes are correct because the retrieved knowledge supports a possible relation between the test and liver disorder, discovering ‘cirrhosis’ and ‘liver’ in the sentences.

IV. Concluding Remarks

In this paper, we have reviewed several existing knowledge-based systems using experiential knowledge. We also reviewed strengths and weaknesses of main methods used in each of the system. Unlike these extant knowledge-based systems, our proposed system maintains and expands experiential knowledge with field experts involved, but without the constant involvement of knowledge engineers. The proposed system is built upon the two types of knowledge: descriptive knowledge and procedural knowledge. In this paper, we have showed the overall flow of how each knowledge is used to help field experts make decisions, especially in a medical diagnosis domain.

Although there is yet much room for improvement, we expect that the proposed system will not only lessen the workload of field experts but also significantly decrease decision errors and inconsistencies through the knowledge validation process that involves field experts. Furthermore, given that our system is free from the overfitting problems, the system can be more readily adaptable to actual use.

Acknowledgment

This work was supported by the Industrial Strategic Technology Development Program, 10052955, Experiential Knowledge Platform Development Research for the Acquisition and Utilization of Field Expert Knowledge funded by the Ministry of Trade, Industry & Energy (MI, Korea).

References

[1] S. Jang, J. Choi, K. Hong, J. Jung, Knowledge Activity Processes and Knowledge Management System: An Empirical Examination of the Relationship Between Behavioral Features of Knowledge Management System, Management of Engineering and Technology, 2001, vol. 1, pp. 72-73.
[2] http://www.cyberdigm.co.kr/product_intro/applied_case/success_story/SuccessStory(incheon-KMS).pdf
[3] http://www.techmahindra.com/media/press_releases/Satyam-Launches-Virtual-Learning-World.aspx
[4] http://symptoms.webmd.com/#introView
[5] http://www2.sparklinglogic.com/
[6] http://www.knowledgebank.irri.org/decision-tools/rice-doctor
[7] P. A. Rodgers, A. P. Huxor, N. H. M. Caldwell, Design Support Using Distributed Web-based AI Tools, Research in Engineering Design, 1999, 11(1), pp. 31-44.
[8] M. Fahim, M. Idris, R. Ali, C. Nugent, B. Kang, E. Huh, S. Lee, ATHENA: A Personalized Platform to Promote an Active Lifestyle and Wellbeing Based on Physical, Mental and Social Health Primitives, Sensors (Basel), 2015, 14(5), pp. 9313-9329.
[9] http://www.lpa.co.uk/cbr.htm
[10] https://azure.microsoft.com/en-us/services/machine-learning/
[11] http://www.ibm.com/smarterplanet/us/en/ibmwatson/knowledge-studio.html
[12] B Kang, P Compton, P Preston, Multiple classification ripple down rules: evaluation and possibilities, The 9th Knowledge Acquisition for Knowledge Based Systems Workshop, 1995.
[13] R. Johnson, Bethany, and M. Schneider. Developing conceptual and procedural knowledge of mathematics. Oxford handbook of numerical cognition, 2014.
[14] Goodman, D. Noah, and S. Andreas. Knowledge and implicature: Modeling language understanding as social cognition. Topics in cognitive science 5.1, 2013, pp. 173-184.
[15] http://www.lemurproject.org/indri/
[16] S. Yu., D. Cai, J. Wen, W. Ma, Improving pseudo-relevance feedback in web information retrieval using web page segmentation, Proceedings of the 12th international conference on World Wide Web, 2003, pp. 11-18
[17] https://www.nlm.nih.gov/research/umls/
[18] http://www.namok.or.kr/