Available online at www.sciencedirect.com

ScienceDirect

Journal of Biomedical Informatics 41 (2008) 318–326

Journal of Biomedical Informatics

www.elsevier.com/locate/yjbin

A collaborative medical case authoring environment based on the UMLS

Siriwan Suebnukarn [a,*], Peter Haddawy [b], Phattanapon Rhienmora [b]

[a] *Thammasat University, Graduate School of Dentistry, Pathumtani 12121, Thailand*
[b] *Asian Institute of Technology, School of Engineering and Technology, Pathumtani 12120, Thailand*

Received 6 May 2007
Available online 2 September 2007

Abstract

In this paper, we present a novel collaborative authoring tool that was designed to allow medical teachers to formalize and visualize their knowledge for medical intelligent tutoring systems. Our goal is to increase the efficiency and effectiveness in creating the domain model representing the problem solution—often referred to as the bottleneck in developing intelligent tutoring systems. We incorporate the Unified Medical Language System (UMLS) knowledge base to assist the authors in creating the problem solution collaboratively via a videoconferencing platform. The system consists of a shared workspace gathering information visualization and tools necessary for collaborative problem-solving tasks. We found that the authoring tool can be used to effectively elicit the knowledge structure of the domain model. This was achieved in hours compared to months for the conventional paper-based approach.

Keywords: Medical education; Collaborative authoring; Unified Medical Language System; Intelligent tutoring systems; Domain model

1. Introduction

There has been increasing interest in intelligent medical tutoring systems that utilize a wide range of Artificial Intelligence techniques to model the domain knowledge that specifies what to teach, and pedagogic strategies that specify how to teach. This explicit knowledge encoding makes it possible to make inferences about a student's mastery of topics or tasks in order to dynamically adapt the content or style of instruction [1]. The Collaborative Medical Tutor (COMET) provides an environment that emulates that of human tutored medical problem-based learning (PBL) sessions while at the same time permitting the students to participate collaboratively from disparate locations [2,3]. The system combines concepts from computer-supported collaborative learning with those from Intelligent Tutoring System (ITS). COMET contains four primary components similar to any typical ITS: domain clinical reasoning model (or domain model), student clinical reasoning model (or student model), pedagogic module, and student multimodal interface.

In PBL group discussion the students evaluate the patient problem presented to them exactly as they would a real patient, attempting to determine the possible underlying anatomical, physiological, or biochemical dysfunctions and to enumerate all possible causal paths (hypotheses and their causal links) that would explain the progression of the patient's problems. Generating appropriate tutorial actions requires a model of the students' understanding of the problem domain and of the problem solution. However, as in human tutored PBL sessions, COMET must provide an unrestricted interaction style that gives students the freedom to solve the patient case without having to explain the rationale behind their actions. This complicates the modeling task because the system must infer the student's level of understanding from a limited number of observations. To deal with the resulting uncertainty, we selected Bayesian networks as our modeling technique. For each scenario (patient problem) taken from the PBL curriculum, the domain model is contained

[*] Corresponding author. Fax: +66 2 986 9205.
E-mail address: ssiriwan@tu.ac.th (S. Suebnukarn).

1532-0464/$ - see front matter © 2007 Elsevier Inc. All rights reserved.
doi:10.1016/j.jbi.2007.08.007

in the part of the structure of the network that represents the hypotheses (possible solutions) and the cause–effect relations among them as well as how the hypotheses are derived. The student model is contained in the part of the network that represents how the hypotheses are derived and in the network's probabilities.

The system implementation is modular and the tutoring algorithms are generic so that adding a new scenario or case requires only adding the appropriate model representing how to solve a particular case (domain model). Creating the domain model is not a trivial task and requires significant expert knowledge and time. The domain model for each case required about one person month to build for a 3-h PBL tutorial session. The aim of this study was to develop an authoring tool for medical PBL tutor to assist the creation of new cases.

Researchers have been investigating ITS authoring tools almost since the beginning of ITS research in order to allow non-programmers to formalize and visualize their knowledge. Several authoring tools have been built for authoring the domain model, e.g., IDE [4], Eon [5]. They were all designed to work in a single-user mode. However, developing a complicated case particularly in medical PBL usually requires experts from different areas such as anatomist, physiologist, pathologist, and specialist in the field. The necessity to collaborate to produce high-quality PBL cases lead us to the requirement of collaboration support in the authoring tool, which is the first contribution of our work.

A second issue is to support use of standardized medical terminology. In medical problem solving, there are often many ways of expressing the same idea using synonymous phrases, like "Pneumonia", "Lung inflammation", or "Pulmonary inflammation". Consequently, free text is not suitable to represent medical problem solutions in the domain model where student hypotheses are matched against. Terminological standardization helps to solve this problem. In this way, students have freedom to use the synonyms representing the same standard terminology and they will be recognized as correct if they appear in the domain model. Furthermore, using standard terminology permits ready linking to rich sources of medical knowledge. Our approach is to integrate the Unified Medical Language System (UMLS) [6] directly into the authoring environment in such a way that case authoring becomes a process of browsing through UMLS and selecting the relevant medical concepts. This ensures standardization of terminology and linkage of the problem solution to a rich source of medical knowledge, which can be used by the tutoring module. An additional benefit is that because authoring becomes a process of browsing and selecting, the authors may be reminded of hypotheses that they might otherwise neglect to include.

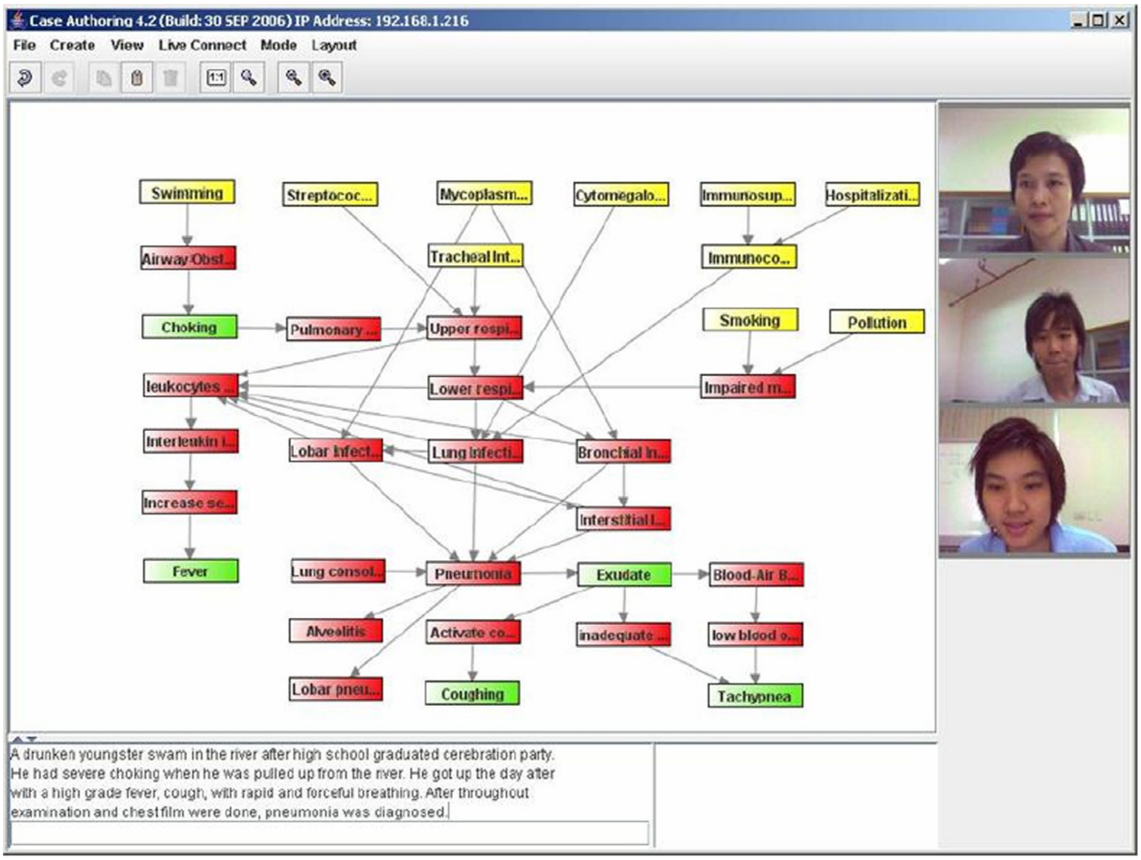
The UMLS was designed by the National Library of Medicine to integrate many authoritative biomedical source terminologies into a unified knowledge representation. The UMLS knowledge base includes: (i) a Metathesaurus of terms and concepts from many different biomedical vocabularies and classifications. Each concept in the Metathesaurus is assigned to one or more semantic types; (ii) a Semantic network of relationships among the semantic categories to which concepts of Metathesaurus are assigned; and (iii) a Specialist lexicon and programs for generating the lexical variants of biomedical terms. There have been several attempts to reuse the UMLS knowledge base in a variety of contexts, e.g., in patient care [7], clinical radiology [8]. The use of UMLS to obtain relevant domain knowledge and remind oneself of useful relations between concepts that might otherwise be overlooked by a free-text search engine or a human being has been addressed, e.g., in a medical search engine for all [9] and HealthCyberMap's tool for building an RDF metadata base [10]. We are the first to use it in this way in an ITS domain model authoring tool and the first to combine it with a collaborative tool that includes text chatting and video conferencing.

2. The COMET authoring tool description

The overall goals of the COMET authoring tool are to: (i) decrease the effort (time, cost, other resources) for creating the domain models; (ii) decrease the skill threshold for building intelligent tutors by allowing non-programmers to take part in the design process; (iii) help the authors articulate or organize their domain knowledge; and (iv) support good design principles. Consequently, our interface incorporates UMLS for knowledge management and knowledge reuse, as well as the graph visualization for knowledge visualization with the tools essential for collaborative work via the videoconference.

2.1. Knowledge management

As in many other ITSs, COMET predefines some of the main components including the tutoring module, student model and student interface, and requires from the author to construct only the domain model for each scenario. Essentially, authors who are medical PBL experts, are asked to create the problem solution. In this process, medical problem-solving consists in the real-time construction of a problem model in the form of a network comprising hypothesis nodes characterized by their ontological levels—enabling conditions, faults, and consequences—that the authors can enumerate further from a scenario. The semantic relations are cause–effect relationships among hypotheses. Every hypothesis node has a unique apply node as one of its parents. The apply node represents the application of a medical concept to a goal in order to derive the hypothesis. During this process, the problem is progressively transformed into a solution model by assigning particular goals and medical concepts to each hypothesis. The resulting network which is a shared mental model of the authors becomes the structure of the Bayesian network (BN) domain model. The details of the BN domain model are explained in [11].



A drunken youngster swam in the river after high school graduated cerebration party. He had severe choking when he was pulled up from the river. He got up the day after with a high grade fever, cough, with rapid and forceful breathing. After throughout examination and chest film were done, pneumonia was diagnosed.

Fig. 1. User interface of the COMET authoring tool.

The COMET authoring tool provides online communication among authors from different locations via text chat and video conferencing. Connection to medical terminology in the Unified Medical Language System (UMLS) is included so that the effort required to bind medical concepts to nodes in the model is reduced.

Fig. 1 illustrates the main interface of the tool. The drawing pane (Fig. 1, upper left pane) dominates most of the interface and allows interaction and visualization of the created model. Basic graphical tools are provided in the toolbar, e.g., copy, paste, undo, redo, and zoom (Fig. 1, menu bar). The experts with the *write* privilege can edit the model simultaneously while those with *read* privilege can only see the changes in real-time and discuss with others via the chat pane.

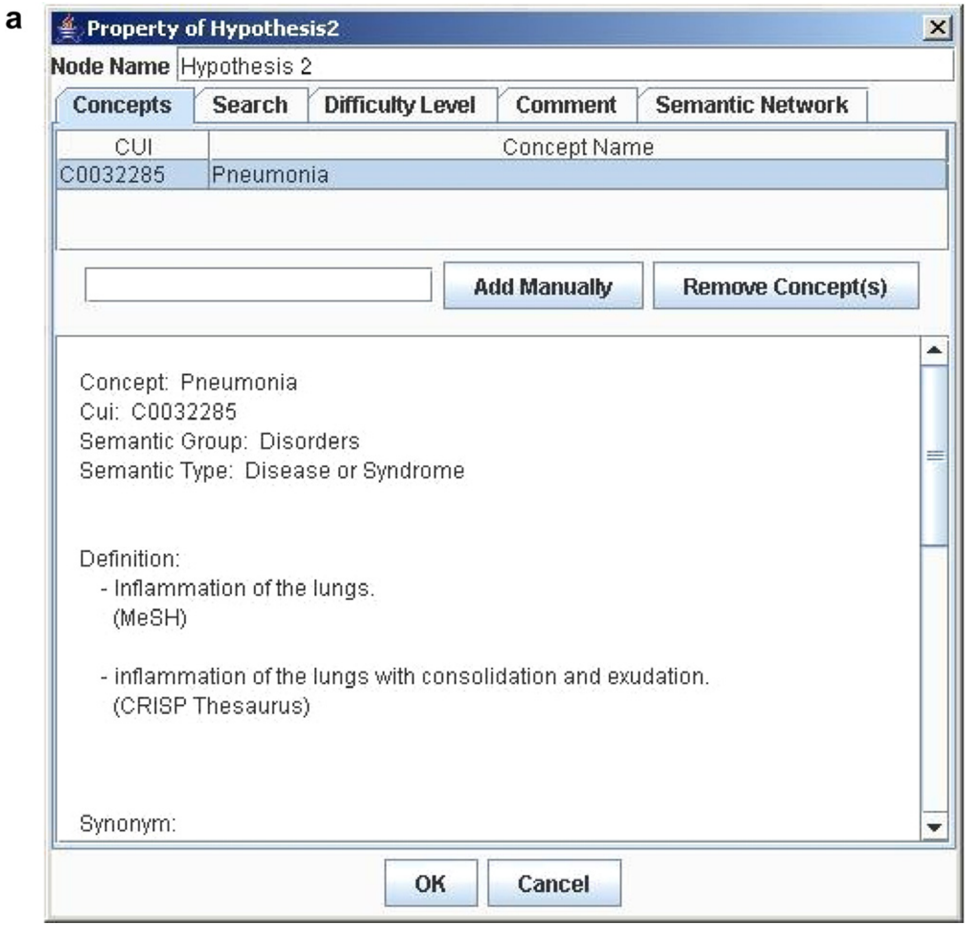
A hypothesis node can be created from the context menu and a link can be drawn to represent the cause–effect relationship between two nodes. Colors are used to indicate different kinds of nodes. Goal, concept, and apply nodes are shown in gold, pink, and blue[1], respectively. Hypothesis nodes have three sub-categories, enabling condition, fault, and consequence. They are in red, yellow, and green, respectively, as shown in Fig. 1 (drawing pane). Double clicking each node brings up its property window as shown in Fig. 2a. The *Search* tab allows searching for a medical concept in UMLS. Returned concept results including their explanation can be selected and added to a node name. Note that the input for searching can be a phrase or a sentence containing many medical terms, e.g., lung inflammation (Fig. 2b). The Metamap Transfer (MMTx) program provided by the UMLS is called to convert the input string to a list of medical concepts to be searched (lung, pneumonia, and entire lung).

Below the drawing pane is the chat pane (Fig. 1, lower left pane). The last pane located on the right is the video conference interface. This allows experts with web camera and microphone to communicate via voice and video in addition to text. The video and audio conference is implemented over the Java Media Framework (JMF) using the unicast Real-Time Protocol (RTP). Changes made to model data are sent to all authors synchronously using a TCP/IP socket. Text data in the chat pane is also transmitted this way.

2.2. Knowledge reuse

There have been several attempts to reuse the UMLS knowledge base in a variety of contexts, e.g., patient care and clinical laboratory [7], building a terminological

[1] For interpretation of the references to color in the text, the reader is referred to the web version of this article.



Node Name: Hypothesis 2

CUI	Concept Name
C0032285	Pneumonia

Concept: Pneumonia
Cui: C0032285
Semantic Group: Disorders
Semantic Type: Disease or Syndrome

Definition:
- Inflammation of the lungs. (MeSH)
- inflammation of the lungs with consolidation and exudation. (CRISP Thesaurus)

Synonym:

Fig. 2. (a) The property window of a hypothesis node showing the UMLS concept name (pneumonia) including its semantic group, semantic type, and synonym. (b) The search tab of the property window showing the input medical term (lung inflammation) and the search results (lung, pneumonia, and entire lung).

knowledge base [12], and identifying concepts in medical school curriculum [13]. The COMET authoring tool is built to take advantage of the reusability of concept relations in the UMLS Semantic networks. We have experimented with the UMLS knowledge base for providing classes, concepts, terms and interconcept links for creating the medical PBL domain model. Browsing through the UMLS Semantic networks helps remind the author of the potential relationships between concepts of two particular semantic types which can be added to the domain model. Fig. 3 shows part of a pneumonia model built using the UMLS Semantic network of pneumonia concept, its parent and child.

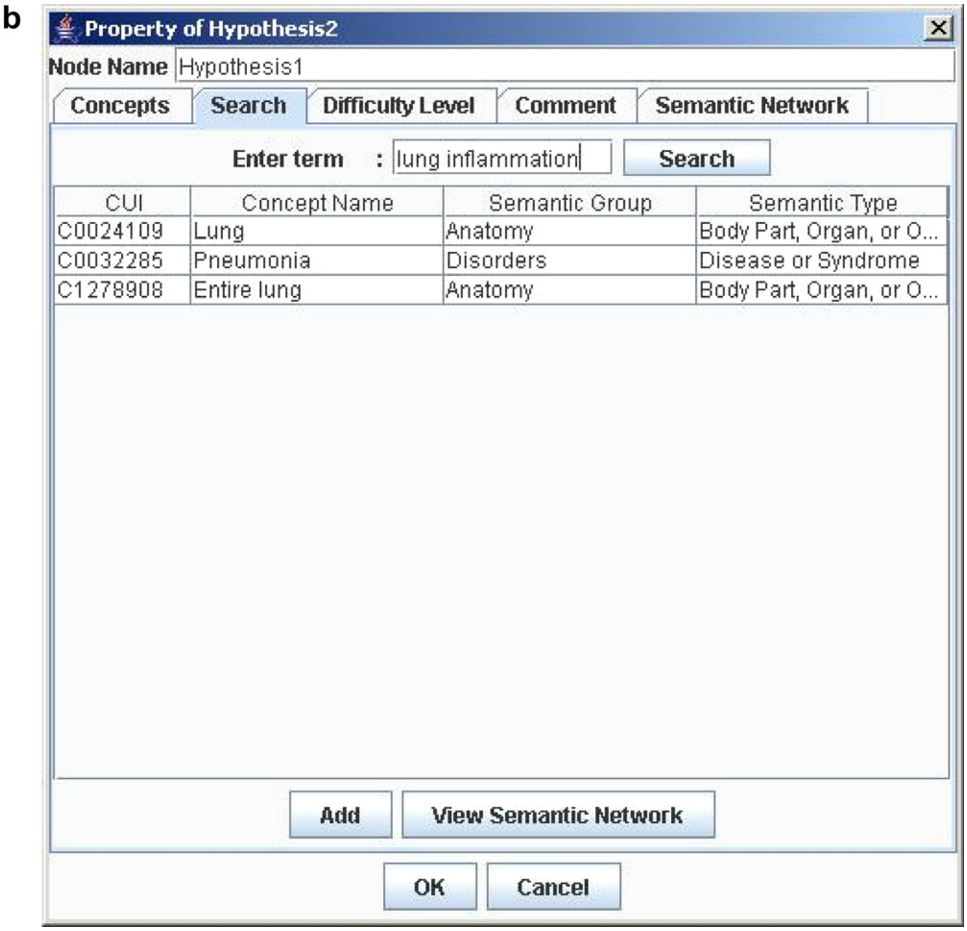
2.3. Knowledge visualization

One effective way to help authors comprehend large amounts of complexly interconnected knowledge is with visualization tools. The system provides key features such as copying, pasting, zooming, undo/redo, event-handling, and drag-and-drop support. These functions are implemented using JGraph an open source Java graph visualization library [14]. JGraph is compliant to Java Swing[2] API. Therefore, it is compatible with Swing features such as serialization[3], and data-transfer.

The medical case for our application has its specific layout. The goal and concept nodes, which are parents of apply nodes, are placed on the left most. The apply nodes, which can have multiple parents, are placed next to the right. The hypothesis nodes are placed on the right most and in hierarchical layout to illustrate the causal relationship among them. The authors can draw a case easily by following this layout; however, the authoring tool also provides the automatic layout feature using JGraph to help creating a better visualization of the layout such as moving nodes to the straight alignment and minimizing the crossing of edges.

[2] Swing is a graphical user interface (GUI) toolkit for Java. It is one part of the Java Foundation Classes. Swing includes GUI widgets such as text boxes, buttons, split-panes, and tables.

[3] Serialization is the storing of an object’s current state on any permanent storage media for later reuse.



Node Name: Hypothesis1

Enter term : lung inflammation

CUI	Concept Name	Semantic Group	Semantic Type
C0024109	Lung	Anatomy	Body Part, Organ, or O...
C0032285	Pneumonia	Disorders	Disease or Syndrome
C1278908	Entire lung	Anatomy	Body Part, Organ, or O...

Fig. 2 (*continued*)

3. Evaluation

3.1. Effectiveness of the authoring tool

In order to evaluate the effectiveness of the authoring tool, we measured the quality of the created problem solution. The problem solution is in the form of a network consisting of nodes (hypotheses) linked together by semantic relations. We compared the network built using the COMET authoring tool, with the network of a benchmark solution for the same domain. The benchmark solution was built from scratch using a paper-based problem solving approach.

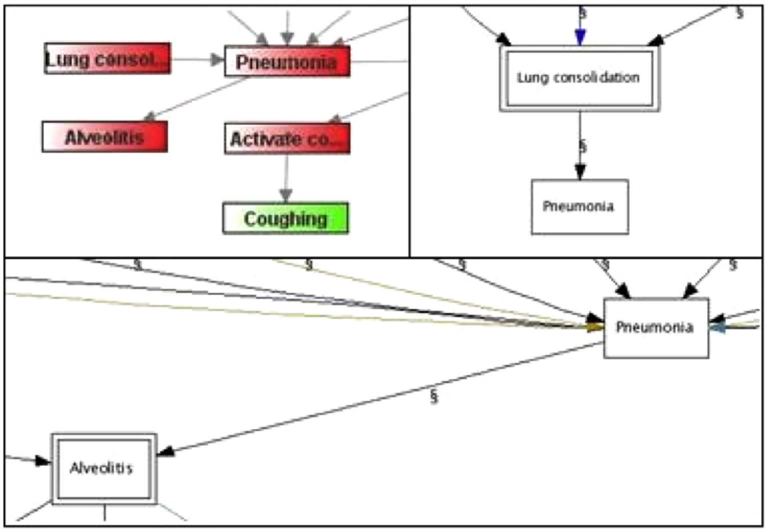


Fig. 3. Part of pneumonia model (upper left) built using the UMLS Semantic network of pneumonia concept and its parent (Lung consolidation, upper right) and child (Alveolitis, lower).

3.1.1. Methods

Six medical PBL tutors participated in the study: 2 physiology majors, 2 anatomy majors, 2 pathology majors. Each tutor had at least five years experience in conducting the PBL course at Thammasat university medical school. The participants were divided into two groups with the same proportion of expertise. Each group was asked to create a problem solution for three different scenarios Pneumonia, Heart attack, and Diabetes—which provide a reasonable diversity of the case. The first group used the COMET authoring tool via the school LAN to elicit the problem solution on given domains collaboratively, and the resulting semantic networks of the problem solutions were compared with the benchmark group. The benchmark group created the problem solution collaboratively using the conventional paper-based problem solving approach. Medical Textbooks, journals, and online resources were used in both groups.

All valid medical hypotheses and the links created were counted. The validity of hypotheses and links were verified by an expert in the area of Pneumonia, Heart attack, and Diabetes. In order to represent a global property of the networks that could account for the interdependence of the elements in the representation, we used a numeric function called Integration [15]. An integration mark (I) is defined by taking the ratio of the number of links (L) within the network to the number of nodes (N). We can generalize that formula for the whole representation: $I = L/N$

3.1.2. Results

Fig. 4 shows resulting networks for Pneumonia scenario created by benchmark group and the COMET authoring tool. The numbers of hypotheses, links and value of integration of all scenarios are presented in Table 1. With the support from the COMET authoring tool, there was a greater number of nodes and number of links, and the value of integration demonstrated by the authoring tool group was greater in all scenarios. The results show that from the same scenario, the authors who work with the COMET authoring tool built solution networks that were both richer and more integrated.

Although there was no gold standard in this evaluation, we found high degree of overlap of the resulting networks between the benchmark and the COMET authoring tool group. The following is the explanation of some differences. Despite the fact that the results reflect the effectiveness of the authoring tool, we are aware that we did not provide an analysis that clearly addresses the effectiveness of each tool feature in this study. Consequently, other explanations of our data might exist. For example, in the Pneumonia case, some of the pneumonia node's parents and children; e.g., Lung consolidation, Alveolitis and Lobular pneumonia; which did not appear in the benchmark group are UMLS concepts (Fig. 4, box B and D). This could be explained by the reminding that occurs when browsing through UMLS in the authoring process. Another feature of the tool that can affect the problem solution is the use of colors to represent hypothesis categories. This helps the users to quickly perceive and comprehend the evolving model, particularly in a multi-author environment. This feature could explain the fact that there are more ideas related to the enabling condition for the Upper respiratory tract infection in the COMET authoring tool group (Fig. 4, box A and C).

3.2. Usability of the authoring tool

The COMET authoring tool has been tested and improved to fulfill the authors' requirements during the system design and implementation. Initial training in the use of the tool requires between 1 and 2 h. Authors took between 6 and 11 h (to author a 3 h problem analysis session on pneumonia, heart attack, and diabetes)—a ratio of around 3 h per hour of tutoring. According to the follow-up interview, no author has found creating the problem solution incompatible with his or her approach. For the most part authors wanted the functionality that COMET provides and they were able to use it to create problem solutions that reflect their own pedagogic preferences. The participants mentioned in the follow-up interview that the visual communication, allowed by the real-time video-conferencing modality, as well as synchronous data transmission increased their level of confidence in their discussions and facilitated problem-solving.

4. Discussion

Intelligent Tutoring Systems have been shown to be effective in a number of domains, but they remain hard to build. One way to make ITSs more widespread is to create authoring tools that speed up tutor development. Authoring tools have the potential to increase the efficiency of building ITSs through reuse of common elements. Realizing reuse would require a resource library structure, where authored topics, activities, strategies, interface components, and/or domain knowledge could be stored independently from a tutor, and loaded from this library into any tutor. REDEEM [16] is built to take advantage of courseware libraries. The content and interactive screens of a REDEEM ITS are not authored using REDEEM, but are authored using ToolBook, an off-the-shelf multimedia authoring tool. ToolBook authored content is exported to a library and from there it is imported by REDEEM. Some ITS authoring systems infer or create new knowledge or information from scratch, saving the author from having to derive, articulate, and enter this information. RIDES [17] uses example-based programming techniques to infer general procedures from specific examples given by the author. RIDES creates a device's operational procedure by recording the author's actions as he uses the device simulation to illustrate the procedure. The DIAG system [18] infers a large body of device fault diagnosis information from a relatively small number of qualitative failure symptoms entered by the author. In this paper we describe the

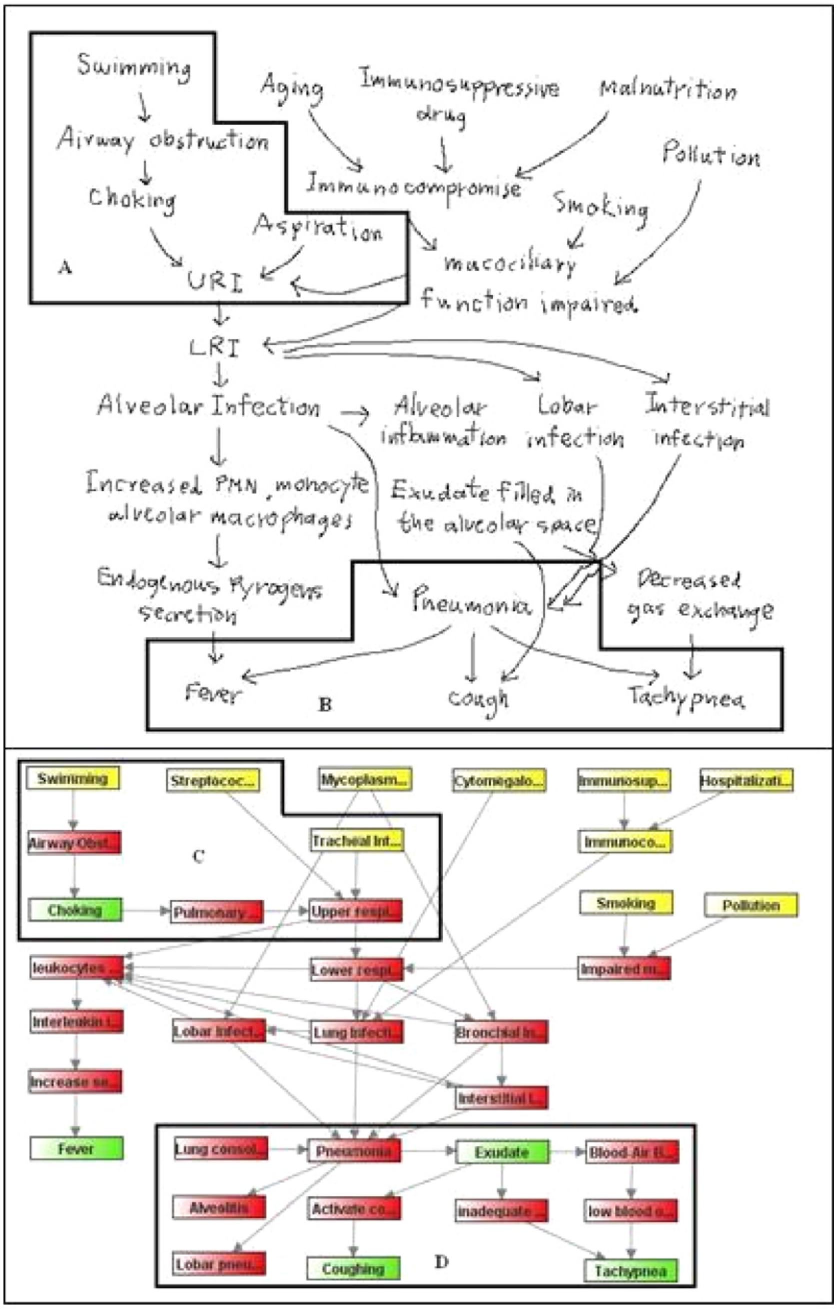


Fig. 4. Resulting networks of Pneumonia case created by benchmark group (upper) and the COMET authoring tool (lower).

Table 1
Numbers of hypotheses and links as well as value of integration, by cohorts and scenarios

Scenario	Cohort	Number of hypotheses	Number of links	Value of integration
Pneumonia	Benchmark	26	28	1.08
	COMET	35	43	1.23
Heart attack	Benchmark	22	23	1.05
	COMET	36	49	1.36
Diabetes	Benchmark	26	29	1.11
	COMET	34	49	1.44
All	Benchmark	74	80	1.08
	COMET	105	141	1.34

COMET authoring tool built to take advantage of the reusability of medical knowledge in the UMLS. All the terms in the domain model are linked to the UMLS Metathesaurus. All the possible relations among them are automatically determined using the UMLS Semantic network. According to our preliminary results, the UMLS provides a useful corpus of medical knowledge for designing a domain model for medical ITS.

The development of learning content is a collaborative process in which authors with different backgrounds, experiences, and points of view can take part. However, the collaborative development support in current learning contents authoring tools is scarce, since they usually provide a uni-personal vision of the creation process. But educational technologies in general are moving towards the use of second generation web-based communities or Web 2.0, which facilitate collaboration and sharing between users by the move to the internet as platform [19]. WEAR [20] is a web-based authoring tool for the construction of ITSs in Algebra-related domains, such as physics, economics, chemistry. To promote collaborative work among instructors, authors are offered the choice of seeing what other authors have done along two dimensions; the structure of a similar course and a list of problems constructed by other authors. In CARLOS (A Collaborative Authoring Tool for Reusable Learning Objects) [21], key factors of the collaborative creation process are taken into consideration, first negotiation and assessments of ideas between developers, who delegate those processes on their representing agents. Secondly, the results of those negotiations are automated included in the reusable learning objects corpus. CARLOS is able to trace of all changes and versions of the reusable learning objects during it creation process and those changes occur simultaneously on the same section of the same physical file. Whitehead and Goland [22] took a protocol-centric approach with a focus on interoperability to generate network effects and to add collaborative authoring capability to existing tools. The protocol is a set of extensions to HTTP which provide facilities for concurrency control, namespace operations, and property management. The protocol allows users to collaboratively author their content directly to an HTTP server, allowing the Web to be viewed not just as a read-only way to download information, but as a writeable, collaborative medium. In the COMET authoring tool, we integrate the most advanced technologies of computer-supported collaborative work and the information processing to provide a cooperative environment for authors to communicate for building the ITS medical domain model. The system can be operated by a number of authors in different locations to negotiate and solve the same patient case through the graph visualization tools and distributed environment. Although only a small number of subjects (six authors—three using COMET and three using pen and paper) were involved in this pilot evaluation of the tool, the study gave us encouraging results.

5. Conclusions and future work

We have exploited computer-supported collaborative work environments and reused UMLS in the development of the COMET authoring tool. This has allowed us to provide flexible mechanisms and interfaces to allow authors to collaboratively building the domain knowledge of the medical PBL cases. The evaluation showed that the authors who worked with the COMET authoring tool built solution networks that were both richer and more integrated than the network of a benchmark solution built from scratch using the conventional paper-based approach for the same domain. This was achieved in hours compared to months for the conventional paper-based approach. Although our primary goal was to build the authoring tool for an ITS for medical PBL, we believe that the tool will also be useful for case authoring in medical PBL in general. In our future work, we plan to develop reasoning algorithms from the UMLS Semantic networks in order to automatically convert them into the medical ITS domain models.

References

[1] Wenger E. Artificial intelligence and tutoring systems. Los Altos: Morgan Kaufman; 1987.

[2] Suebnukarn S, Haddawy P. A Bayesian approach to generating tutorial hints in a collaborative medical problem-based learning system. Artif Intell Med 2006;38:5–24.

[3] Suebnukarn S, Haddawy P. COMET: a collaborative intelligent tutoring system for medical problem-based learning. IEEE Intell Syst 2007;22:70–7.

[4] Russell D. IDE: the interpreter. In: Massey P, Mutter S, editors. Intelligent Tutoring Systems, Lessons Learned. Hillsdale: Lawrence Erlbaum; 1988. p. 323–49.

[5] Murray T. Authoring knowledge-based tutors: tools for content, instructional strategy, student model, and interface design. J Learn Sci 1988;7:5–64.

[6] National Library of Medicine. Available from: http://umlsks.nlm.nih.gov/kss.

[7] Cimino JJ. Use of the unified medical language system in patient care at the Columbia-Presbyterian Medical Center. Methods Inf Med 1995;34:158–64.

[8] Friedman C. The UMLS coverage of clinical radiology. In: Proceedings of the 17th symposium computer applications in medical care. New York: McGraw Hill; 1993, p. 307–16.

[9] Can AB, Baykal N. MedicoPort: a medical search engine for all. Comput Methods Programs Biomed 2007;86(1):73–86.

[10] Boulos MN, Roudsari AV, Carson ER. Towards a semantic medical Web: HealthCyberMap's tool for building an RDF metadata base of health information resources based on the Qualified Dublin Core Metadata Set. Med Sci Monit 2002;8(7):124–36.

[11] Suebnukarn S, Haddawy P. Modeling individual and collaborative problem-solving in medical problem-based learning. User Modeling User-Adapted Interact 2006;16:211–48.

[12] Schulz S, Hahn U. Medical knowledge reengineering—converting major portions of the UMLS into a terminological knowledge base. Int J Med Inform 2001;64:207–21.

[13] Denny JC, Smithers JD, Miller RA, Spickard A. "Understanding" medical school curriculum content using KnowledgeMap. J Am Med Inform Assoc 2003;10(4):351–62.

[14] JGraph—the java open source graph drawing component. Available from: http://www.jgraph.com/jgraph.html.

[15] Raufaste E, Eyrolle H, Marine C. Pertinence generation in radiological diagnosis: spreading activation and the nature of expertise. Cogn Sci 1998;22:517–46.

[16] Major N, Ainsworth S, Wood D. REDEEM: exploiting symbiosis between psychology and authoring environments. Int J Artif Intell Educ 1997;8:317–40.

[17] Munro A, Johnson MC, Pizzini QA, Surmon DS, Towne DM, Wogulis JL. Authoring simulation-centered tutors with RIDES. Int J Artif Intell Educ 1997;8:284–316.

[18] Towne DM. Approximate reasoning techniques for intelligent diagnostic instruction. Int J Artif Intell Educ 1997;8:262–83.

[19] Kamel Boulos MN, Wheeler S. The emerging Web 2.0 social software: an enabling suite of sociable technologies in health and health care education. Health Info Libr J 2007;24(1):2–23.

[20] Moundridou M, Virvou M. WEAR: a web-based authoring tool for building intelligent tutoring systems. In: Proceedings of the second Hellenic conference on artificial intelligence. Greece: Springer; 2002, p. 203–14.

[21] Padron CL, Dodero JM, Diaz P, Aedo I, Fernandez C. CARLOS: a collaborative authoring tool for reusable learning objects. In: Proceedings of the 14th international workshop on database and expert systems applications. Czech Republic: Springer; 2003, p. 269–73.

[22] Whitehead EJ, Goland YY. WebDAV a network protocol for remote collaborative authoring on the Web. In: Proceedings of the sixth European conference on computer supported cooperative work. Copenhagen: Kluwer; 1999, p. 12–6.