*Journal of ICT in Education (JICTIE)*
*ISSN 2289-7844 / Vol. 2 / 2015 / 40 – 54*

# Hybrid Similarity Measure in SmartLP: A Case Based System in Lesson Planning

**Aslina Saad[1], Paul W.H. Chung[2], Christian W. Dawson[2]**
[1]Faculty of Art, Computing & Creative Industry,
Universiti Pendidikan Sultan Idris
[2]Computing Science Department, Loughborough University
aslina@fskik.upsi.edu.my, {P.W.H.Chung, C.W.Dawson1}@lboro.ac.uk

**Abstract**

Planning for teaching imposes a significant burden on teachers as they need to prepare different lesson plans for different classes according to various constraints such as students' ability and previous knowledge. SmartLP, a case-based lesson planning system, has been implemented as a means of assisting teachers in constructing quality lesson plans more quickly and efficiently. SmartLP applied Case Based Reasoning (CBR) approach which consists of the following processes; retrieve, reuse, revise and retain within its cycle. CBR is the process of solving new problems based on the solutions of similar past problems. The first and most crucial step of solving new problem in CBR is the process of retrieval relevant cases from the case base. There are several factors that affect case retrieval such as similarity measure, indexing approach and all techniques within the approaches. Both computational (measures of similarity) and representational approaches were used for case retrieval in SmartLP which is done via lesson plan searching. There are five types of search in SmartLP; Advanced Search, Hierarchical Search, Search by Browsing, Boolean and Basic search. In Advanced Search, a hybrid approach; combination of distance based (computational) and representational was applied for case retrieval. In addition, each indexed element is assigned with certain number to indicate their relative importance as a terms weighting mechanism. Terms expansion within semantic approach was implemented in Hierarchical Search, based on lesson plan ontology. In Search by Browsing, the search terms are organised in general to specific manner which also derived from lesson plan ontology. Boolean search applied Boolean concept while Basic Search is a general search with exact search keywords. This paper presents the retrieval methods in SmartLP via five types of search.

**Keywords** Case retrieval, computational approach, representational approach, similarity measure, weighting, terms expansion, case search

## INTRODUCTION

SmartLP, a web base system was developed based on Case Based Reasoning (CBR) approach to assist teachers in constructing lesson plans that meets various constraints for different classes. CBR is an approach to problem solving that emphasizes the role of prior experience during future problem solving (i.e., new problems are solved by reusing and if necessary adapting the solutions to similar problems that were solved in the past). CBR has enjoyed considerable success in a wide variety of problem solving tasks and domains (Mantaras et al., 2005).

CBR is useful for a wide variety of problem solving tasks; diagnosis, creation and planning. Kolodner, a popular researcher and author in CBR has realized CBR potential in education even though not much research was done on CBR in education. Kolodner et al. (2006), reviewed work on the application of CBR to the design and construction of educational approaches and computer-based teaching systems. They concluded that although not much evaluation and assessment of case-based tools and pedagogical approaches has been done, what does exist shows positive indicators. Wang et al. (2007) report that while many successful CBR systems have been developed as knowledge repositories for preserving intellectual capital and for problem solving in business communities, both the concept and methodology of CBR are still novelties in education communities and developing and evaluating practical CBR educational applications is conspicuously scant.

Solving a problem by CBR involves obtaining a problem description, measuring the similarity of the current problem to previous problems stored in a database, retrieving one or more similar cases and reuses the solution of one of the retrieved cases, with or without modification to account for differences in problem descriptions. It is seen as time efficient to customise one's existing lesson plans rather than starting everything from scratch. The adaptation process of the previous solutions in CBR will fit the current problem's context which subsequently brings new solutions to the problem after being retrieved.

SmartLP enables teachers to retrieve previous lesson plans and customise them according to their constraints rather than start everything from scratch, as lesson plans should be tailored to accommodate differences according to the profiles of students and teachers as well as the facilities available. This paper presents approaches and techniques applied in SmartLP for case retrieval which are relevant lesson plans.

## CASE RETRIEVAL IN CBR SYSTEM

The implementation of CBR systems may vary but they include the following five steps; representation, retrieve, reuse, revise and retain in some form or other (Raman, 1995; Watson and Marir, 1994). In case retrieval, the closest-matching precedent is identified. This is the issue of retrieving one or more plans which solve problems similar to the current one. In CBR, the basic processes of solving a new problem or interpreting a new situation entail the retrieval of relevant cases from a memory of cases (case base) followed by the adaptation of the past solution.

Given the description of a new problem called the query case, the first, and arguably most crucial step, in a CBR system is to retrieve those cases from the case base that are most relevant to solving the query case. Spallazi (2001) remarks that one of main activities in case based planner is plan retrieval The key factors affecting the performance of the retrieval mechanism are representation, indexing and similarity metric of parts. A good representation, indexing and similarity metric will enable the system to retrieve the most similar case rapidly and correctly (Chang et al., 2000).

Azuaje et al. (2000) point out three fundamental approaches for the retrieval of relevant cases in CBR namely:

- computational approaches, (based upon measures of similarity)
- representational approaches, (based upon indexing structures)
- hybrid approaches, which combines computational and representational.

Matching and ranking is a procedure in case retrieval that selects which cases are appropriate among the cases in the case library. As the process of searching the library is done, the search process asks the matching function to compute for the degree of match among indexes. Based on the result of the matches, the search function collects a set of cases that partially match the new situation. The matching cases are then ranked to identify which best address the requirements of the new situation (Reyes and Sison, 2002).

According to Chang et al. (2000), proper query terms significantly affect the performance of document retrieval systems and can be improved by using query expansion techniques. They present a new method for query expansion based on user relevance feedback techniques for mining additional query terms. According to the user's relevance feedback, the proposed query expansion method calculates the degrees of importance of relevant terms of documents in the document database.

Guha et al. (2003) distinguish two major forms of search: Navigational and Research. In navigational search, the user is using the search engine as a navigation tool to navigate to a particular intended document. On the other hand, in research search, the user provides the search engine with a phrase which is intended to denote an object about which the user is trying to gather/research information.

Rather than using ranking algorithms such as Google's PageRank to predict relevancy, semantic search uses semantics or the science of meaning in language, to produce highly relevant search results. In most cases, the goal is to deliver the information queried by a user rather than have a user sort through a list of loosely related keyword results. Other authors primarily regard semantic search as a set of techniques for retrieving knowledge from richly structured data sources like ontology as found on the Semantic Web. There are so many techniques applied for case retrieval in various CBR applications. Techniques used in other retrieval systems might be useful in being considered for case retrieval in SmartLP.

## CASE RETRIEVAL IN SMARTLP

The case retrieval in SmartLP refers to search functions that acquire relevant lesson plans from the case base due to the constraints teachers have. SmartLP system provides five types of search; namely, Advanced Search, Hierarchical, Boolean, Basic Search and Browsing. Computational approach which is based upon measures of similarity was used in conjunction with representational approaches in different types of search. They are elaborated in the corresponding type of search.

Similarity definitions and similarity characterisation (weighting/ranking) were implemented in an Advanced Search. In Hierarchical search, the similarity is based on the curriculum and students' hierarchy of the term in their domain structure. Search by Browsing utilises the same approach as Hierarchical Search. Free keywords are allowed in Basic Search (full text) while the Boolean Search applies Boolean concept. Wild card queries and uppercase/lowercase flexibility are implemented in all types of search for flexible case retrieval.

The next subsections describe the details of each type of search together with the various approaches and techniques within the search. At the end of each subsection, a retrieval example is presented to illustrate the retrieval process. The explanation includes a retrieval algorithm and similarity measure ending with an example, which illustrates the function of each type of search.

### Computational and Representational Approach in Advanced Search

A hybrid approach, which combines computational (also known as distance-based approach) and representational approach, was used for case retrieval in the Advanced Search. The distance-based approach in this system applied a standard function-based measure for hierarchical and linear similarity while the index-based approach enforced weight adaptation for the indexed attributes which is discussed in this section.

According to Ashford and Willett (1988), best match searching implies the calculation of some quantitative measure of similarity between the query and each document in the file- the calculated similarity then forming the basis for the ranking. They emphasised that the most important component of a similarity measure is the term weighting scheme which is used to allocate numerical values to each of the index terms in a query or a document to demonstrate their relative importance. Therefore, query weighting is used in this system to give flexibility for users in determining the importance of each element of a lesson plan. Besides, they will get a better search result based on their constraints because the results are varies from one users to another based on the weight they assigned to each indexed element. Ranking, which gives significant value to the search result, was also implemented in the SmartLP system as those at the top are likely to have a strong degree of relevance to the query.

### *Terms Weighting*

Searched keywords may have different importance to different users. Therefore, in the Advanced Search, weights are assigned to each searched keyword to indicate their relative importance. It tells the system how much weight is to be assigned to each attribute as compared to the other attributes that make up the case. The weights are taken into account in calculating the similarity of the searched keywords in comparison to attributes in each case in the case base.

Users are allowed to rate each element which implies the importance of the searched keywords weighting in the range of 1 (least important) to 5 (most important). Alternatively, they can simply use the default values defined in the system. The default values were gathered in the knowledge acquisition process as teachers need to specify the importance of each element in lesson plans. This is essential for the similarity calculation between the problems (searched keywords) and cases in the case-base.

### *Similarity Measure*

In the Advanced Search, the similarity of two cases is calculated rather than calculating the difference by taking into account each elements index. Patterson (2003) stated that case indexing has been widely applied in CBR as a method of improving search efficiency to combat the effects of the utility problem. As only a selective portion of the case-base is made available during retrieval, search time is reduced and the efficiency of identifying a possible solution is increased dramatically.

A similarity calculation is applied in order to find the most similar cases to the given problem. A similarity value is in the range of 0 and 1, whereby 0 corresponds to totally dissimilar while 1 is a perfect match. For similarity values, some indexed attributes are based on hierarchical matching and some are linear matching. Learning areas, topics, year and learning outcomes are attributes that use hierarchical matching concepts while ability, knowledge, motivation, time period and number of students per class use linear matching concepts.

The hierarchical matching is based on hierarchical structure of curriculum domain presented in Figure 1. The detail of the curriculum hierarchy is explained in four levels. The first level is the subject, ICT, followed by the second level, learning area. There are six learning areas in ICT subject. This is then detailed in the third level; topic, where each learning area has several topics and each topic has one or more learning outcomes. Based on these learning outcomes, teachers should construct learning objectives based on constraints they have in hand; their students profile and facilities available. In this diagram only multimedia learning area is illustrated.

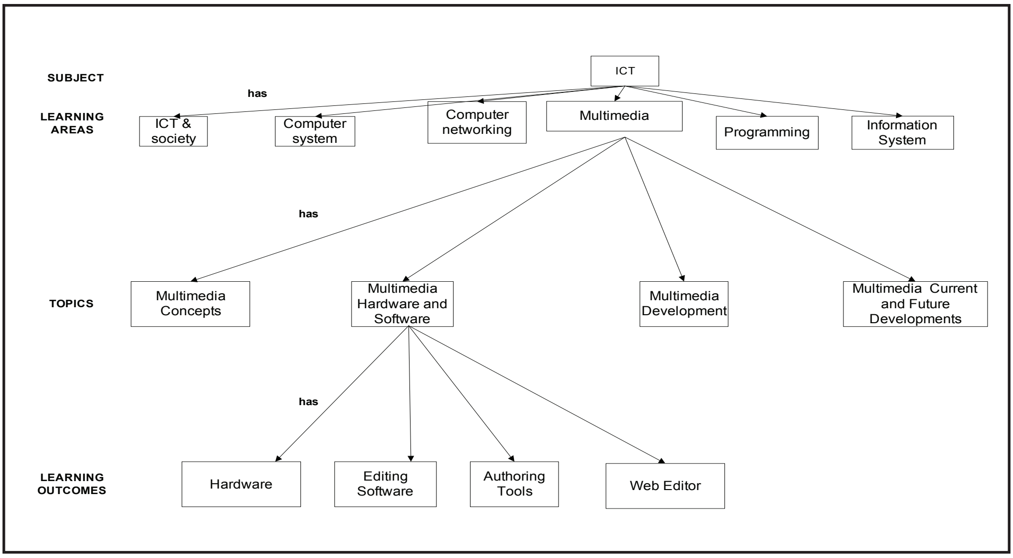


**Figure 1** Hierarchical Structure of ICT syllabus

The following equation 1 is used to calculate the similarity between problems searched by users and the cases in the database for hierarchical matching (Chung, 2007):

$$\text{Similarity (Problem Attribute}_{i,}\ \text{PA}_i\text{, Case Attribute}_i\ \text{CA}_i) =$$

$$1 - \frac{(\text{Distance PA}_i \text{ to common parent} + \text{Distance CA}_i \text{ to common parent})}{(\text{Distance PA}_i \text{ to root} + \text{Distance CA}_i \text{ to root})}$$

The example of the similarity calculation for topic in curriculum hierarchy is shown in the Equation 2:

$$\text{Similarity (Editing Software , Web Editor)} =$$

$$1 - \frac{(1 + 1)}{(3 + 3)}$$
$$= 1 - (0.33)$$
$$= 0.67$$

The hierarchical similarity based on the hierarchical structure of a curriculum syllabus produces the similarity values in Table 1 (topic) and Table 2 (learning area). As there are several levels in the curriculum and the search keywords are organised in a hierarchical menu, only attributes in the same level will be compared to the parents.

**Table 1** Similarity Value for Learning Area (Multimedia)

| Case -Query | Multimedia Concepts | Multimedia Hardware & Software | Multimedia Development | Multimedia Current and Future Developments |
|---|---|---|---|---|
| Multimedia Concepts | 1 | 0.5 | 0.5 | 0.5 |
| Multimedia Hardware & Software | 0.5 | 1 | 0.5 | 0.5 |
| Multimedia Development | 0.5 | 0.5 | 1 | 0.5 |
| Multimedia Current and Future Developments | 0.5 | 0.5 | 0.5 | 1 |

**Table 2** Similarity Value for Learning Outcome (Multimedia Hardware and Software)

| Case - Query | Web Editor | Editing Software | Authoring Tools | Hardware |
|---|---|---|---|---|
| Web Editor | 1 | 0.67 | 0.67 | 0.67 |
| Editing Software | 0.67 | 1 | 0.67 | 0.67 |
| Authoring Tools | 0.67 | 0.67 | 1 | 0.67 |
| Hardware | 0.67 | 0.67 | 0.67 | 1 |

In linear similarity, the distances of the path between the searched keywords and the related data in the database were assessed. The similarity for class time period (in minutes) is presented in Table 3.

**Table 3** Similarity Value for Time Period (In Minutes)

| Case - Query | 40 | 50 | 60 | 70 | 80 |
|---|---|---|---|---|---|
| 40 | 1 | 0.8 | 0.6 | 0.4 | 0.2 |
| 50 | 0.8 | 1 | 0.8 | 0.6 | 0.4 |
| 60 | 0.6 | 0.8 | 1 | 0.8 | 0.6 |
| 70 | 0.4 | 0.6 | 0.8 | 1 | 0.8 |
| 80 | 0.2 | 0.4 | 0.6 | 0.8 | 1 |

The above similarity is based on the distance that is illustrated in Figure 2. The time period for each teaching lesson ranges from 40 minutes to 80 minutes. The range of each 10 minutes interval is 0.2. When it goes further from the case value, the similarity decreases by 0.2, as each range represents 0.2. For example, if the search keyword is 40 and the cases in the case base contain the same value (40), then the similarity is 1, a perfect match. If the case is 50, then the similarity is 1-0.2, which yields 0.8.

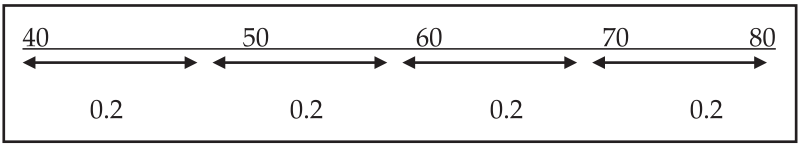


**Figure 2** Linear Similarity Value Based on Distance

These similarity values, which are in the range of 0 and 1, are kept in the database. All records in the case base were taken into account in calculating the similarity. The similarity of each element is then multiplied with the weight defined by users. If users do not specify any weight, the default value will be used. For weighted similarity, the calculation is shown in Equation 3.

$$\text{Weighted similarity} \quad \sum_{i=1}^{n} \frac{\text{Similarity}(\text{attribute}_i) * \text{weight}_i}{\sum_{i=1}^{n} \text{weight}}$$

The similarity between query Q1 and case *C*1 is defined as in Equation 4. This example is explained further in retrieval example.

$$S(Q;C1) = \frac{wA*S(A1;A2) + wB*S(B1;B2) + wC*S(C1;C2)}{wA + wB + wC}$$

*Where*

S= Similarity
*Q*1 = Query
*C*1 = Case
*w*A, *wB, wC* = weight of attribute A, B and C.
A1, B1, C1= attribute1, 2 and 3 from Q1 (query)
A2, B2, C2= attribute1, 2 and 3 from C1 (case)

## Retrieval Algorithm

The nearest neighbour, one of retrieval algorithm is used for Advanced Search with weights in comparing the attributes in the new case with each old case in SmartLP. According to Patterson et al. (2003), the nearest neighbor (NN) algorithm is a commonly used similarity metric in CBR. Its appeal includes its simplicity, its transparency, its robustness in the presence of noise and the fact that it does not require training.

In SmartLP, the similarity for all the indexed fields is calculated based on the hierarchical and linear similarity measure as explained before. In the end, all the similarities are summed to find the total similarity for all cases. The cases with the highest similarity are ranked on top. This is demonstrated in Figure 3.

1. Read entered keyword 1 – attribute A ,
2. Read entered weight of attribute A, WA
3. Read entered keyword 2 – attribute B ,
4. Read entered weight of attribute B, WB
5. Read entered keyword…n – attribute n ,
6. Read entered weight of attribute n, Wn
7. Search for attribute A, B, n in lesson table
8. Search for attribute A in similarity table A
9. Compare and read similarity value of attribute A in similarity table.
10. Search for attribute B in similarity table B
11. Compare and read similarity value of attribute B in similarity table.
12. Search for attribute n in similarity table n
13. Compare and read similarity value of attribute n in similarity table.
14. Calculate similarity:
    = multiply similarity value of each attributes and weight of that attribute
    = (SA * WA, SB * WB , Sn * Wn)
15. Total up the similarity = ∑ ( SA * WA, SB * WB , Sn * Wn)
16. Get the percentage = (total of the similarity/total weight)*100
17. Rank cases from table lesson.

**Figure 3** *Retrieval Algorithm for Weighted Search of SmartLP System* A Retrieval Example

In the following example for case retrieval, only two attributes are taken into account. In this system there are 10 attributes altogether and it depends on how many attributes were keyed-in by the user.

In Table 4, Case 1 has a 92.5% similarity to the query; compared to Case 2 that scores only 64.4%. Therefore, Case 1 is more similar to the query and will be displayed above Case 2 in the result list (ranked result). Whenever users decide to view a particular case, they will see the details of that lesson plan in a text-based format; a similar format to what they constructed manually.

**Table 4** Advanced Search Similarity Calculation

| | A1 | W1 | S1 | A2 | W2 | S2 | Total similarity/ total weight | Total similarity (%) |
|---|---|---|---|---|---|---|---|---|
| Query | 50 | 3 | | Input Devices | 5 | | | |
| *Case 1* | 40 | | 0.80 | Input Devices | | 1 | 7.4/8 | 92.5 |
| *Case 2* | 70 | | 0.60 | Output Devices | | 0.67 | 5.15/8 | 64.4 |

A1: Attribute 1 (time period)
W1: Weight of A1
S1 : Similarity of A1 to other attribute values (case 1 and case 2)
A2: Attribute 2 (objectives)
W2: Weight of A2
S2 : Similarity of A2 to other attributes values (case 1 and case 2)

The snapshot in Figure 4 shows the Advanced Search page. Instructions are presented clearly at the top of the page. The elements of a lesson plan are presented in a text-based format and are structured in a similar form to the paper-based format that they should have been familiar with. Default values of the elements' weight that show their importance are shown in the list box. If users would like to assign different values for the weights, they can select the values from the list box which holds numbers 1 to 5 whereby 5 indicates most important and 1 is least important.

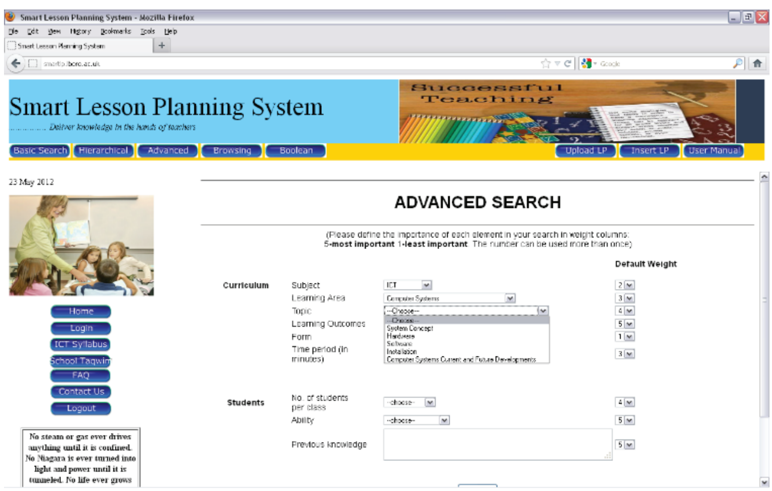


**Figure 4** *Weighting and Hierarchical Drop-Down Menu in Advanced Search*

The hierarchical drop-down menu is implemented on this page. Whenever users select a particular learning area, only topics related to the learning area will be listed. The same steps are applied to learning outcome. After selecting keywords and specifying the value of each element together with the weight, users will be presented with a list of lesson plans that are relevant to the query in descending order starting with the most relevant case to the least relevant. To view the detail, users should click the lessonID and the content of that lesson plan will be presented to the users.

## Semantic Approach in Hierarchical Search

Semantic approach is a mechanism of a query terms expansion technique that can improve performance of the system whereby users can specify their intent in more specific ways. Query terms expansion provides flexibility for users to choose related terms to the searched keywords, based on user relevance feedback techniques for mining additional query terms. Whenever users select any keywords, a list of related keywords will be suggested to them.

Hierarchical search in SmartLP system uses a semantics approach to produce highly relevant search results rather than using ranking algorithms as in Advanced Search to predict relevancy. In addition, it seeks to improve search accuracy by understanding searcher intent via the contextual meaning of terms as they appear in the searchable database within the system, to generate more relevant results.

The related terms are generated from richly structured data sources, the lesson plans ontology of curriculum and student domain as illustrated in Figure 5. This is based on a semantic relationship that has been transformed into hierarchical representation. Lesson plans taxonomy consists of four main nodes which are curriculum, students, facilities and content. Each node is then divided into detail nodes.

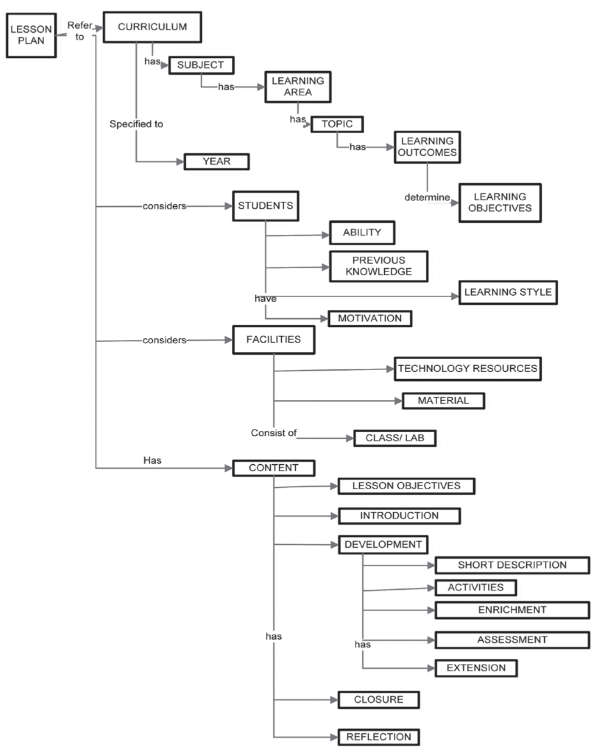


**Figure 5** Taxonomy of Lesson Plan Domain

In Hierarchical Search, when users click any of these terms, other related terms are presented to them in three levels; upper level (ancestors), same level (siblings) and lower level (children). They can freely choose those terms for further searching the lesson plans. It aims to support users in looking up keywords (searched attributes) that users do not understand. The algorithm for this hierarchical search is shown in Figure 6.

1. Read entered keyword 1 – attribute A ,
2. Search for attribute A in syllabus table
3. Search for attribute A in student table
4. Display results from related table
5. Particular terms selected by users
6. Show results of the terms in hierarchical structure

**Figure 6** *Retrieval Algorithm for Semantic Search*

### *A Retrieval Example*

Say users search for the keyword 'multimedia', all terms in the multimedia context will be displayed to users. Users are expected to select any of the given terms. For instance, if users select one of the displayed term-Multimedia Concept, they will be presented with other detailed terms in the hierarchical structure as in Figure 7.

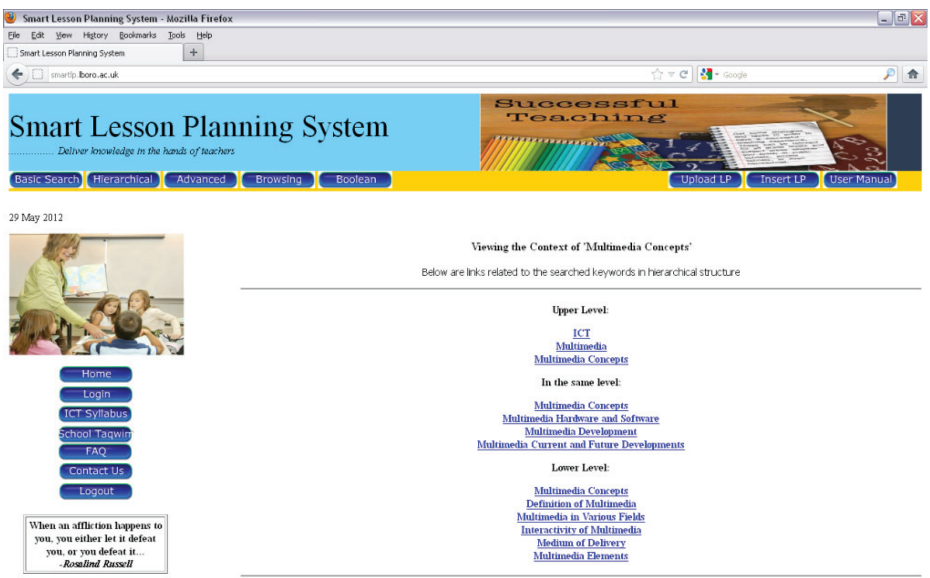


**Figure 7** Hierarchical Presentation of the Chosen Terms from the Previous Page.

*Journal of ICT in Education (JICTIE)*
*ISSN 2289-7844 / Vol. 2 / 2015 / 40 – 54*

## BOOLEAN SEARCH, BASIC SEARCH, BROWSING

Boolean searching allows users to narrow down their search by using special terms before the keywords. It is useful because it can help users make sure they do not get thousands of results when they search (BBCi, 2010). Bosswell (2010) explained that Boolean searches allow users to combine words and phrases using the words AND, OR, NOT and NEAR or use their math equivalents to limit, widen, or define search. Clapperton (2010) explains that with Boolean searching users use AND to make sure a keyword is included, AND NOT (ANDNOT, NOT) to make sure a keyword is not included and OR to give alternative keywords.

Boolean search allows users to combine words and phrases using the words AND, OR, NOT and NEAR to limit, widen, or define their search. In SmartLP, the terms 'with all of these words' represent AND, 'with at least one of these words' means OR while 'without this word' implies NOT. This is to help users understand the context of the Boolean usage in SmartLP. In basic search, lesson plans that contain the exact searched keywords will be displayed to users. The keywords are searched from all fields and tables in the case base. This search implements wild card queries for tolerant retrieval purposes.

In SmartLP, users who choose to search by browsing are presented with a choice of subject area by taking a broad subject area and drilling down through various subject headings and subheadings until the specific subject is reached. Furthermore, the terms are organised in a general to specific manner, and visualised by cascade menus. Therefore, users can expand and shrink the tree to find lesson plans with specific terms. Users can browse from two main areas which are 'Students' and 'Subjects' which derived from lesson plan ontology as shown in Figure 5 and the instances as illustrated in Figure 1. Figure 8 shows the browsing page of SmartLP System.

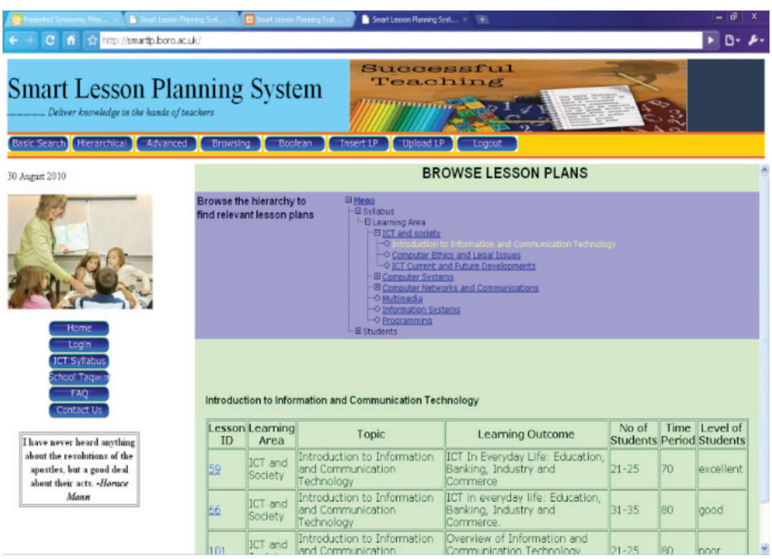


**Figure 7** Browsing Page of SmartLP

## CONCLUSION

The implementation of SmartLP emphasizes on case retrieval within the CBR cycle. Hybrid approach, which combines computational and representational technique, was used for case retrieval in the system. Hierarchical representation, based upon measures of similarity was used together with a computational approach, in terms of weighting. Query expansion and query weighting are used to give flexibility for users and to get a better search result. Similarity definitions and similarity characterisation (weighting/ranking) were implemented in an advanced search. In hierarchical search, the similarity is based on the curriculum and students' hierarchy of the term in their domain structure. Search by browsing utilises the same approach as hierarchical search. Free keywords are allowed in basic search (full text) while the Boolean Search applies Boolean concept. Representation of cases by constructing lesson plan ontology with the instances in hierarchical representation influenced case retrieval efficiency.

## REFERENCES

Allsopp, D. J., Harrison, A., & Sheppard, C. (2002). A database architecture for reusable commonkads agent specification components. *Knowledge-Based Systems*. 15, pp. 275-283.

Ashford, J. & Willett, P. (1989). *Text retrieval and document databases*. Bromley: Chartwell Bratt.

Azuaje, F. et al. (2000). Retrieval strategies for case-based reasoning: A categorised bibliography. *The Knowledge Engineering Review*. 15(4), pp. 371-379.

Boswell, W. (2010). Does boolean search mean? Retrieved December, 20 2010 from http://websearch.about.com/od/2/g/boolean.htm.

Chang, H.C. et al. (2000). Indexing and retrieval in machining process planning using case-based reasoning. *Artificial Intelligence in Engineering.* 14(1), pp. 1-13.

Clapperton, G. (2010). Boolean search. Ask Bruce. What are Boolean operators. Retrieved December 12, 2010 from http://www.bbc.co.uk/webwise/guides/boolean-operators.

Guha, R., McCool, R., & Miller, E. (2003). Semantic search. *Proceedings of the 12th International Conference on World Wide Web ACM*, New York, NY, USA. ISBN:1-58113-680-3.

Kolodner, J.L, et al. (2005). Case-based reasoning-inspired approaches to education. *The Knowledge Engineering Review*. 20(3), pp. 299-303.

Mantaras, et. al. (2006). Retrieval, reuse, revision and retention in case-based reasoning. *The Knowledge Engineering Review*. 20(3), pp. 215-240.

Patterson, D., Rooney, N., & Galushka, M. (2003). Efficient retrieval for case-based reasoning. http://www.aaai.org/Library/FLAIRS/2003/flairs03-029.php.

Reyes, R. L. & Sison, R. C. (2002). Case retrieval in CBR-Tutor. Published in Proceedings of the *International Conference on Computers in Education* (ICCE'02). IEEE Computer Society Washington, DC, USA.

Reyes, R.L. & Sison, R. C. (2002). Using case-based reasoning approach in planning instructional activities. Published in Proceedings of the *Seventh Pacific Rim International Conference on Artificial Intelligence (PRICAI 2002)*. Springer-Verlag; pp. 502-511.

Saad, A. (2011). *A case based system for lesson plan construction*. Submitted as PhD Thesis, Loughborough University, UK.

Spalazzi, L. ( 2001). A survey on case-based planning. *Artificial Intelligence Review.* 16; 3–36.

Wang, W.M. et al. (2007). Knowledge-based treatment planning for adolescent early intervention of mental healthcare: A hybrid case-based reasoning approach. *Expert Systems*, 24(4).

Watson, I. & Marir F. (1994). Case based reasoning: A review. *The Knowledge Engineering Review*. 9(4), pp. 327-354.