# Combining Content-based and Collaborative Filters in Learning Object Selection

Noppamas Pukkhem
Department of Computer and Information Technology
Faculty of Science, Thaksin University
Phattalung, Thailand 93110
noppamas@tsu.ac.th

***Abstract*— The explosive growth of online digital learning resources that commonly referred to as "*learning objects*" in e-learning community, demands effective recommendation solutions. We explore the hybrid techniques, which are by combining the content-based with the conditions of the learning object attributes and collaborative filtering with the condition of a learner similarity, to recommend the most suitable learning object to learners. The preferred content-based algorithm and nearest neighbor-based algorithm has been developed, combining with ranking method to strengthen predictive results. A learning object raking example is discussed to demonstrate the method implementation by using setting experiment. This result shows that the combining method can reduce the error rate of learner dissatisfaction.**

***Keywords- Collaborative filtering recommendation, Content-based recommendation, learning object, learner model***

## I. INTRODUCTION

Learning Objects offer a new way of thinking about learning content. Actually, learning objects can be educational components presented in any format. Learning objects are commonly stored in learning object repositories which facilitate various functions, such as learning object creation, submission, search, comment, review, etc. Rapidly evolving internet and web technologies have unlocked using learning objects in Learning Management System (LMS), such as Blackboard [1], Moodle [2], ATutor [3] or dotLRN [4]; these represent integrated systems which offer support for a wide area of activities in the e-learning process. These systems provide instructors can create the courses and test suites, for communicating with the learners, for monitoring and evaluating their work. Learners can learn, communicate and collaborate by means of LMS. However, they do not offer personalized services and it due to the "one-size-fit-all" problem. All learner being given access to the same set of learning objects and tools without taking into account the difference in interest, prior knowledge, experience, motivation and goals. This gives result in lack of learner information to perform accurate prediction of the most suitable learning object. Researchers have tried to find out how learners learn; it is a part of this work to provide a pattern of learner with their learning style that can be used in the recommendation model.

Our focus is on building the recommendation method on the basis of learners' learning style. The learners' learning styles is used as an adaptation criterion that different learners have their distinctive characteristics and learning styles, since it is one of the individual difference that play an important role in learning, according to educational field. Learning style refers to the individual manner in which a person approaches a learning task.

## II. LEARNING OBJECT AND LEARNER MODEL

### *2.1. Learning Object*

Learning object is a digital learning resource that facilitates a single learning objective and which may be reused in a different context. Many researchers defined different definition of learning object as follows:

- "Any entity, digital or non-digital, that may be used for learning, education or training." –IEEE 1484.12.1-2002. July, 15 2002, Draft Standard for Learning Object Metadata, IEEE Learning Technology Standards Committee (LTSC) [5].
- "Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy," in D. A. Wiley, ed., *The Instructional Use of Learning Objects: Online Version* [6]

In our experiment we use the IEEE LOM standard [7] of learning objects to be used as feature criteria of recommendation algorithms and define the set of its feature as following:

***Definition 1**: Learning Object Set $LOS_{LO}$ is the discrete set of all learning object feature. $LOS_{LO}$ is denoted by* $LOS_{LO} = \{F_i \mid \forall F_i \in LOM_F, F_i \neq F_j\}$ .

For example, from the selected learning object features in our work as following in Table 1:

We can define $LOS_{LO01} = \{ F1, F2, F3, F4, F5\} = \{ animation, active, 4, 8, simulation \}$.

Table 1: The learning object feature criterion.

| Feature ID | Name | Element Path | Value Space |
|---|---|---|---|
| F1 | Format | LOM/Technical/ Format | Video, Image, Text, Audio, Animation |
| F2 | Interactivity type | LOM/Education/Interactivity_Type | Active, Expositive, Mixed |
| F3 | Interactivity level | LOM/Education/Interactivity_level | Very low(0), Low(1), Medium(2), High(3), Very high(4) |
| F4 | Semantic density | LOM/Education/Semantic_Density | Very low(5), Low(6), Medium(7), High(8), Very high(9) |
| F5 | Learning resource type | LOM/Education/Learning_Resource_Type | Exercise Simulation, Experiment, Definition, Algorithm, Example |

*2.2. Learner Model*

A learner model is the model constructed from observation of interaction between a learner and learning system of instructional environment. Before construct the learner model, we have to know about the information about learners.

The Index of Learning Styles (ILS)- is an instrument use to assess preferences on four dimension of learning style model [8]. Each dimension of the ILS as a two-pan scale, with each pan representing one of the two categories of the dimension (**A**ctive/**R**eflective, **S**ensing /**I**ntuiting, vis**U**al, ver**B**al, se**Q**uential/**G**obal), and weights in a pan representing skills associated with that category. If you have a preference for sensing, it means you have more weights in the sensing pan than the intuitive pan, and conversely if you have a preference for intuition. The ILS-Thai version was administered to all participation. Learners were asked to complete the self-administered questionnaire. Therefore for this model we have defined the definition of it as following:

***Definition 2:*** *Learner Style Set* $LSS = \{(P_i, Pw_i)\} \mid P1 \in \{A/R\}, P2 \in \{S/I\}, P3 \in \{U/B\}, P4 \in \{Q/G\}\}$ *and* $Pw_i$ *is the weight which has interval [0-1] of each* $P_i$ .

All learners L by Definition 2, we define LSS(L) ∈ LSS. For example, for a particular learner $L_1$ we might have $LSS(L_1)= \{(A,1), (S,0.5), (I,0.5), (U,1,), (G,1)\}$

## III. Proposed Framework and Strategies

In our work, the hybrid recommendation technique is used to solve the problem of content-based (incomplete learning object or learner information) and collaborative recommendation (cold start problem). A system can recommend learning object according to a learner's preferences will attract the learner to come back for more. The architecture of our hybrid recommendation model for specific learner is presented in Figure 1 and the detail will be described in subsections below.

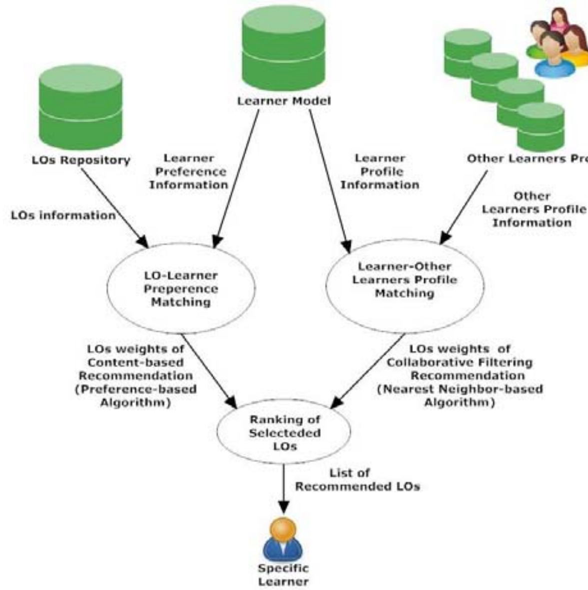



Figure 1. The architecture of hybrid recommendation.

*3.1. Concepts Selection*

Identifying a concept map and manageable groupings of contents is the first step in developing learning object recommendation. We developed a multiple instructor collaborative model that solving the different designs of various course maps and supporting a multi-agent based learning object recommendation system [9].

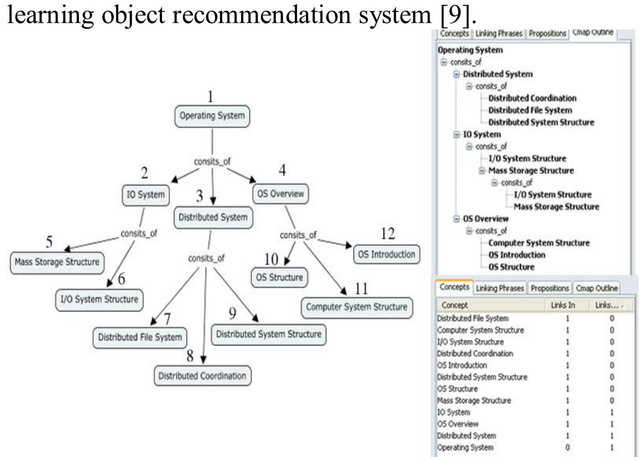



Figure 2. The example of concept map

In order to do recommendation process, we exploit the techniques of Goldsmith's closeness index [10] and ontology-based explaining for the course concept map generation. For example the integrated concept map of Operating System course is presented in Figure 2.

*3.2 Learning Object Adaptation Rule*

In each selected concept, the instructor imports related learning object and it's metadata for generating the LOS set of each learning object. The Learner Style Set (LSS) will be considered with learning object adaptation rule for crating the Learner Preference Set (LPS).

***Definition 3:*** *Learner Preference Set LPS is the set of learning object features which specific learner prefer with preferring weight.*

$$LPS = \{(PF_i, Fw_i) | PF_i \in F_i, Fw_i \in [0, 1]\}$$

For generating the LPS set, we developed the learning object adaptive rules for matching the learner preference to suitable features of learning object (LO-learner preference matching). The examples of some rules in our approach are presented as follows:

*Examples of Learning Object Adaptive Rule*

**Rule 1.** Adapt learning object for "**A-A**ctive" learner
If "**A**" ∈ LSS(L)
Then Lom.educational.LearningResourceType = *exercise* or *simulations* or *experiment*

**Rule 2.** Adapt learning object for "**R-R**efective" learner
If "**R**" ∈ LSS(L)
Then Lom.educational.ResourceType = *definition* or *algorithm* or *example*
. . .

From the rules presented above, the LPS can define as $LPS_{001} = \{(\{exercise, simulations, experiment\},1), (\{exercise, simulation, experiment\}, 0.5), \{definition, algorithm, example\}, 0.5), (\{video, image, animation, active, 3, 4\},1), (\{7, 8, 9\},1)\}$. This LPS will be used as input value in the recommendation algorithm in next subsection.

*3.3 Hybrid Recommendation Algorithm*

The hybrid recommendation algorithm is combine two approaches of recommendation technique. First, the content-based that we applied to the preference-based algorithm. Another is collaborative filtering technique that we implemented as nearest neighbor-based algorithm.

*3.3.1 Preference-based Algorithm*

This algorithm is used to compute the preference score (PS) that specific learner prefer to each learning object. It shows the relation between content (LO) and learners in mathematic computational.

**ALGORITHM:** Preference-based Algorithm

**INPUT:** Specific learner preference set *(LPS)*
Specific learning object set *(LOS)*
**OUTPUT :** Preference Score *(PS)* of specific *LO*

```
FUNCTION: Preference_Score_Calculation()
FOR EACH  LOS of learning object i
INT PS = 0
    FOR EACH  PF_i ∈ LPS (L)
            IF  ( PF_i = F_i)
            THEN  PS = FW_i +PS
            BREAK
    RETURN  Preference_Score_Calculation()=PS/10

END FUNCTION
```

*3.3.2 Nearest Neighbor-based Algorithm*

Because of the problem of content-based recommendation approach that we have to know the information about learning objects before recommend them to the specific learner. In some situations, the uncompleted metadata filling when import learning objects to the system may occur. So, it hides some suitable learning objects from learner accessing.

We noted that the suggestions from other learner can solve this problem. The assumption is the learner who has the similar preference as the specific learner should has a higher probability for selecting the same learning object. For this reason, we integrated the collaborative filtering approach to our recommendation algorithm to strengthen the precise of recommendations. This algorithm is called "Nearest neighbor-based algorithm", it predicts how helpful a learning object will be for a learner by analyzing other similar learner's profile. Three main steps are carried out in the nearest neighbor-based algorithm.

*Step 1:* Collect the related learning object in the same concept by using the concept map that described in section *3.1*.
*Step 2:* Extract preferred learners of each related learning object.
*Step 3*: Compute the neighbor-based score (NS) of each learning object with following algorithm. The Euclidean distance measuring is used to calculate the distance between two learners.

The result of this algorithm is the average of ranking of the three most similarity neighbors between specific learner(SL) and preferred LO learners (PL). We normalized the weight of this value with discount from 1. So the neighbor score (NB) is 1 – MDIS, where MDIS in the mean of distance. The NB score will be assigned to each preferred learning object for the ranking method in next subsection.

**ALGORITHM:** Nearest Neighbor-based Algorithm

**INPUT:** Preferred Learning object ID
LSS of specific learner (SL)
LSS of preferred learner (PL) of preferred LO
n = number of learner style preference
k = number of nearest neighbors (k=3)

**OUTPUT :** Neighbor Score *(NS)* of preferred *LO*

**FUNCTION:** Neighbor_Score_Calculation()
FOR EACH *LSS of SL*
FLOAT DIS =0, MDIS=0
// compute the distance between SP and PL by using
// the learner style preference in ever dimension

FOR EACH *LSS of PL of preferred LO*
FOR EACH ($P_i$ in LSS)
$DIS_{(SL,PL)} = DIS_{(SL,PL)} + Sqr((P_{SL}\wedge 2)- (P_{PL}\wedge 2))$

// Ordering the 3 least distances of all PLs
FOR ALL $DIS_{(SL,PL)}$ between SL and PLs
Rank($DIS_{(SL,PL)}$)
RETURN Last 3 of $DIS_{(SL,PL)}$)
MDIS = SUM($DIS_{(SL,PL)}$)/3

RETURN *Neighbor_Score_Calculation()=1-MDIS*

**END FUNCTION**

### *3.3.3 Ranking Method*

The ranking method for learning object recommendation is developed by combining two algorithms that we proposed in previous sections. The ranking value (RV) is computed from combining of two scores by equation :

$$RV = \mu PS + (1-\mu)\ NS,$$

where $\mu$ is the tuning parameter. In general, the $\mu$ =0.5 is the optimal ratios for the two weights dynamically. RV will be assigned to learning object candidates and the learning object that has the highest score will be recommended to specific learner.

## VI. METHOD DEMOSTRATION AND RESULT

### *4.1 Participants and Methods*

In this study, we examined the learning style of students in major of Computer Science (CS), information technology(IT) at Thaksin University. The Index of Learning Styles (ILS)-Thai version was administered to all participation. Students were asked to complete the self-administered questionnaire at the end of one lecture period in the first semester. This instrument consisted of 44-item sentences in the Thai language, translated with permission from the English version.

Table 2 presents the example of overall result about their learning preference styles of 34 fourth-year students of IT major. The results show about their information and the level of preference weight in 3 levels: weak (0), mild (0.5) and strong (1).

For example, in Active/Reflective preference dimension in Table 2 shows the learners strongly prefer of Active is 61.8% and strongly prefer of Reflective is 38.2%.

Table 2. The overall result of learner preference analysis

| Preference Type | Preference Level | N [N=34] | % of Total N | Preference Type | Preference Level | N [N=34] | % of Total N |
|---|---|---|---|---|---|---|---|
| Active | Weak | 0 | 0% | Visual | Weak | 0 | 0% |
| | Mild | 13 | 38.2% | | Mild | 12 | 35.3% |
| | Strong | 21 | 61.8% | | Strong | 22 | 64.7% |
| Reflective | Weak | 0 | 0% | Verbal | Weak | 0 | 0% |
| | Mild | 21 | 61.8% | | Mild | 22 | 64.7% |
| | Strong | 13 | 38.2% | | Strong | 12 | 35.3% |
| Sensing | Weak | 0 | 0% | Sequential | Weak | 6 | 17.6% |
| | Mild | 19 | 55.9% | | Mild | 27 | 79.4% |
| | Strong | 15 | 44.1% | | Strong | 1 | 2.9% |
| Intuitive | Weak | 0 | 0% | Global | Weak | 1 | 2.9% |
| | Mild | 15 | 44.1% | | Mild | 27 | 79.4% |
| | Strong | 19 | 55.9% | | Strong | 6 | 17.6% |

This means the learner who has a same learning style can has a different level of preference and it can be used in weighing process in recommendation algorithm.

### *4.2 Computation Example*

The information of our experiment in section 4.1 was use as an input for our hybrid recommendation algorithm; the detail will be described as following.

Table 3. The weight level of each preference

| Learner ID | Level Learning Preference | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | A | R | S | I | U | B | Q | G |
| 001 | 3 | 1 | 2 | 2 | 3 | 1 | 1 | 3 |
| 011 | 3 | 1 | 3 | 1 | 3 | 1 | 2 | 2 |
| 027 | 3 | 1 | 3 | 1 | 3 | 1 | 3 | 1 |

3=strong (w=1), 2=mild (w=0.5), 1=weak (w=0), w=preference weight

We can generate the information about learners following Table 3 by using the definition 2 as:

$$LSS_{001} = \{(A,1), (S,0.5), (I,0.5), (U,1,), (G,1)\}$$
$$LSS_{011} = \{(A,1), (S,1), (U,1,), (Q,0.5), (G,0.5)\}$$
$$LSS_{027} = \{(A,1), (S,1), (U,1,), (Q,1)\}$$

Then, by using the learning object adaptation rules in section 3.2, we can define the LPS of leaner ID 001 with definition 3 as:

$LPS_{001}$ = *{ ({exercise, simulations, experiment },1), ({exercise, simulation, experiment}, 0.5), {definition, algorithm, example}, 0.5), ({video, image, animation, active, 3, 4},1), ({7, 8, 9},1)}*

For example, we use the concept of "Process" of Operating System course to demonstrate learning object recommendation for specific learner. The information about related learning object is represented as following:

$LOS_{001}$ = *{ animation , active, 4, 8, simulation }*
$LOS_{002}$ = *{ text , expositive, 2, 7, algorithm }*
$LOS_{003}$ = *{ video , active, 4, 7, definition }*

When use the Preference-based algorithm for computing the PS of each LO of Learner ID 001, the results are $PS(LO_{001}) = 0.65$, $PS(LO_{002})=0.2$ and $PS(LO_{003})= 0.5$.

Computation of Neighbor Score (NS) of each related learning objects will help to strengthen the recommendation for the specific learner. These algorithm starts with to collect the group of learners that prefer the same learning object, For example, if we have a set of learners (SelLO) who prefer the same learning object of each ID 001, 002 and 003.

$SelLO_1=\{L_{002}, L_{003}, L_{004}, L_{005}, L_{007}, L_{014}, L_{015}, L_{018}, L_{021}, L_{022}, L_{024}, L_{025}, L_{030}, L_{031}, \}$

$SelLO_2=\{L_{008}, L_{010}, L_{013}, L_{016}, L_{020}, L_{026}, L_{028}, L_{032}\}$

$SelLO_3=\{L_{006}, L_{009}, L_{011}, L_{012}, L_{017}, L_{019}, L_{023}, L_{027}, L_{029}, L_{033}, L_{034}\}$

For this process, if the specific learner is Learner ID 001(The information is shown in Table 3) the result shows in the Table 4.

Table: The NS scores of related LO for learner ID 001

| LO ID | Top 3 of Similar Learners | NS |
|---|---|---|
| 001 | $L_{005}, L_{018}, L_{030}$ | 0.647 |
| 002 | $L_{013}, L_{016}, L_{020}$ | 0.420 |
| 003 | $L_{012}, L_{019}, L_{034}$ | 0.520 |

The last process is to rank the order of learning object by using the combination score from two algorithms (Preference-based and Nearest Neighbor-based). The hybrid score (HS) of each learning object ID001, ID002 and ID003 in our computational example are 0.65+0.647= 1.297, 0.2+0.420=0642 and 0.5+0.520=1.02. So, the recommended order of learning object in the selected concept is ID001>ID003> ID002.

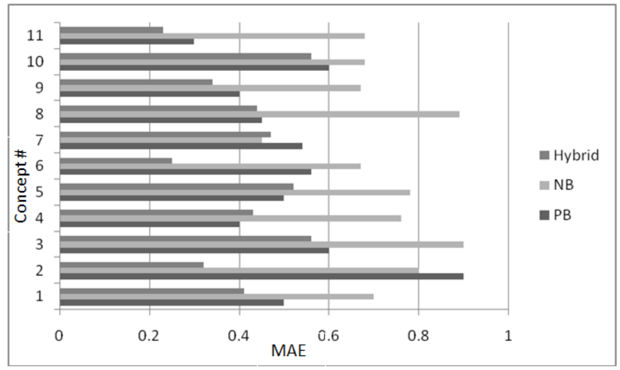
## V. Evaluation

The evaluation of our hybrid algorithm is to measure the Mean Absolute Error (MAE) of learner satisfaction between recommendation algorithm and learner self selection by using equation:

$$MAE = \frac{\sum_{i=1}^{K} | LO_S - LO_L |}{K}$$

where, $K$ is the number of comparison. The MAE result in experiment of 34 fourth-years IT's students with 11 concepts in Operating System course is shown in Figure 3.

The experiment shows the result that the hybrid algorithm can reduce the error rate of learning object selection about 5-30%.




Comparing MAE results of three algorithms

## VI. Conclusion

We propose hybrid recommendation algorithm that solving the problem in both of content-based and collaborative filtering approach. This algorithm works with our concept map combination model, which solving the different designs of various instructors for increasing collaboration among instructors in e-learning environment and supporting the learning object recommendation process. The results of learner satisfy in selecting the most suitable learning object when use the hybrid algorithm is higher than the two simple algorithms; it shows the performance of this method. The educational contribution of this research is a methodology for handling self-organized learning that specially provide for saving costs and the time of the learners.

## References

[1] Blackboard, Available at: http://www.blackboard.com , 2011.

[2] Moodle, Available at: http://moodle.org , 2011.

[3] ATutor, Available at: http://www.atutor.ca ,2011.

[4] dotLRN, Available at: http://dotlrn.org ,2011.

[5] IEEE, IEEE Standard for Learning Object Metadata, 1484.12.1-2002, 2002.

[6] D. A.Wiley, "Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy," in D. A. Wiley, ed., *The Instructional Use of Learning Objects: Online Version*, 2000.

[7] LOM, Final LOM Draft Standard, Available at: http://ltsc.ieee.org/wg12/20020612-Final-LOM-Draft.html , 2008

[8] R. M. Felder and R. Brent , Effective strategies for cooperative learning, *Journal of Cooperation & Collaboration in College Teaching* , 10(2), pp. 69-75, 2003.

[9] N. Pukkhem, W. Vatanawood, "The Algorithms for Collaborative "Course Concept Map" Decision Making in Learning Object Recommenders", *An International Journal of Research and Surveys*, Vol. 3, No. 3, 2009.

[10] T. E. Goldsmith and D. M. Davenport, Assessing structural similarity of graphs. In *Path_nder associative networks: studies in knowledge organization*, pp.75.87, 1990.