# Cluster to User Profile Ontology Mapping

Leyla Zhuhadar[1], Olfa Nasraoui[1], Robert Wyatt[2,2], and Elizabeth Romero[2]

[1] Knowledge Discovery and Web Mining Lab
Department of Computer Engineering and Computer Science
University of Louisville, Louisville, KY 40292, USA.
[2] Office of Distance Learning
Western Kentucky University, KY 42101, USA.

**Abstract.** In this paper, we present an approach that uses cluster analysis techniques to extend the ontology of an E-learning domain. This approach is significantly different from any current information retrieval systems, it uses a global ontology model that represents the whole E-learning domain combined with clusters' centroids vocabularies (terms) to extend the core ontology model. The most important advantage of clustering from the personalization perspective is that the clusters are later used as automatically constructed labels for each user profile. Hence, depending on the document collection and its evolution, both the user profiles and their underlying ontology labels are allowed to change or evolve accordingly. Our proposed approach has been implemented on the *HyperMany-Media*[1] platform at Western Kentucky University, USA.

## 1 Introduction

The main research question guiding this paper is whether it is feasible and beneficial to add the clusters' centroids to our E-learning ontology, while still being able to retrieve personalized learning resources that are satisfactory and effective for the learner. To achieve this objective, a cluster-based retrieval system was implemented. This system uses clustering techniques to divide the documents into an optimal categorization that is not influenced by the hand-made taxonomy of the colleges and course titles. The framework consisted of (1) clustering the documents (lectures) to discover more refined sub-concepts (top terms in each cluster) than provided by the available cluster and course taxonomy, (2) re-ranking the learner's search results based on the matching concepts in the learning content and the user profile, and (3) providing the learner with semantic recommendations during the search process, in the form of terms from the closest matching clusters of their profile. This approach used a combination of authoritatively supplied taxonomy by the colleges, with the data driven extraction (via clustering) of a taxonomy from the documents themselves, thus making it easier to adapt to different learning platforms, and making it easier to evolve with the document/lecture collection. In other words, clustering was a helpful technique to refine the college-based ontology, and also as a mechanism to "shake" the rigidness of an otherwise entirely manually constructed ontology that may not be appropriate for all users and for all

[1] http://hypermanymedia.wku.edu
[2] Director of Distance Learning, WKU, USA.

times. The most important advantage of clustering from the personalization perspective was that the clusters are later used as automatically constructed labels for each user profile. Hence, depending on the document collection and its evolution, both the user profiles and their underlying ontology labels are allowed to change or evolve accordingly.

## 2 Background

Cluster analysis is a process of grouping objects in groups where the similarity between the objects within the same group is greater than the similarity with the other groups [1]. "One application of clustering is the analysis of big text collections such as Web pages. The basic assumption, called *cluster hypothesis*, states that relevant documents tend to be more similar to each other than to non-relevant ones. If this assumption holds for a particular document collection, the clustering of documents based on similarity of their content may help to improve the search effectiveness [2]." In particular, the following improvements can be expected:

- Improving Search Recall: Search engines retrieve documents related to a specific query term. Generally, the same concepts can be expressed using different terms, thus searching for one of these terms will not retrieve the others. Clustering, which is based on overall similarity between the documents, can improve the recall since the search query will match an entire cluster instead of only one or more terms.
- Improving Search Precision: Assessing the relevance of documents to a query in a big collection of documents is considered a difficult task. Clustering those documents into smaller collections, ordering them by relevance, and returning only the most relevant group of documents, may help in finding a user's specific interests.

### 2.1 Clustering Documents

Applying clustering on a set of documents involves the following processes:

- Data Representation examples inculde the Vector Space Model (VSM), Metric Space Model (MSM), and Graph Model (GM)
- Similarity Measures (Inter-Object Similarity, Inter-Cluster Similarity)
- Clustering Algorithms (agglomerative, partitional)
- Evaluation and Validation

### 2.2 Data Representation

In order to cluster documents, they need first to be represented in a model. While a number of modeling representations are discussed in the literature, the most common ones are: the Vector Space Model (VSM), the Metric Space Model (MSM) and the Graph Model (GM). Among the three models, the Vector Space Model is the most ubiquitous [3]. Our focus is on VSM.

### 2.3 Similarity Measures

Two principles of measuring similarity can be considered: (i) Inter-Object Similarity and (ii) Inter-Cluster Similarity. The former deals with the similarity between two individual objects, while the latter deals with the similarity between the entire groups of objects. Table 1 lists some inter-object similarity measures while Table 2 lists inter-cluster similarity measures. Other different approaches of measuring similarities exist. For detailed review see [4,5,6].

### 2.4 Clustering Algorithms

Based on [6,1], clustering algorithms can be divided into two categories: agglomerative approaches [7,8,9,10] and partitional approaches [11,7,8]. Each criterion function uses a different methodology to produce the optimal clustering solution. In the case of *internal*, it searches for the best solution based only on the documents inside each cluster. In *external*, the focus is on finding the optimal solution in which the clusters are very different from each others. *Graph* models represent the documents as a graph and then finds the optimal solution. Finally, *hybrid* uses a mixture of criterion functions [8,12,13,14]. The criterion functions can use a choice of different similarity measures, as listed in Table 3.

- Hierarchical Agglomerative Algorithms: Agglomerative algorithms start by assigning each document to its own cluster; the goal is to find the pairs of clusters to be merged at the next step, and this can be done using classical approaches, such as single-link, weighted single-link, complete-link, weighted
- complete link, UPGMA, or using different criterion functions [14]: I1, I2, E1, G1, G1*, H1, H2, with each criterion measuring different aspects of intra-cluster similarity and inter-cluster dissimilarity Table 3.
- Partitional Clustering Algorithms: The goal is to find the clusters by partitioning the set of documents into a predetermined number of disjoint sets, each related to one specific cluster by optimizing various criterion functions [8,12,13,14]. Two methods of partitioning are very popular: (i) direct K-way clustering (similar to K-means), and (ii) repeated bisection or Bisecting K-Means (makes a sequence of bisection to find the best solution).

One of the differences between hierarchical agglomerative and partitional clustering algorithms is that the latter do not generate an agglomerative tree. The tree can be a very useful tool to discover the relationship between the documents in clusters at different levels of granularity. In 2002, [8] suggested the following solution to be used with the partitional clustering algorithms:

- For each cluster build an aggromerative tree of its documents,
- Combine these trees by creating an agglomerative tree whose leaves are discovered by the partitional clusters.

**Table 1.** Similarity Measures-I: Inter-Object Similarity.

| Inter-Object Similarity | Formula | |
|---|---|---|
| Metric Distance Coefficients | $\hat{s}(d_i,d_k)=\left[\sum_{j=1}^{m}\left\|d_{ij}-d_{kj}\right\|^r\right]^{1/r}: r\geq 1$ | Minkowski metric |
| | $\hat{s}_{Euclid}(d_i,d_k)=\sqrt{\sum_{j=1}^{m}\left(d_{ij}-d_{kj}\right)^2}=\left\|d_i-d_k\right\|_2: r=2$ | Euclidean distance |
| | $\hat{s}_{Manhattan}(d_i,d_k)=\sum_{j=1}^{m}\left\|d_{ij}-d_{kj}\right\|=\left\|d_i-d_k\right\|_1: r=1$ | Manhattan city block |
| | $\hat{s}_{Sup}(d_i,d_k)=max_{1\leq j\leq m}\left\|d_{ij}-d_{kj}\right\|=\left\|d_i-d_k\right\|_\infty: r\to\infty$ | Chebyshev |
| Association Coefficients | $s_{Dice}(d_i,d_k)=2\sum_{j=1}^{m}d_{ij}d_{kj}/\left(\sum_{j=1}^{m}d_{ij}^2+\sum_{j=1}^{m}d_{kj}^2\right)$ | Dice coefficient |
| | $s_{Jaccard}(d_i,d_k)=\sum_{j=1}^{m}d_{ij}d_{kj}/\left(\sum_{j=1}^{m}d_{ij}^2+\sum_{j=1}^{m}d_{kj}^2-\sum_{j=1}^{m}d_{ij}d_{kj}\right)$ | Jaccard coefficient |
| Cosine Similarity | $s_{Cosine}(d_i,d_k)=\frac{d_i^T d_k}{\|d_i\|_2\cdot\|d_k\|_2}=\sum_{j=1}^{m}d_{ij}d_{kj}/\sqrt{\sum_{j=1}^{m}d_{ij}^2\cdot\sum_{j=1}^{m}d_{kj}^2}$ | Cosine similarity |
| Statistical Coefficients | $s_{Pearson}(d_i,d_k)=\frac{1}{2}\left(\frac{(d_i-\bar{d}_i)(d_k-\bar{d}_k)}{\left\|d_i-\bar{d}_i\right\|_2\left\|d_k-\bar{d}_k\right\|_2}+1\right)$ | Pearson correlation coefficient |

**Table 2.** Similarity Measures-II: Inter-Cluster Similarity.

| Inter-Cluster Similarity | Definition |
|---|---|
| Centroid | Each cluster is represented by the mean |
| Medoid | Each cluster is represented by the most central object in the cluster, called medoid |
| Nearest-Neighbor | Single-linkage method: for each pair of clusters finds the nearest two vectors |
| Furthest-Neighbor | Complete-linkage method: for each pair of clusters finds the furthesttwo vectors |
| Group Average | Comparison between the mean of all vectors in each cluster is compared to the other cluster |
| Minimum Variance | Based on the smallest value of Information Loss (sum of square errors) |

**Table 3.** Summary of Various Clustering Criterion Functions.

| Criterion Function | Category | Optimization Function |
|---|---|---|
| I1 | Internal | $maximize\sum_{r=1}^{k}n_r\left(\frac{1}{n_r^2}\sum_{d_j,d_j\subset S_r}^{k}cos(d_i,d_j)\right)=\sum_{r=1}^{k}\|D_r\|^2$ |
| I2 | Internal | $maximize\sum_{r=1}^{k}\sum_{d_i\subset S_r}^{k}cos(d_i,C_r)=\sum_{r=1}^{k}\sum_{d_i\subset S_r}^{k}\frac{d_i^T C_r}{\|C_r\|}=\sum_{r=1}^{k}\|D_r\|$ |
| E1 | External | $minimize\sum_{r=1}^{k}n_r\frac{D_r^T D}{\|D_r\|}$ |
| G1 | Graph-Based, Hybrid | $minimize\sum_{r=1}^{k}\frac{D_r^T D}{\|D_r\|^2}$ |
| G1' | Graph-Based, Hybrid | $minimize\sum_{r=1}^{k}n_r^2\frac{D_r^T D}{\|D_r\|^2}$ |
| H1 | Hybrid | $maximize\ \frac{I_1}{E_1}$ |
| H2 | Hybrid | $maximize\ \frac{I_2}{E_1}$ |

### 2.5 Evaluation and Validation

In the special case where external class labels are available for the input data, the quality of a clustering solution can be measured using the *Entropy* [8].

**Definition:** Entropy is calculated as the weighted average of the entropies of the k individual clusters, each weighted in proportion to its cluster size.

$$Entropy = \sum_{r=1}^{k} \frac{n_r}{n} E(S_r) \tag{1}$$

For a specific cluster S, of size n, the entropy of this cluster is defined [8] as:

$$E(S_r) = \frac{1}{log\ q} \sum_{i=1}^{q} \frac{n_r^i}{n_r}\ log\ \frac{n_r^i}{n_r} \tag{2}$$

where q= number of classes in the dataset, and $n_r^i$= number of documents of the $i^{th}$ class that were assigned to the $r^{th}$ cluster.

## 3 Implementing Cluster-based Semantic Profiles

### 3.1 Generating Cluster-based Semantic Profiles

We compared different hierarchical algorithms for a dataset consisting of 2,812 documents using the clustering package Cluto[3]. We repeatedly applied each clustering algorithm with all possible combinations of clustering criterion functions for different numbers of clusters: 20, 25, 30, 35, 40. By considering each college as one broad class (thus 10 categories), we tried to ensure that the clusters are as pure as possible, i.e. each cluster contains documents mainly from the same category. However, since a class may be partitioned into several clusters (as was the case here), the clusters are more refined versions of the college categories, which is our desired aim.We used the *cluster entropy measure* [8,12,13,14] to evaluate the quality of each clustering solution. Then the *entropy* of the entire partition [8,12,13,14]. We implemented three different clustering algorithms that are based on the agglomerative, partitional, and graph partitioning paradigms [14]. In agglomerative algorithms, starting from assigning each document to its own cluster, the goal is to find the pair of clusters to be merged at the next step, and this can be done using known approaches, such as single-link, weighted single-link, complete-link, weighted complete link, UPGMA or others, using different criterion functions [14]: I1, I2, E1, G1, G1*, H1, H2, with each criterion measuring different aspects of intra-cluster similarity and inter-cluster dissimilarity.

## 4 Evaluation

From our experiments, we found, as shown in Table 4, the best performing criterion to be the H2 (given below), with $u$ and $v$, being documents and $S_i$ being the $i^{th}$ cluster, containing $n_i$ documents, while $sim(u,v)$ denotes the similarity between $u$ and $v$ [8,12,13,14].

$$H2 = \frac{I2}{\sum_{i=1}^{k} n_i \sqrt{\frac{\sum_{v,u \in S_i} sim(v,\ u)}{\sum_{u,v \in S_i} sim(v,\ u)}}} \tag{3}$$

[3] http://glaros.dtc.umn.edu/gkhome/cluto/cluto/overview

In partitional clustering algorithms, the goal is to find the clusters by partitioning the set of documents into a predetermined number of disjoint sets, each related to one specific cluster by optimizing various criterion functions [8,12,13,14].

We also experimented with two partitional algorithms, direct K-way clustering (similar to K-means), and repeated bisection or Bisecting K-Means, which makes a sequence of bisections (running K-means with K=2 clusters) to find the best solution; and experimented with all criterion functions. For direct Kway, I2 [8,12,13,14] performed best, whereas H1 [8,12,13,14] performed best for repeated bisection, as shown in Table 4. I2 and H1 are given below.

$$I2 = \sum_{i=1}^{k} \sqrt{\left( \sum_{u,v \varepsilon S_i} sim(v,u) \right)} \tag{4}$$

$$H1 = \frac{\sum_{i=1}^{k} \frac{1}{n_i} \sqrt{\left( \sum_{u,v \varepsilon S_i} sim(v,\, u) \right)}}{\sum_{i=1}^{k} n_i \sqrt{\frac{\sum_{v \varepsilon S_i, u \varepsilon S_i} sim(v,\, u)}{\sum_{u,v \varepsilon S_i} sim(v,\, u)}}} \tag{5}$$

We also experimented with graph-partitioning-based clustering algorithms which use a sparse graph to model the affinity relations between different documents, and then discover the desired clusters by partitioning this graph [?] [15]. Of all the algorithms mentioned so far, graph-partitioning produced the best clustering results as shown in Table 4, with 35 clusters and the lowest entropy.

**Table 4.** Clustering Entropy Measures for various algorithms (rows) and partioning criteria (columns).

| Agglomerative Methods | | | | | |
|---|---|---|---|---|---|
| *I1* | *I2* | *E1* | *G1* | *G1∗* | *H1* |
| 0.040 | 0.025 | 0.039 | 0.102 | 0.043 | 0.024 |
| *H2* | *Slink* | *WSLink* | *Clink* | *WCLink* | *UPGMA* |
| 0.023 | 0.493 | 0.493 | 0.060 | 0.060 | 0.067 |
| Direct k-way Methods | | | | | |
| *I1* | *I2* | *E1* | *G1* | *G1∗* | *H1* |
| 0.036 | 0.020 | 0.040 | 0.067 | 0.055 | 0.038 |
| *H2* | *Slink* | *WSLink* | *Clink* | *WCLink* | *UPGMA* |
| 0.037 | - | - | - | - | - |
| Repeated Bisection Methods | | | | | |
| *L1* | *L2* | *E1* | *G1* | *G1∗* | *H1* |
| 0.027 | 0.034 | 0.036 | 0.058 | 0.036 | 0.022 |
| Graph Partitional Methods | | | | | |
| *pe* | $pG_1$ | *pH1* | *pH2* | *pI1* | *pI2* |
| 0.033 | 0.051 | 0.042 | 0.01 | 0.32 | 0.017 |
| *H2* | *Slink* | *WSLink* | *Clink* | *WCLink* | *UPGMA* |
| 0.032 | - | - | - | - | - |

Graph partitioning of the entire collection into 35 clusters generated the confusion matrix shown in Table 5, with only 41 misclassified documents out of 2812 (~1%).We relabeled each cluster, based on the majority of assigned documents in each college and from each course, as follows: college-name\ coursename, as shown in the last column in Table 5.

### 4.1 Cluster to Profile Ontology Mapping

Each learner's profile Ui is considered as a set D of documents docs(Ui) = [l k=1dki . The domain clusters CL = [nk =1CLk are obtained from the clustering in section 3.1. The mapping procedure, shown in Algorithm 1, measures the similarity Sim(D;CLi) between the learner profile documents and each cluster description (frequent terms). The most similar cluster is considered as a recommended cluster. The recommended cluster has two effects on our searching mechanism: first, on the re-ranking algorithm, and second, on the learner's semantic term recommendation, more details about this work can be found in [16].

**Table 5.** Cluster to Category (10 colleges) Confusion Matrix (majority based college/course assignment and labeling).

| | Classified | miss-Classified | re-label |
|---|---|---|---|
| CL0 | 29 (English) | 2 | English\Introduction to Lit |
| CL1 | 38 (Social Work) | 4 | Social Work\344 |
| CL2 | 29 (Math) | 2 | Math\History |
| CL3 | 77 (Communication disorders) | 3 | Communication disorders\voice |
| CL4 | 58 (Math) | 2 | Math\smfap1 |
| CL5 | 86 (History) | 3 | History\Western civil |
| CL6 | 77 (Communication disorders) | 2 | Communication disorders |
| CL7 | 29 (Social work) | 4 | SW\344 |
| CL8 | 77 (Communication disorders) | 2 | Communication disorders |
| CL9 | 67 (Communication disorders) | 3 | Communication disorders\voice |
| CL10 | 67 (Chemistry) | 2 | Chemistry |
| | | ... | |
| CL34 | 96 (History) | 1 | History\Western civil |
| Total | 2771 | 41 | 2812 |

**Algorithm 1** Best Cluster Mapping algorithm for a learner $U_i$

Input: $D = \cup_{k=1}^{l} d_{ki}$;//$l$ = # *of visited docs*
Output: *BestCluster* ;// *most similar cluster*
CL = $\cup_{k=1}^{n} CL_k$;// $n$ = # *of clusters*
*BestCluster* =$CL_1$
foreach $CL_i \in CL$
if *Sim*($D$, $CL_i$) > *Sim*($D$, *BestCluster*) then
*BestCluster* = $CL_i$
end

### 4.2 Changing the Learner's Semantic Profile

After extracting the most similar cluster $C_i$ = *BestCluster* (recommended-cluster), which is summarized by the $Top_n$ keywords (significant or frequent terms), we modified the learner's semantic ontology (in the OWL description) accordingly, by adding the cluster's terms as semantic terms under the concepts (parent nodes) that these documents belong to. Fig 1 is a mountain view visualization of the clustering solutions, in addition to the features that represent a descriptive information about each cluster, as also shown in Table 6. These features have been used in our ontology.

## 5 Conclusion and Future Works

In this paper we presented an approach that used cluster analysis techniques to extend the ontology of E-learning domain. This approach used a combination of an authoritatively supplied taxonomy by the colleges, with the data driven extraction (via clustering) of a taxonomy from the documents themselves. In other words, clustering was used to refine the college-based ontology, and also as a mechanism to "shake" the rigidness of an otherwise entirely manually constructed ontology that may not be appropriate for all users and for all times. The most important advantage of clustering from the personalization perspective was that the clusters can later been used as automatically constructed labels for each user profile. Hence, depending on the document collection and its evolution, both the user profiles and their underlying ontology labels are allowed to change or evolve accordingly. Our proposed approach has been implemented on the *HyperManyMedia*[4] platform. Our future plan is to merge this repository with as many external open source resources as we can accommodate, such as MIT OpenCourseWare[5] and BerkeleyWebcast[6]. This can be realized in the following phases:

– Collecting similar courses/lectures located on these external repositories
– Parsing the Metadata of those learning objects
– Creating local plugins for the current search engine that accommodate the external Metadata [17]
– Downloading all the externals learning objects and extracting the documents (Text only version)
– Clustering those documents using the same document clustering techniques that we used in section 2.4
– Building an Extended Ontology Structure [18]
– Adding the centroids of the clusters to the Extended Ontology Structure under their appropriate Concepts/Subconcepts
– Crawling and indexing the local domain ("HyperManyMedia" platform) and the external domains
– Collecting users queries and, users activities
– Building Users Models
– Coping with Concept Drift [19]
– Re-evaluating the System

[4] http://hypermanymedia.wku.edu
[5] MIT OpenCourseWare: http://ocw.mit.edu/OcwWeb/web/home/home/index.htm
[6] Berkeley Webcast: http://webcast.berkeley.edu/

**Table 6.** Cluster Descriptions.

| Cluster | Descriptions |
|---|---|
| CL0 (English\Introduction to Lit ) | western course english literature online time challenge setting years face |
| CL1 (Social Work\344) | probability population hypothesis difference sample score curve percentile rank deviation |
| CL2 (Math\History) | angle triangle congruent isosceles proposition divisor prime theorem square circle |
| CL3(Communication disorder\voice) | phonological consonant sound language therapy vocabulary patient thematic stimulation aphasia |
| CL4(Math\smfap1) | factor polynomial negative set complex solutions base prime number product |
| CL5 (History\West civil) | divisor equation diophantine negative solution probability experiment fibonacci product square |
| CL6 (Communication disorder) | vocal muscle thyroid larynx voice pitch disorders treatment nerve laryngeal |
| CL7 (Social Work\344) | laser material machining electrical sheet milling cutting tool cutter spindle |
| CL8 (Communication disorder) | child narration expository pragmatics cognitive sentence phrase syntax vowel metalinguistics |
| CL9 (Communication disorder\voice) | point circle side angle segment line divisibility congruent prime sigma |
| CL10 (Chemistry) | fraction metal compound charge copper hydrogen oxygen hydroxide aspirin solution |
| ... | |
| CL34 (History\West civil) | france king spanish parliament james cromwell louis dutch russia frederick napoleon |

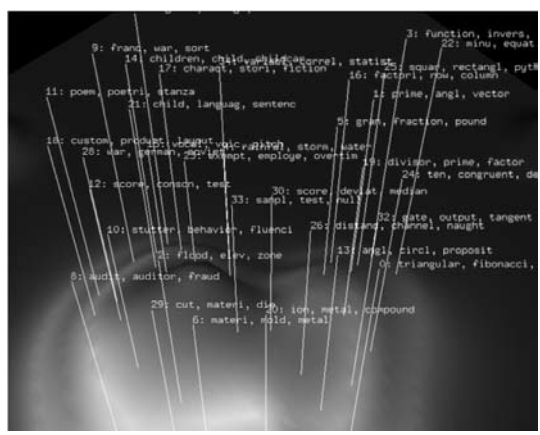


**Fig. 1.** Mountain Visualization with Features.

## References


1. Pang-Ning, T., Steinbach, M., Kumar, V.: Introduction to data mining. Boston: Person Addison Wesley EducatioPress (2005)
2. Feldman, R., Sanger, J.: The text mining handbook. Cambridge University Press (2006)
3. McGill, M., Salton, G.: Introduction to modern information retrieval. McGraw-Hill (1983)
4. Romesburg, C.: Cluster Analysis for Researchers. Lulu. Com (2004)
5. Kaufman, Rousseeuw: Finding Groups in Data: An Introduction to Cluster Analysis. Wiley (1990)
6. JAIN, A., MURTY, M., FLYNN, P.: Data Clustering: A Review. ACM Computing Surveys 31(3) (1999)
7. Steinbach, M., Karypis, G., Kumar, V.: A comparison of document clustering techniques (2000)
8. Zhao, Y., Karypis, G., SCIENCE, M.U.M.D.O.C.: Comparison of Agglomerative and Partitional Document Clustering Algorithms. Defense Technical Information Center (2002)
9. Zamir, O., Etzioni, O.: Web document clustering: a feasibility demonstration. In: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, ACM Press New York, NY, USA (1998) 46–54
10. El-Hamdouchi, A., Willett, P.: Comparison of Hierarchic Agglomerative Clustering Methods for Document Retrieval. The Computer Journal 32(3) (1989) 220–227
11. Zhao, Y., Karypis, G.: Soft clustering criterion functions for partitional document clustering: a summary of results. In: CIKM '04: Proceedings of the thirteenth ACM international conference on Information and knowledge management, New York, NY, USA, ACM (2004) 246–247
12. Zhao, Y., Karypis, G., Fayyad, U.: Hierarchical clustering algorithms for document datasets. Data Mining and Knowledge Discovery 10(2) (2005) 141–168
13. Karypis, G.: Evaluation of item-based top-n recommendation algorithms. In: Proceedings of the tenth International conference on Information and knowledge management, ACM New York, NY, USA (2001) 247–254
14. Zhao, Y., Karypis, G.: Evaluation of hierarchical clustering algorithms for document datasets. Proceedings of the eleventh international conference on Information and knowledge management (2002) 515–524
15. Karypis, G., Aggarwal, R., Kumar, V., Shekhar, S.: Multilevel hypergraph partitioning: applications in vlsi domain. Very Large Scale Integration (VLSI) Systems, IEEE Transactions on 7(1) (1999) 69–79
16. Zhuhadar, L., Nasraoui, O.: Personalized cluster-based semantically enriched web search for e-learning. (2008)
17. Zhuhadar, L., Nasraoui, O., Wyatt, R.: Metadata domain-knowledge driven search engine in "hypermanymedia" e-learning resources. In: CSTST '08: Proceedings of the 5th international conference on Soft computing as transdisciplinary science and technology, New York, NY, USA, ACM (2008) 363–370
18. Zhuhadar, L., Nasraoui, O., Wyatt, R.: Visual ontology-based information retrieval system. In: Information Visualisation, 2009 13th International Conference. (July 2009) 419–426
19. Zhuhadar, L., Nasraoui, O., Wyatt, R.: Dual representation of the semantic user profile for personalized web search in an evolving domain. In: Proceedings of the AAAI 2009 Spring Symposium on Social Semantic Web, Where Web 2.0 meets Web 3.0. (2009) 84–89