Cluster-Lift Method for Mapping Research Activities over a Concept Tree

Boris Mirkin, Susana Nascimento, and Luís Moniz Pereira

Abstract The paper builds on the idea by R. Michalski of inferential concept interpretation for knowledge transmutation within a knowledge structure taken here to be a concept tree. We present a method for representing research activities within a research organization by doubly generalizing them. To be specific, we concentrate on the Computer Sciences area represented by the ACM Computing Classification System (ACM-CCS). Our cluster-lift method involves two generalization steps: one on the level of individual activities (clustering) and the other on the concept structure level (lifting). Clusters are extracted from the data on similarity between ACM-CCS topics according to the working in the organization. Lifting leads to conceptual generalization of the clusters in terms of "head subjects" on the upper levels of ACM-CCS accompanied by their gaps and offshoots. A real-world example of the representation is provided.

Key words: Cluster-lift method; additive clustering; concept generalization; concept tree; knowledge transmutation.

Boris Mirkin
School of Computer Science, Birkbeck University of London,
London, UK WC1E 7HX, e-mail: mirkin@dcs.bbk.ac.uk

Susana Nascimento
Computer Science Department and Centre for Artificial Intelligence (CENTRIA)
Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa
Caparica, Portugal, e-mail: snt@di.fct.unl.pt

Luís Moniz Pereira
Computer Science Department and Centre for Artificial Intelligence (CENTRIA)
Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa
Caparica, Portugal, e-mail: lmp@di.fct.unl.pt

1 Introduction: Inductive Generalization for Concept Interpretation

In his work on inferential learning theory [5, 6], R. Michalski pointed out the importance of knowledge transmutation defined as the process of deriving desirable knowledge from a given input and background knowledge. He envisioned that knowledge transmutation can be performed in terms of pairs of operations such as

> selection vs. generation, replication vs. removal, reformulation vs. randomization, abstraction vs. concretion, similization vs. dissimilization, and generalization vs. specialization.

[6], p. 3. In this paper, we would like to draw attention to the possibility of formalizing the generalization step within the framework of knowledge represented by a concept tree, such as decision trees advocated by R. Michalski in the framework of conceptual clustering [7, 17]. Concept trees currently are well recognized knowledge structures being important part of ontologies, taxonomies and other forms of knowledge representation.

Consider, for example, ACM Computing Classification System (ACM-CCS), a conceptual four-level classification of the Computer Science subject area, built to reflect the vast and changing world of computer oriented writing. This classification was first published in 1982 and then thoroughly revised in 1998, and it is being updated since [1]. ACM-CCS comprises eleven major partitions (first-level subjects):

A. General Literature
B. Hardware
C. Computer Systems Organization
D. Software
E. Data
F. Theory of Computation
G. Mathematics of Computing
H. Information Systems
I. Computing Methodologies
J. Computer Applications
K. Computing Milieux

These are subdivided into 81 second-level subjects. For example, item *I. Computing Methodologies* consists of eight subjects:

I.0 GENERAL
I.1 SYMBOLIC AND ALGEBRAIC MANIPULATION
I.2 ARTIFICIAL INTELLIGENCE
I.3 COMPUTER GRAPHICS
I.4 IMAGE PROCESSING AND COMPUTER VISION
I.5 PATTERN RECOGNITION
I.6 SIMULATION AND MODELING (G.3)
I.7 DOCUMENT AND TEXT PROCESSING (H.4, H.5)

which are further subdivided into third-layer topics as, for instance, *I.5 PATTERN RECOGNITION* which consists of seven topics:

I.5.0 General
I.5.1 Models
I.5.2 Design Methodology
I.5.3 Clustering

Algorithms
Similarity measures

I.5.4 Applications
I.5.5 Implementation (C.3)
I.5.m Miscellaneous

These are further subdivided in unlabeled subtopics such as those two shown for topic *I.5.3 Clustering*.

As can be seen from the examples above, there are a number of collateral links between topics both on the second and the third layers - they are in the parentheses in the ends of some topics such as *I.6*, *I.7*, and *I.5.5* above.

Concept tree structures such as the ACM-CCS are used, mainly, as devices for annotation and search for documents or publications in collections such as that on the ACM portal [1]. However, being adequate domain ontologies, concept trees can and should be used for other tasks as well. For example, the ACM-CCS tree has been applied as:

– A gold standard for ontologies derived by web mining systems such as the CORDER engine [18];
– A device for determining the semantic similarity in information retrieval [9] and e-learning applications [21];
– A device for matching software practitioners' needs and software researchers' activities [2].

Here we concentrate on yet another application of ACM-CCS, mapping research activities in Computer Sciences. However, our method can be utilized in other knowledge domains as well. The method works for any domain if its structure has been represented with a concept tree. We propose the use of concept tree structures for representing activities of research organizations by using a two-stage generalization of individual research topics over the tree topology.

A concept tree such as the ACM-CCS taxonomy can be seen as a representative generic ontology, with its explicitly expressed hierarchical subsumption relation between subject classes along with the collateral relation of association between different nodes. The art of representation of various items on an ontology is of interest in many areas such as text analysis, web mining, bioinformatics and genomics. In web mining, representations are extracted from domain ontologies: the ontologies are used to automatically characterize usage profiles by describing user's interests and preferences for web personalisation [20]. There are also recommender systems for on-line academic research papers [8], which extract user profiles based on an ontology of research topics. In bioinformatics several clustering techniques have been successfully applied in the analysis of gene expression profiles and gene function prediction by incorporating gene ontology information into clustering algorithms [4].

However, this line of thinking has never been applied for representing the activities of research organizations. The very idea of representing the activities of a research organization as a whole may seem rather odd because conventionally it is only the accumulated body of results that does matter in the sciences, and these always have been and still are provided by individual efforts. The assumption of individual research efforts implicitly underlies systems for reviewing and comparing different research departments in countries such as the United Kingdom in which scientific organizations are subject to regular comprehensive review and evaluation practices. The evaluation is based on the analysis of individual researchers' achievements, leaving the portrayal of a general picture to subjective declarations by the departments [14]. Such an evaluation provides for the assessment of relative strengths among different departments, which is good for addressing funding issues. Yet there exists another aspect, that of the integral portrayal rather than comparative analysis of the developments. This aspect is important for decisions regarding long-term or wide-range issues of scientific development such as national planning or addressing the so-called 'South–North divide' between developed and underdeveloped countries. The latter would require comparing between integral systems of scope and capabilities of scientific organizations and university departments in both the South and North (see, for instance, The United Nations Millennium Project task force web-site [19]).

Representation of activities over the ACM-CCS concept tree can be used for:

1. Overviewing scientific subjects that are being developed in the organization.
2. Positioning the organization over ACM-CCS.
3. Overviewing scientific disciplines being developed in organizations over a country or other territorial unit, with a quantitative assessment of controversial subjects, for example, those in which the level of activity is not sufficient or the level of activities by far exceeds the level of results.
4. Assessing the scientific issues in which the character of activities in organizations does not fit well onto the classification; these can be potentially the growth points or other breakthrough developments.
5. Planning research restructuring and investment.

Similar lists of objectives can be drawn for the analysis of other activities.

2 Cluster – Lift Method

We represent a research organization by clusters of ACM-CCS topics to reflect communalities between activities of members or teams working on these topics. Each of the clusters is mapped to the ACM-CCS tree and then lifted in the tree to express its general tendencies. The clusters are found by analyzing similarities between topics which are derived from either automatic analysis of documents posted on web by the teams or by explicitly surveying the members of the department. The latter option is especially convenient in situations in which the web contents do not properly reflect the developments. If such is the case, a tool for surveying research activities of the members and teams is needed.

Accordingly, this work involves developing:

1. e-screen based ACM-CCS topic surveying device,
2. method for deriving similarity between ACM-CCS topics,
3. method for finding possibly overlapping topic clusters from similarity data, and
4. method for parsimoniously lifting topic clusters on ACM-CCS.

In the following subsections, we describe these four.

2.1 E-Screen Survey Tool

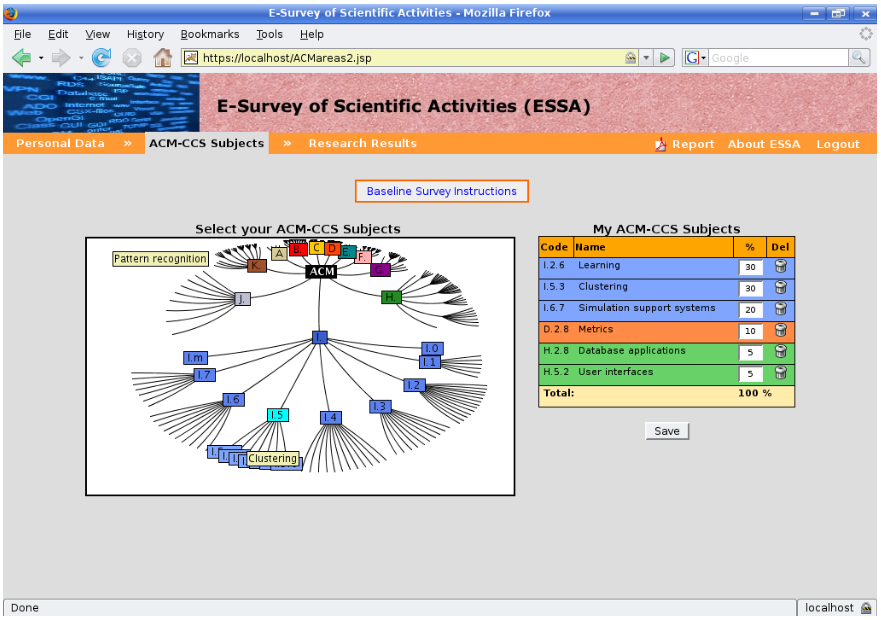
An interactive survey tool has been developed to provide two types of functionality: i) data collection about the research results of individual members, described in terms of the ACM-CCS topics; ii) statistical analysis and visualization of the data and results of the survey. The period of research activities comprises the survey year and the previous four years. This is supplied with interactive “focus + context” navigation functionalities [16]. The respondent is asked to select up to six topics among the leaf nodes of the ACM-CCS tree and assign each with a percentage expressing the proportion of the topic in the total of the respondent’s research activity. Figure 1 shows a screenshot of the interface for a respondent who has chosen six ACM-CCS topics during his/her survey session. Another, “research results” form allows to make a more detailed assessment in terms of individual research results of the respondent in categories such as refereed publications, funded projects, and theses supervised.

The leaf nodes of the ACM-CCS tree are populated thus by the respondent supplied weights, which can be interpreted as fuzzy membership degrees of the respondent’s activity with respect to ACM-CCS topics.

2.2 Deriving Similarity between ACM-CCS Topics

We define similarity between ACM-CCS topics i and j as the weighted sum of individual similarities. The individual similarity is just the product of weights f_i and f_j assigned by the respondent to the topics. Clearly, topics that are left outside of the individual’s list, have zero similarities with other topics.

We assign weights to the surveyed individuals too. An individual’s weight is inversely proportional to the number of subjects they selected in the survey. This smoothes out the differences between topic weights imposed by the selection sizes.


E-Survey of Scientific Activities - Mozilla Firefox
File Edit View History Bookmarks Tools Help
https://localhost/ACMareas2.jsp
Google
E-Survey of Scientific Activities (ESSA)
Personal Data » ACM-CCS Subjects » Research Results
Report About ESSA Logout
Baseline Survey Instructions
Select your ACM-CCS Subjects
Pattern recognition
A
B.
C
D
E.
F.
G.
H.
J.
K.
ACM
I.
I.0
I.1
I.2
I.3
I.4
I.5
I.6
I.7
I.m
Clustering
Save
Done
localhost

My ACM-CCS Subjects

Code	Name	%	Del
I.2.6	Learning	30	
I.5.3	Clustering	30	
I.6.7	Simulation support systems	20	
D.2.8	Metrics	10	
H.2.8	Database applications	5	
H.5.2	User interfaces	5	
Total:		**100 %**	

Fig. 1 Screenshot of the interface survey tool for selection of ACM-CCS topics.

It is not difficult to see that the resulting topic-to-topic similarity matrix $A = (a_{ij})$ is positive semidefinite.

2.3 *Finding Overlapping Clusters*

The topic clusters are to be found over similarity matrix $A = (a_{ij})$ with no conventional mandatory nonoverlapping condition imposed.

We employ the data recovery approach described in [10, 11] for the case of crisp clustering and in [13] for the case of fuzzy clustering. We consider only the crisp clustering case in this paper. We find clusters one by one as subsets of ACM-CCS leaf topics S maximizing criterion

$$g(S) = s^T As/s^T s = a(S)|S|. \tag{1}$$

where

1. $s = (s_i)$ denotes a binary membership vector corresponding to subset S so that $s_i = 1$ if $i \in S$ and $s_i = 0$, otherwise;
2. $a(S)$ is the average similarity a_{ij} within S and
3. $|S|$ is the number of topics in S.

Criterion (1) can be considered as a compromise between two contradicting criteria: (a) maximizing the within-cluster similarity and (b) maximizing the cluster size. When squared, the criterion expresses the proportion of the similarity data scatter, which is taken into account by cluster S according to the data recovery model described in [10, 11].

It should be pointed out that this criterion emerges not only in the data recovery framework but it also fits into some other frameworks such as (i) maximum density subgraphs [3] and (ii) spectral clustering [15].

We use a version of ADDI-S algorithm from [10] for locally optimizing criterion (1) that starts from singleton $S = \{i\}$ for a topic $i \in I$. Then the algorithm iteratively finds an entity j to move in or remove from S by maximizing $g(S \pm j)$ where $S \pm j$ stands for $S + j$ if $j \notin S$ or $S - j$ if $j \in S$. It appears that this can be done easily - just by comparing the average similarity between j and S, $a(j,S)$, with the threshold $\pi = a(S)/2$; the greater the difference, the better the j. The process stops when the change of the state of j with respect to S is not beneficial anymore, that is, if π is greater than $a(j,S)$ if $j \notin S$, or smaller than $a(j,S)$ if $j \in S$. In this way, by starting from each $i \in I$, ADDI-S produces a number of potentially overlapping or even coinciding locally optimal clusters S_i – of which that with the highest contribution is taken as the algorithm's output S.

Thus produced S is rather tight because each $j \in S$ has a high degree of similarity with S, greater than half the average similarity within S, and it is also well separated from the rest, because for each entity $j \notin S$, its average similarity with S is less than that.

Next cluster can be found with the same procedure applied to residual similarity matrix $A' = A - a(S)ss^T$. Its contribution to the initial data scatter is computed as g^2 where g is defined in (1) by using the residual matrix A' rather than A. More clusters can be extracted in a similar manner by using residual matrices obtained by subtraction of all the previously found clusters [10].

2.4 Parsimonious Lifting Method

To generalise the main contents of a cluster of topics, we translate it to higher layers of the taxonomy by lifting it according to the principle: if all or almost all children of a node in an upper layer belong to the cluster, then the node itself is taken to represent the cluster on a higher level of ACM-CCS taxonomy. Such a lift can be done differently leading to different portrayals of the cluster on ACM-CCS tree depending on the relative weights of accompanying events, "gaps" and "offshoots", as described below.

A cluster can fit quite well into the classification or not (see Fig. 2), depending on how much its topics are dispersed among the tree nodes.

The best possible fit would be when all topics in the subject cluster fall within a parental node in such a way that all the siblings are covered and no gap occurs. The parental tree node, in this case, can be considered as the head subject of the cluster.

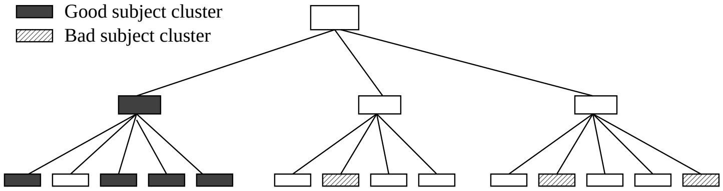


Fig. 2 Two clusters of second-layer topics, presented with checked and diagonal-lined boxes, respectively. The checked box cluster fits within one first-level category (with one gap only), whereas the diagonal line box cluster is dispersed among two categories on the right. The former fits the classification well; the latter does not fit at all.

A second best case is when one of the children does not belong to the cluster (a gap) or when one of the children is covered by a different parent (an offshoot). A few gaps, that is, head subject's children topics that are not included in the cluster, although diminish the fit, still leave the head subject unchanged. A larger misfit occurs when a cluster is dispersed among two or more head subjects (see Fig. 3). It is not difficult to see that the gaps and offshoots are determined by the head subjects specified in a lift.

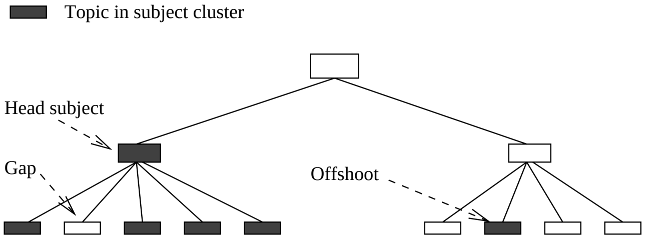


Fig. 3 Three types of features in mapping of a subject cluster to the ontology.

The total count of head subjects, gaps and offshoots, each type weighted accordingly, can be used for scoring the extent of the cluster misfit needed for lifting a grouping of research topics over the classification tree as illustrated on Fig. 4. The smaller the score, the more parsimonious the lift and the better the fit. When the topics under consideration relate to deeper levels of classification, such as the third layer of ACM-CCS, the scoring may allow some tradeoffs between different gap-offshoot configurations at different head subject structures. In the case illustrated on Fig. 4, the subject cluster of third-layer topics presented by checked boxes, can be lifted to two head subjects as in (A) or, just one, the upper, category in (B), with the "cost" of three more gap nodes and one offshoot less. Depending on the relative weighting of gaps, offshoots and multiple head subjects, either lifting can minimize the total misfit.

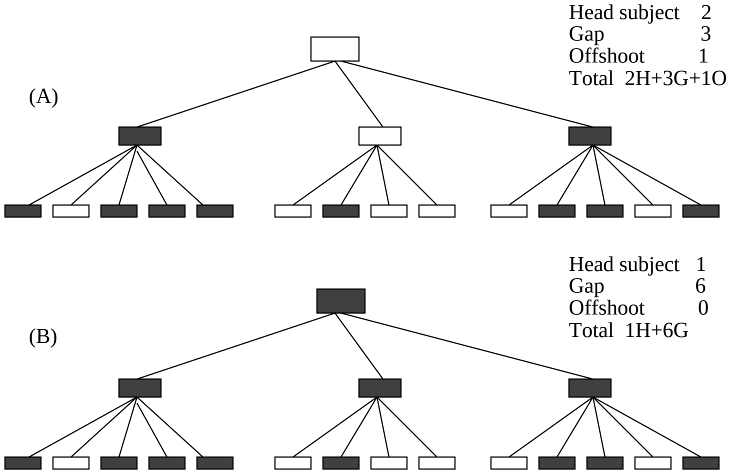


Fig. 4 Tradeoff between different liftings of the same subject cluster: mapping (B) is more parsimonious than (A) if gaps are much cheaper than additional head subjects.

Altogether, the set of topic clusters, their head subjects, offshoots and gaps constitutes what can be referred to as a profile of the organization under consideration. Such a representation can be easily accessed and expressed as an aggregate. It can be further elaborated by highlighting those subjects in which members of the organization have been especially successful (i.e., publication in best journals or award) or distinguished by a special feature (i.e., industrial product or inclusion to a teaching program). Multiple head subjects and offshoots, when persist at subject clusters in different organizations, may show some tendencies in the development of the science, that the classification has not taken into account yet.

A parsimonious lifting of a subject cluster can be achieved by recursively building a parsimonious scenario for each node of the ACM-CCS tree based on parsimonious scenarios for its children. In this, we assume that any head subject is automatically present at each of the nodes it covers, unless they are gaps (as presented on Fig. 4 (B). This assumption allows us to set the algorithm as a recursive procedure.

The procedure determines, at each node of the tree, sets of head gain, gap and offshoot events to iteratively raise them to those of the parents, under each of two different assumptions that specify the situation at the parental node. One assumption is that the head subject has been inherited at the parental node from its own parent, and the second assumption is that it has not been inherited but gained in the node only. In the latter case the parental node is labeled as a head subject. Consider the parent-children system as shown in Fig. 5, with each node assigned with sets of offshoot, gap and head gain events under the above two inheritance of head subject assumptions.

Let us denote the total number of events, to be minimized, under the inheritance and non-inheritance assumptions by e_i and e_n, respectively. A lifting result at a given node is defined by a triplet of sets (H, G, O), representing the tree nodes at which events of head gains, gaps and offshoots, respectively, have occurred in the subtree rooted at the node. We use (Hi, Gi, Oi) and (Hn, Gn, On) to denote lifting results under the inheritance and non-inheritance assumptions, respectively. The algorithm computes parsimonious scenarios for parental nodes according to the topology of the tree, proceeding from the leaves to the root in the manner which is similar to that described in [12].

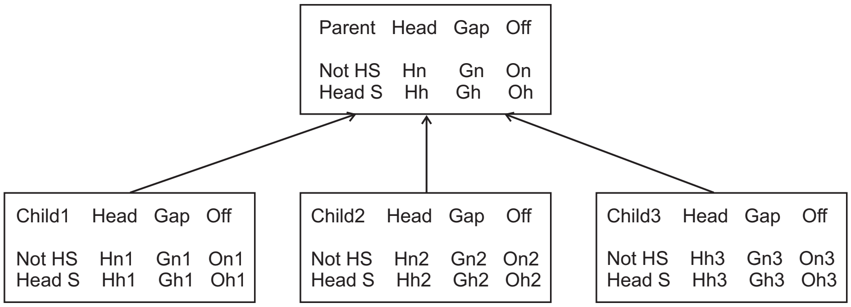


Fig. 5 Events in a parent-children system according to a parsimonious lifting scenario; HS and Head S stand for Head subject.

At a leaf node the six sets Hi, Gi, Oi, Hn, Gn and On are empty, except that Hn =$\{S\}$ if the given leaf belongs to topic cluster S or Gi =$\{S\}$ if not. The algorithm then will compute parsimonious scenarios for parental nodes according to the topology of the tree, proceeding from the leaves to the root. Let us, for the sake of simplicity, consider the case when the penalty for an offshoot is taken to be zero while penalties for the head gain and gap events are specified by arbitrary positive h and g, respectively. Then, in a parsimonious scenario, the total score of events, weighted by h and g, can be derived from those of its children (indicated by subscripts 1, 2 and 3 for the case of three children on Fig. 5) as $e_i = \min(e_{n1}+e_{n2}+e_{n3}+g,\ e_{i1}+e_{i2}+e_{i3})$ or $e_n = \min(e_{i1}+e_{i2}+e_{i3}+h,\ e_{n1}+e_{n2}+e_{n3})$, under the inheritance or non-inheritance assumption, respectively; the proof given in [12] for the binary tree case can be easily extended to an arbitrary rooted tree.

3 An Example of Implementation

Let us illustrate the approach by using the data from a survey conducted at the Department of Computer Science, Faculty of Science & Technology, New University of Lisboa (DI-FCT-UNL). The survey involved 49 members of the academic staff of the department.

For simplicity, we use only data of the second level of ACM-CCS, each coded in the format V.v where V=A,B,...,K, and v =1,..,mK, with mK being the number of second level topics. Each member of the department supplied three ACM subjects most relevant to their current research. Altogether, these comprise 26 of the 59 topics at the second level in ACM-CCS. (Two subjects of the second level, General and Miscellaneous, occurred in every first-level division, are omitted because they do not contribute to the representation.)

With the algorithm ADDI-S sequentially applied to the 26×26 similarity matrix, the following six sequentially extracted clusters have been obtained:

1. Cl1 (contribution 27.08%, intensity 2.17), 4 items: *D.3*, *F.1*, *F.3*, *F.4*;
2. Cl2 (contribution 17.34%, intensity 0.52), 12 items: *C.2*, *D.1*, *D.2*, *D.3*, *D.4*, *F.3*, *F.4*, *H.2*, *H.3*, *H.5*, *I.2*, *I.6*;
3. Cl3 (contribution 5.13%, intensity 1.33), 3 items: *C.1*, *C.2*, *C.3*;
4. Cl4 (contribution 4.42%, intensity 0.36), 9 items: *F.4*, *G.1*, *H.2*, *I.2*, *I.3*, *I.4*, *I.5*, *I.6*, *I.7*;
5. Cl5 (contribution 4.03%, intensity 0.65), 5 items: *E.1*, *F.2*, *H.2*, *H.3*, *H.4*;
6. Cl6 (contribution 4.00%, intensity 0.64), 5 items: *C.4*, *D.1*, *D.2*, *D.4*, *K.6*.

The next 7th cluster's contribution is just 2.5%, on par with the contributions of each of the 26 individual topics, which justifies halting the process at this point.

The six found clusters lifted in the ACM-CCS are presented on Fig. 6 along with the relevant first-level categories.

The lifting results show the following:

- The department covers, with a few gaps and offshoots, six head subjects shown with pentagons filled in by different patterns;
- The most contributing cluster, with the head subject *F. Theory of Computation*, comprises a very tight group of a few second level topics;
- The next contributing cluster has two, not one, head subjects, *D* and *H*, and offshoots to every other head subject in the consideration, which shows that this cluster currently is the structure underlying the unity of the department;
- Moreover, the two head subjects of this cluster come on top of two other clusters, each pertaining to just one of the head subjects, *D. Software* or *H. Information Systems*. This means that the two-headed cluster signifies a new direction in Computer Sciences, that combines *D* and *H* into a single direction, which seems a feature of the current developments in Computer Sciences indeed; this should eventually get reflected in an update of the ACM classification (by raising *D.2 Software Engineering* to the level 1?);
- There are only three offshoots outside the department's head subjects: *E.1 Data Structures* — from *H. Information Systems*, *G.1 Numerical Analysis* — from *I. Computing Methodologies*, and *K.6 Management of Computing and Information Systems* — from *D. Software*. All three seem natural and should be reflected in the list of collateral links between different parts of the classification tree if supported by similar findings at other departments.

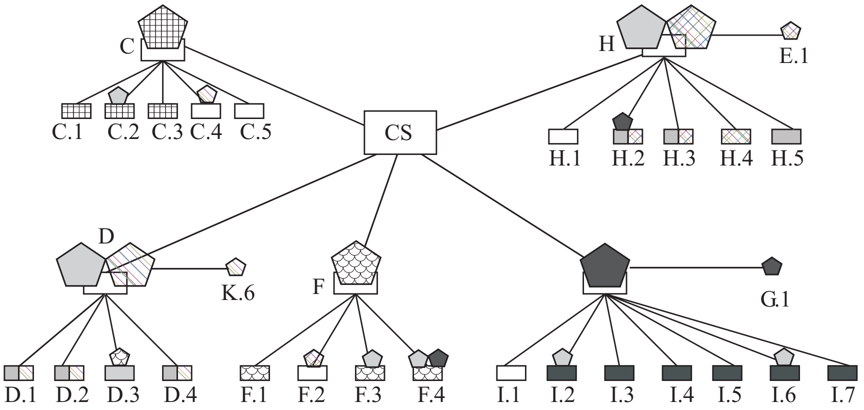


Fig. 6 Six subject clusters in the DI-FCT-UNL represented over the ACM-CCS ontology. Head subjects are shown with differently patterned pentagons. Topic boxes shared by different clusters are split-patterned.

4 Concluding Remarks

We have described a method in the area of knowledge transmutation – for representing aggregated research activities over a concept tree. The method involves two generalization steps: (a) clustering research topics according to their similarities in terms of the efforts by individuals involved, with no relation to the concept tree in question, and (b) generalization of clusters mapped to a concept tree by lifting them to more general categories - this is done over the tree only. Therefore, the generalization steps cover both sides of the representing process.

This work is part of the research project *Computational Ontology Profiling of Scientific Research Organization* (COPSRO), whose main goal is to develop a methodology to represent a Computer Science organization such as a University department over the ACM-CCS classification tree. Such an approach involves the following steps:

1. surveying the members of ACM-CCS topics they are working on; this can be supplemented with indication of the degree of success achieved (publication in a good journal, award, etc.);
2. deriving similarity between ACM-CCS topics resulting from the survey and clustering them;
3. mapping clusters to the ACM-CCS taxonomy and lifting them in a parsimonious way by minimizing the weighted sum of counts of head subjects, gaps and offshoots;
4. aggregating results from different clusters and, potentially, different organizations by means of the taxonomy;
5. interpretation of the results and drawing conclusions.

Current research work includes a survey that is being conducted over several C.S. departments in Universities in Portugal and the U.K., the exploration of fuzzy similarity measures between research topics of the ACM-CCS tree according to the weighted choices of the respondents, and the extension of the additive clustering model to a fuzzy additive version in the framework of the data recovery approach.

In principle, the approach can be extended to other areas of science or engineering, provided that such an area has been systematized into a comprehensive concept tree representation. Potentially, this approach could lead to a useful instrument for visually feasible comprehensive representation of developments in any field of human activities.

Acknowledgements The authors are grateful to DI-FCT-UNL members that participated in the survey. Igor Guerreiro is acknowledged for developing the interface shown in Fig. 1. This work has been supported by the grant PTDC/EIA/69988/2006 from the Portuguese Foundation for Science & Technology.

References

1. *ACM Computing Classification System* (1998) http://www.acm.org/about/class/1998. Cited 9 Sep 2008
2. Feather, M., Menzies, T., Connelly, J.: Matching software practitioner needs to researcher activities. *Proc. of the 10th Asia-Pacific Software Engineering Conference (APSEC'03)*. IEEE, pp. 6 (2003) doi:10.1109/APSEC.2003.1254353
3. Gallo, G., Grigoriadis, M.D., Tarjan, R.E.: A fast parametric maximum flow algorithm and applications. *SIAM Journal on Computing*, **18**(1), pp. 30-55 (1989) doi: 10.1137/0218003
4. Liu, J., Wang, W., Yang, J.: Gene ontology friendly biclustering of expression profiles. *Proc. of the IEEE Computational Systems Bioinformatics Conference*. IEEE, pp. 436-447 (2004) doi: 10.1109/CSB.2004.1332456
5. Michalski, R.S.: Two-tiered concept meaning, inferential matching and conceptual cohesiveness, In: Vosniadou, S., Ortony A. (eds.) *Similarity and Analogical Reasoning*. N.Y., Cambridge University Press (1989)
6. Michalski, R.S.: Inferential learning theory: A conceptual framework for characterizing learning processes. Reports on the Machine Learning and Inference Laboratory, MLI 91–9. George Mason University (1991)
7. Michalski, R.S., Stepp, R.E.: Learning from observation: Conceptual clustering. In: Michalski, R. S., Carbonell, J. G., Mitchell, T. M. (eds.) *Machine Learning: An Artificial Intelligence Approach*. Morgan Kauffmann, San Mateo, CA, pp. 331-363 (1983)
8. Middleton, S., Shadbolt, N., Roure, D.: Ontological user representing in recommender systems. *ACM Trans. on Inform. Systems*, **22**(1), pp. 54-88 (2004) doi: 10.1145/963770.963773
9. Miralaei, S., Ghorbani, A.: Category-based similarity algorithm for semantic similarity in multi-agent information sharing systems. *IEEE/WIC/ACM Int. Conf. on Intelligent Agent Technology*, pp. 242-245 (2005) doi: 10.1109/IAT.2005.50
10. Mirkin, B.: Additive clustering and qualitative factor analysis methods for similarity matrices. *Journal of Classification*, **4**(1), pp. 7-31 (1987) doi:10.1007/BF01890073
11. Mirkin, B.: *Clustering for Data Mining: A Data Recovery Approach*. Chapman & Hall /CRC Press, 276 p. (2005)
12. Mirkin, B., Fenner, T., Galperin, M., Koonin, E.: Algorithms for computing parsimonious evolutionary scenarios for genome evolution, the last universal common ancestor and dominance

of horizontal gene transfer in the evolution of prokaryotes. *BMC Evolutionary Biology*, **3**:2 (2003) doi:10.1186/1471-2148-3-2

13. Nascimento, S., Mirkin, B., Moura-Pires, F.: Modeling proportional membership in fuzzy clustering. *IEEE Transactions on Fuzzy Systems*, **11**(2), pp. 173-186 (2003) doi: 10.1109/TFUZZ.2003.809889
14. *RAE2008: Research Assessment Exercise* (2008) http://www.rae.ac.uk/. Cited 9 Sep 2008.
15. Shi, J., Malik, J.: Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(8), pp. 888-905 (2000)
16. Spence, R.: *Information Visualization*. Addison-Wesley, ACM Press, 206 p. (2000)
17. Stepp, R., Michalski, R.S.: Conceptual clustering of structured objects: A goal-oriented approach. *Artificial Intelligence*, **28**(1), pp. 43-69 (1986) doi: 10.1016/0004-3702(86)90030-5
18. Thorne, C., Zhu, J., Uren, V.: Extracting domain ontologies with CORDER. *Tech. Report kmi-05-14*. Open University, pp. 1-15 (2005)
19. *The United Nations Millennium Project Task Force*. http://www.cid.harward.edu/cidtech. Cited 1 Sep 2006.
20. Weiss, S.M., Indurkhya, N., Zhang, T., Damerau, F.J.: *Text Mining: Predictive Methods for Analyzing Unstructured Information*. Springer Verlag, 237 p. (2005)
21. Yang, L., Ball, M., Bhavsar, V., Boley, H.: Weighted partonomy-taxonomy trees with local similarity measures for semantic buyer-seller match-making. *Journal of Business and Technology*. Atlantic Academic Press, **1**(1), pp. 42-52 (2005)