Method for Intelligent Representation of Research Activities of an Organization over a Taxonomy of its Field

Boris Mirkin, Susana Nascimento and Luís Moniz Pereira

Abstract We describe a novel method for the analysis of research activities of an organization by mapping that to a taxonomy tree of the field. The method constructs fuzzy membership profiles of the organization members or teams in terms of the taxonomy's leaves (research topics), and then it generalizes them in two steps. These steps are: (i) fuzzy clustering research topics according to their thematic similarities in the department, ignoring the topology of the taxonomy, and (ii) optimally lifting clusters mapped to the taxonomy tree to higher ranked categories by ignoring "small" discrepancies. We illustrate the method by applying it to data collected by using an in-house e-survey tool from a university department and from a university research center. The method can be considered for knowledge generalization over any taxonomy tree.

Boris Mirkin
Department of Computer Science, Birkbeck University of London, London, UK, and School of Applied Mathematics and Informatics, Higher School of Economics, Moscow, RF, e-mail: mirkin@dcs.bbk.ac.uk

Susana Nascimento
Department of Computer Science and Centre for Artificial Intelligence (CENTRIA), Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, 2829-516 Caparica, Portugal, e-mail: snt@di.fct.unl.pt

Luís Moniz Pereira
Department of Computer Science and Centre for Artificial Intelligence (CENTRIA), Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, 2829-516 Caparica, Portugal, e-mail: lmp@di.fct.unl.pt

1 Introduction

1.1 Motivation

Our subject should be counted as what can be referred to as organizational knowledge management. We represent activity of an organization in a novel way by mapping it to an ontology of the field. Our method involves three stages: (i) data integration, (ii) ontology usage, and (iii) activity visualization.

To give an intuitive idea of the method, let us first consider three similar stages of data representation for operative control, such as that by a company delivering electricity to homes in a town zone. Fig. 1, taken from [2], represents an energy network over a map of the corresponding district on which the topography and the network data are integrated in such a way that gives the company "an unprecedented ability" to control the flow of energy by following all the maintenance and repair issues on-line in a real time framework.

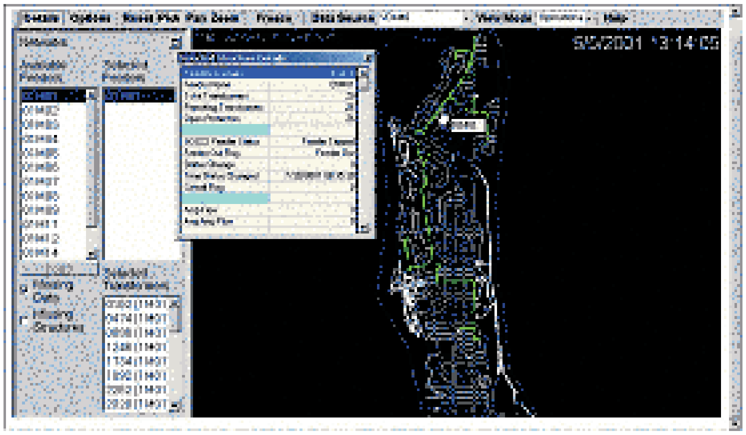


Fig. 1: Energy network of Con Edison Company on Manhattan New-York USA visualized by Advanced Visual Systems [2].

What we are concerned with is whether a similar mapping is possible for a long-term analysis of an organization whose activity is much less tangible, such as a university research department. There are three major ingredients that allow for a successful representation of the energy network:

(1) map of the district,
(2) the energy network units, and
(3) mapping (2) at (1).

Moreover, one could imagine an extension of this mapping to other infrastructure items, such as the water supply, sewage type and transports, so that the map could be used for more long-term city planning tasks such as development of leisure or residential areas and the like. This allows us to take, for a research department, the following analogues to the elements of the mapping in Fig. 1:

(1') an ontology of Computer Science (CS),
(2') the set of CS research subjects being developed by members of the department, and
(3') representation of the research on the ontology.

Why would one want to do that? There can be various management goals such as, for example:

- Positioning of the research organization within the taxonomy;
- Analyzing and planning the structure of research conducted in the organization;
- Finding nodes of excellence, nodes of failure and nodes needing improvement for the organization;
- Discovering research elements that poorly match the structure of the taxonomy;
- Planning of research and investment;
- Integrating data of different research organizations in a region for the purposes of regional planning and management.

Before moving further, let us take a look at the structure of Classification of Computer Subjects developed by the Association for Computing Machinery (ACM-CCS) [1] which is the heart of the Computer Science ontology. Its highest layer is presented in Fig.2 that shows the whole of Computer Science as divided into eleven first-layer subjects:

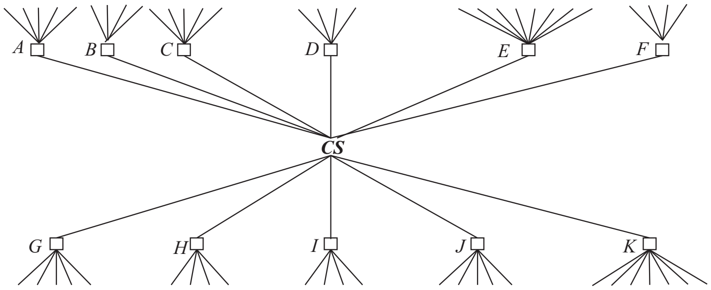


Fig. 2: The higher level of ACM-CCS Taxonomy.

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

Each of the eleven subjects is further subdivided into subjects of the second layer. For example, the subject of our current interest, *'I. Computing Methodologies'*, consists of seven specific subjects plus two of general interest, *'0. GENERAL'* and *'m. MISCELLANEOUS'* these two are present in every division of ACM-CCS:

I. Computing Methodologies

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)
I.m MISCELLANEOUS

Many of these are further subdivided into subjects of the third-layer such as, for instance,

I.5 PATTERN RECOGNITION

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

There can be also units of the fourth layer that are not indexed and used mainly as indications of the contents of the third layer subjects, such as those two shown for topic *'I.5.3 Clustering'*. One can also see a number of collateral links between topics both on the second and the third layers - they are in the parentheses at the end of some topics such as *G.3*, referred to at *I.6*.

At first glance, mapping of subjects under development in a department to the taxonomy is a rather straightforward exercise. For example, a survey found that 25 of the total of 81 meaningful subjects of the second layer are being developed in a research department[1]. After these 25 subjects are mapped to the taxonomy, one can

[1] Survey conducted in the CS department of Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa (DI-FCT-UNL) in 2007 [33].

see them visualized by black boxed nodes in Fig. 3. This portrayal is not entirely unhelpful - the visualization does provide a useful information of the coverage of the taxonomy subjects by the research. Yet this representation gives no hints of the structure of the research: the subjects are presented with no indication of relation between them according to the working of the department. The taxonomy reflects only those relations between the subjects that have been specified in it according to the working of the entire community of computer scientists. The structure of research projects in a department may impose a different taxonomy of the subjects. This "local" taxonomy would reflect the relations between subjects according to research projects worked on in the organization: the larger the number and weight of research projects that involve two taxonomy subjects, the greater association between the subjects in the department. The local taxonomy may not necessarily follow the structure of the global taxonomy, and it would be of interest to map the local taxonomy to the global one. Although we are going to explore this problem in the future, this paper concerns a less challenging undertaking. Instead of presenting the "local" structure of research as a hierarchical taxonomic structure, we present it as a "flat" set of not necessarily disjoint clusters of ACM-CCS subjects in such a way that the clusters reflect the "local" associations between the subjects - the greater the weight and number of the projects at which two subjects are involved the greater the association between those subjects and greater the chance that the two subjects belong to the same cluster.

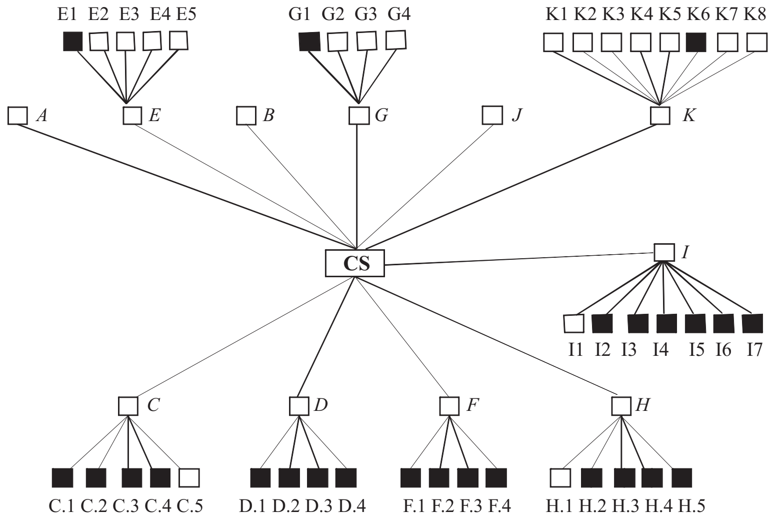


Fig. 3: Twenty five ACM-CCS subjects of a University department mapped to the ACM-CCS taxonomy.

Returning to the 25 ACM-CCS subjects, we have found that they can be reasonably divided into 6 clusters which are mapped to the ACM-CCS taxonomy separately to produce the following portrayal (Fig. 4) [33]. The mapping involves the concept of "head subject" that can be defined as the highest rank node(s) covering, in general, the cluster. Since the coverage is not necessarily exact (see Sect. 4 for definitions and method) two more types of elements emerge. These are: "gap", that is a node covered by a cluster head subject but not belonging to the cluster, and "offshoot", that is a node belonging to the cluster but not covered by its head subject node(s). These are illustrated in Fig. 4 with different graphic elements. Among interesting features of the research conducted in the department, three clusters have been found to relate to head subjects *'D.Software'* and *'H. Information Systems'* so that two of the clusters have only one of them as their respective head subject, whereas the third one received both of the nodes as its head subjects, thus suggesting an integrating development that has been underway in the department. This integrating development can be attributed to establishment in recent years of *'D.2 Software Engineering'* as a major subject in Computer Science which is yet to be reflected by raising the node in the ACM-CCS taxonomy.

Our method for finding clusters of research subjects according to the workings of the organization involves the following steps:

1. defining research units representing the activities;
2. determining research profiles of the units, that is, crisp or fuzzy sets of the taxonomy subjects to represent every unit;
3. integrating the profiles in a matrix of similarity scores between the taxonomy subjects which are worked on in the organization; and
4. finding clusters of taxonomy subjects representing the similarity matrix.

The cluster finding completes the first stage of the approach, (i) the organization data integration. The second stage, (ii) the ontology usage, works as follows: a thematic cluster found at stage (i) would be considered as a query to the ontology requesting to find a node or two in the taxonomy, the head subject(s), as high as possible, to cover all the nodes in the query in such a way that the gaps and offshoots emerging at the high rank head subject would be not too extensive, or too expensive in the penalty function defined for any set of head subjects. The third stage, (iii) the activity visualization, presents the results over the ontology in the manner of Fig. 4. The rules for interpretation of the results are yet to be produced, though some simple observations like those above can be recommended already.

This three-stage approach has been sketched out in our previous paper [33]. The current paper outlines the current state of the approach. Specifically, the following novel features are described here. First, for stage (i), instead of a crisp clustering approach, we developed a genuine fuzzy clustering method based on an additive model of the subject-to-subject similarity matrix and involving the spectral clustering approach [35]. Second, for stage (ii), instead of some informal considerations described in [33], we developed an optimal lifting method based on minimization of a penalty function embracing all the elements of the lift, the head subjects, the gaps and the offshoots [34]. Using the software developed, the visualization stage (iii), now can be conducted automatically, not manually, which raises some new possibilities in manipulating the relative weights of lifting elements.

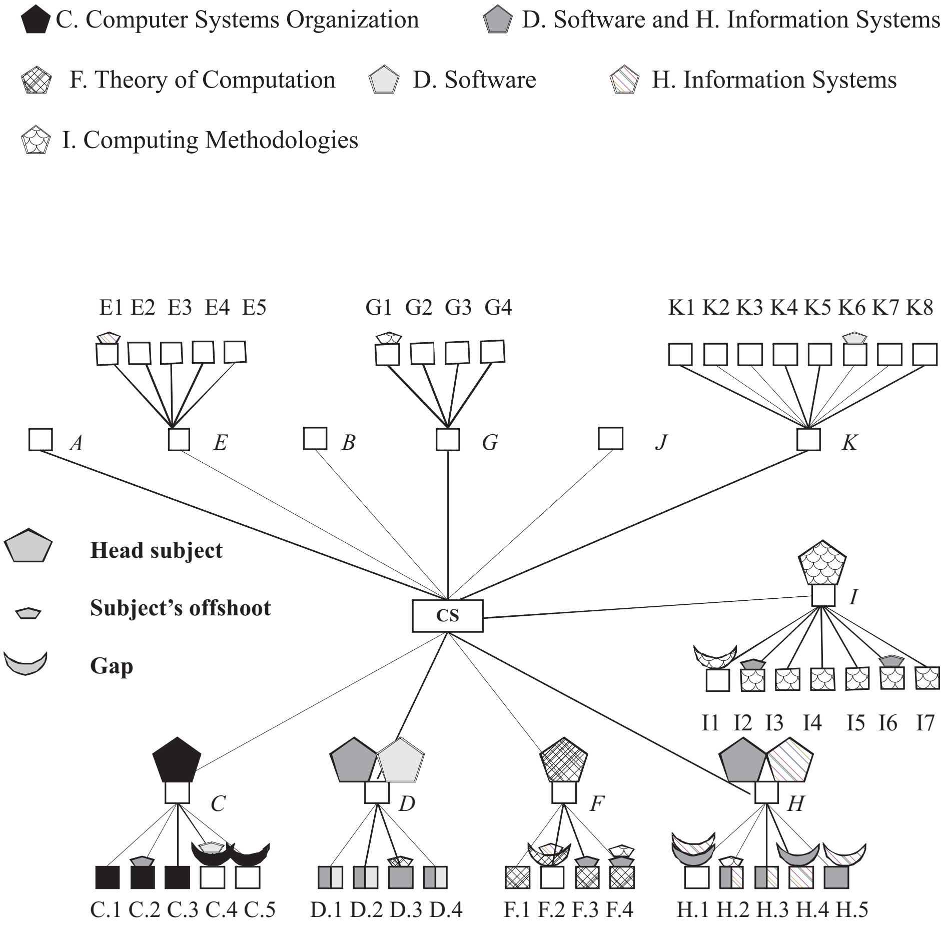


Fig. 4: Six clusters visualized on the ACM-CCS taxonomy by larger pentagrams of "head subjects" differently colored. Now we can see more structure in the organization's research as described in the text.

1.2 Background

Since the stages of our approach involve relatively different techniques, the background will be described in the following subsections along the separate lines of development in the literature: Fuzzy clustering background; Ontology usage background; Activity visualization background.

1.2.1 Fuzzy Clustering Background

The major fuzzy clustering algorithms, such as *c*-means, an extension of the popular *k*-means approach, work on data in the entity-to-feature format [5]. Yet the result of our first stage is a square subject-to-subject similarity matrix. Thus, we are concerned here mainly with the so-called relational fuzzy clustering, an activity of deriving fuzzy clusters from a relation, that is, a matrix of a similarity or dissimilarity index. The published work on this can be divided in two major streams: one utilizing the fuzzy logics operations such as minimum or plus but no operation of division, and the other involving all the numeric operations, including division. The former is rather thin and less developed (see, for instance, [52] and [20]). We adhere to the latter stream, which can be traced to papers [40] and [49] that utilized, essentially, the sum $\sum_{k=1}^{K}\sum_{t,t'} u_{tk}^2 u_{t'k}^2 d(t,t')$ as the criterion to minimize over unknown membership vectors $\mathbf{u}_k=(u_{tk}), k=1,...,K$ where t,t denote ontology subjects. A similar criterion, proven to be equivalent to the criterion of popular fuzzy c-means method [5], was utilized by Hathaway, Davenport and Bezdek [21] to derive their RFCM algorithm, that works in two-phase iterations similar to c-means, including a relational analogue to the concept of cluster centroid. Specifying the so-called "fuzzifying" constant at the level of 2, the RFCM criterion is the sum over $k=1,...,K$ of items $\sum_{t,t'} u_{tk}^2 u_{t'k}^2 d(t,t')/\sum_t u_{tk}^2$ where $d(t,t')$ is the squared Euclidean distance at different d RFCM may lead to negative memberships. But even in this format, RFCM appears to be superior to Windham's assignment-prototype algorithm [4]. Later this restriction was relaxed, initially, by modifying RFCM into NERFCM algorithm to include the addition of a positive number to all off diagonal distances [22] and, more recently, by directly imposing the non-negativity constraint for memberships [9]. The latter paper also extended the concept of fuzzy clustering to include the so-called "noise" cluster to hold the bulk of membership for entities that are far away from the K clusters being built. Brouwer [6] makes use of a two-stage procedure in which the first stage supplies the entities with a few distance-approximating features so that the second stage utilizes a conventional algorithm such as fuzzy c-means for building fuzzy clusters in the feature space. This approach proved superior to the others in experiments reported in [6].

Yet there are a number of issues related to these approaches that are not quite satisfactory:

1. The cluster memberships form what is called a fuzzy partition so that each entity shares its full membership among the clusters. This does not allow an entity to belong to no cluster or partly belong to the clusters.
2. The clusters do not feed back to the similarity data - they do not represent the data as a function of clusters, which would be a desirable option when modelling research activities. A nice additive clustering model of similarity data has been introduced, in English, by Shepard and Arabie [44] for crisp clusters. The model proposes that each cluster is characterized by a positive constant, intensity, that adds up to the similarity between entities in the cluster. Paper [31] referred to earlier publications, in Russian, and proposed an iterative crisp cluster extraction framework in that setting. However, the additive clustering model had not been

extended to relational fuzzy clustering until a simplified version of the model, involving constant, not cluster-specific, intensity weights, was considered in [41] citing no specific applications and using Newton's descent method for fitting the model. This method involves many initialization parameters that need to be prespecified, which is not what an innocent user would be willing to do. In this paper, we would like to use a proper extension of the additive model to fuzzy clustering such as introduced by Mirkin and Nascimento [35].

3. There is no explicit instruction for the choice of number of clusters in the fuzzy clustering models. Typically, the number of clusters is considered to be determined based on some post-processing considerations related to stability of the clusters regarding some random changes in either the initialization or the data themselves [29]. In this regard, the sequential manner of the method proposed in [35] can be considered an advantage because it allows choosing the option of stopping computations after any number of clusters.

1.2.2 Ontology Usage Background

The concept of ontology as a computationally feasible environment for knowledge maintenance has recently emerged to comprise, first of all, the set of concepts and relations between them pertaining to the knowledge of the domain. The initial attempts have been concentrated at automatically generating ontologies from web and other document resources; a review of the efforts to about 2000 can be found in [10]. Meanwhile, it became clear that currently a relevant ontology can be produced only manually, and big ontologies are being built, first of all, in medicine (see SNOMED [46]) and bioinformatics (GO [18]). Currently, most research efforts by computer scientists, beyond developing platforms and languages for ontology representation (see, for example, developments of OWL language (e.g. [39]), are concentrated on computational methods for (a) integrating ontologies and (b) using them for various purposes.

The issue of (a) integration of different ontologies requires developing a common ground for representing different elements of ontologies as well as methods for mapping the elements of the same type between ontologies. Examples of the former can be found in [48, 15]. Examples of the latter can be found in [7, 19]. Both of these are in rather initial states of development, which is supported by the findings of [19]: this shows that a simple text matching method outperforms those involving the ontologies.

The issue of (b) usage of ontologies is of especial importance to us because our paper should be counted in this category. Most efforts here are devoted to building rules for ontological reasoning and querying utilizing the inheritance relation supplied by the ontology's taxonomy in the presence of different data models ([8, 3, 47]). These do not attempt approximate representations but just utilize additional possibilities supplied by the ontology relations. Another type of ontology usage is in using its taxonomy nodes for interpretation of data mining results such as association rules [27] and clusters [11, 14]. Our approach naturally falls within this category. What we want is to generalize the query set within the taxonomy in a 'soft' manner by allowing some "non-costly" discrepancies between the set and the subtree rooted at the generalized concept, which differs from the other work on queries to ontologies that strictly conform to the crisp meanings [8, 3, 47].

1.2.3 Visualization Background

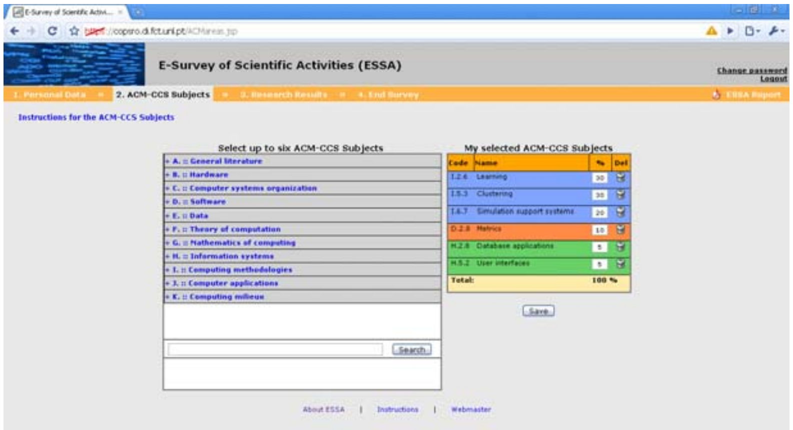
The subject of visualization attracts increasing attention of computer scientists. In this regard, usually the visualization of activities does not much differ from visualization of any other concept, of which many papers and websites inform (see, for a recent reference on visualization [28]). Some aspects of activities have been covered such as, for instance, web related activities [12], and, more recently, modelling activity in general is being considered as well [42, 17]. Our case, as illustrated in Fig.4, is very much clear cut: the organization's activity is represented by a set of clusters that are supplied to a taxonomy as query sets to be lifted to head subjects expressing the general tendencies of the activities, not without some gap or offshoot pitfalls. This visualization attends here to not just cognitive abilities of the user but to more tangible issues related to the analysis of matches and mismatches between the query and taxonomy, that could be interpreted, in respect, as points of strength or weakness, and give rise to questions of their meaning in the context of the taxonomy to be used in planning of the organization changes and investment policies. Potentially, after integration of activities of a number of organizations in the taxonomy, one could use the discrepancies to feedback to the taxonomy itself, as the points requiring a most urgent taxonomy updating.

2 Taxonomy-based Profiles

In the case of investigation of activities of a university department or research center, a research team's profile can be defined as a fuzzy membership function on the set of leaf-nodes of the taxonomy under consideration so that the memberships reflect the extent of the team's effort put into corresponding research topics.

2.1 E-Screen Survey Tool

Fuzzy membership profiles are derived from either automatic analysis of documents posted on the 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, for example, in non-English speaking countries with relatively underdeveloped internet infrastructures for the maintenance of research results. We have developed an interactive survey tool that provides two types of functionality: i) collection of data about ACM-CCS based research profiles of individual members; ii) statistical analysis and visualization of the data and results of the survey on the level of a department. 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 for, say, the past four years. This describes the respondent's activity fuzzy membership profile. Fig. 5 shows a screenshot of the baseline interface for a respondent who has chosen six ACM-CCS topics during his/her survey session.


E-Survey of Scientific Activi...
://copsro.di.fct.unl.pt/ACMareas.jsp
Change password
Logout
1. Personal Data
2. ACM-CCS Subjects
3. Research Results
4. End Survey
ESSA Report
Instructions for the ACM-CCS Subjects
Search
Save
About ESSA | Instructions | Webmaster

E-Survey of Scientific Activities (ESSA)

Select up to six ACM-CCS 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

My selected 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. 5: Screenshot of the interface survey tool for selection of ACM-CCS topics.

The set of profiles supplied by respondents forms an $N \times M$ matrix F where N is the number of ACM-CCS topics involved in the profiles and M the number of respondents. Each column of F is a fuzzy membership function, rather sharply delineated because only six topics maximum may have acknowledged membership in each of the columns.

3 Representing Research Organization by Fuzzy Clusters of ACM-CCS Topics

3.1 Deriving Similarity between ACM-CCS Research Topics

We represent a research organization by clusters of ACM-CCS topics to reflect thematic communalities between activities of members or teams working on these topics. The clusters are found by analyzing similarities between topics according to their appearances in the profiles. The more profiles contain a pair of topics i and j and the greater the memberships of these topics, the greater is the similarity score for the pair.

Consider a set of V individuals ($v = 1, 2, \cdots, V$), engaged in research over some topics $t \in T$ where T is a pre-specified set of scientific subjects. The level of research effort by individual v in developing topic t is evaluated by the membership f_{tv} in profile f_v ($v = 1, 2, \cdots, V$).

Then the similarity $w_{tt'}$ between topics t and t' is defined as

$$w_{tt'} = \sum_{v=1}^{V} \frac{n_v}{n_{max}} f_{tv} f_{t'v}, \tag{1}$$

where the ratios of the number of topics chosen by individual v, n_v, and n_{max}, the maximum n_v over all $v = 1, 2, \cdots, V$, are introduced to balance the scores of individuals bearing different numbers of topics.

To make the cluster structure in the similarity matrix sharper, we apply the spectral clustering approach to pre-process the similarity matrix W using the so-called Laplacian transformation [26]. First, an $N \times N$ diagonal matrix D is defined, with (t,t) entry equal to $d_t = \sum_{t' \in T} w_{tt'}$, the sum of t's row of W. Then unnormalized Laplacian and normalized Laplacian are defined by equations $L = D - W$ and $L_n = D^{-1/2} L D^{-1/2}$, respectively. Both matrices are semipositive definite and have zero as the minimum eigenvalue. The minimum non-zero eigenvalues and corresponding eigenvectors of the Laplacian matrices are utilized then as relaxations of combinatorial partition problems [45, 26]. Of comparative properties of these two normalizations, the normalized Laplacian, in general, is considered superior [26]. Since the additive clustering approach described in the next section relies on maximum rather than minimum eigenvalues, we use the Laplacian PseudoINverse transformation, Lapin for short, defined by

$$L_n^{+}(W) = \tilde{Z} \tilde{\Lambda}^{-1} \tilde{Z}'$$

where $\tilde{\Lambda}$ and $\tilde{Z}$ are defined by the spectral decomposition $L_n = Z \Lambda Z'$ of matrix $L_n = D^{-1/2}(D - W) D^{-1/2}$. To specify these matrices, first, set T' of indices of elements corresponding to non-zero elements of Λ is determined, after which the matrices are taken as $\tilde{\Lambda} = \Lambda(T', T')$ and $\tilde{Z} = Z(:, T')$. The choice of the Lapin transformation can be explained by the fact that it leaves the eigenvectors of L_n unchanged while inverting the non-zero eigenvalues $\lambda \neq 0$ to those $1/\lambda$ of L_n^{+}. Then the maximum eigenvalue of L_n^{+} is the inverse of the minimum non-zero eigenvalue λ_1 of L_n, corresponding to the same eigenvector.

3.2 Fuzzy Additive-Spectral Clustering

In spite of the fact that many fuzzy clustering algorithms have been developed already [5, 25], most of them are ad hoc and, moreover, they all involve manually specified parameters such as the number of clusters or threshold of similarity without providing any guidance for choosing them. We apply a model-based approach of additive clustering, combined with the spectral clustering approach, to develop a fuzzy clustering method that is both adequate and supplied with model-based parameters helping to choose the right number of clusters.

Thematic similarities $a_{tt'}$ between topics are but manifested expressions of some hidden patterns within the organization which can be represented by fuzzy clusters in exactly the same manner as the manifested scores in the definition of the similarity $w_{tt'}$ (1). We propose to formalize a thematic fuzzy cluster as represented by two items: (i) a membership vector $\mathbf{u} = (u_t)$, $t \in T$, such that $0 \leq u_t \leq 1$ for all $t \in T$, and (ii) an intensity $\mu > 0$ that expresses the extent of significance of the pattern corresponding to the cluster, within the organization under consideration. With the introduction of the intensity, applied as a scaling factor to $\mathbf{u}$, it is the product $\mu\mathbf{u}$ that is a solution rather than its individual co-factors. Given a value of the product μu_t, it is impossible to tell which part of it is μ and which u_t. To resolve this, we follow a conventional scheme: let us constrain the scale of the membership vector $\mathbf{u}$ on a constant level, for example, by a condition such as $\sum_t u_t = 1$ or $\sum_t u_t^2 = 1$, then the remaining factor will define the value of μ. The latter normalization better suits the criterion implied by our fuzzy clustering method and, thus, is accepted further on.

Our additive fuzzy clustering model follows that of [44, 31, 41] and involves K fuzzy clusters that reproduce the pseudo-inverted Laplacian similarities $a_{tt'}$ up to additive errors according to the following equations [35]:

$$a_{tt'} = \sum_{k=1}^{K} \mu_k^2 u_{kt} u_{kt'} + e_{tt'}, \tag{2}$$

where $\mathbf{u}_k = (u_{kt})$ is the membership vector of cluster k, and μ_k its intensity.

The item $\mu_k^2 u_{kt} u_{kt'}$ is the product of $\mu_k u_{kt}$ and $\mu_k u_{kt'}$ expressing participation of t and t', respectively, in cluster k. This value adds up to the others to form the similarity $a_{tt'}$ between topics t and t'. The value μ_k^2 summarizes the contribution of the intensity and will be referred to as the cluster's weight.

To fit the model in (2), we apply the least-squares approach, thus minimizing the sum of all $e_{tt'}^2$. Since A is definite semi-positive, its first K eigenvalues and corresponding eigenvectors form a solution to this if no constraints on vectors $\mathbf{u}_k$ are imposed. Additionally, we apply the one-by-one principal component analysis strategy for finding one cluster at a time this makes the computation feasible and is crucial for determining the number of clusters. Specifically, at each step, we consider the problem of minimization of a reduced to one fuzzy cluster least-squares criterion

$$E = \sum_{t,t' \in T} (b_{tt'} - \xi u_t u_{t'})^2 \tag{3}$$

with respect to unknown positive ξ weight (so that the intensity μ is the square root of ξ) and fuzzy membership vector $\mathbf{u} = (u_t)$, given similarity matrix $B = (b_{tt'})$.

At the first step, B is taken to be equal to A. Each found cluster changes B by subtracting the contribution of the found cluster (which is additive according to model (2)), so that the residual similarity matrix for obtaining the next cluster is equal to $B - \mu^2\mathbf{u}\mathbf{u}'$ where μ and $\mathbf{u}$ are the intensity and membership vector of the found cluster. In this way, A indeed is additively decomposed according to formula (2) and the number of clusters K can be determined in the process.

Let us specify an arbitrary membership vector $\mathbf{u}$ and find the value of ξ minimizing criterion (3) at this $\mathbf{u}$ by using the first-order optimality condition:

$$\xi = \frac{\sum_{t,t'\in T} b_{tt'} u_t u_{t'}}{\sum_{t\in T} u_t^2 \sum_{t'\in T} u_{t'}^2},$$

so that the optimal ξ is

$$\xi = \frac{\mathbf{u}'B\mathbf{u}}{(\mathbf{u}'\mathbf{u})^2} \tag{4}$$

which is obviously non-negative if B is semi-positive definite.

By putting this ξ in equation (3), we arrive at

$$E = \sum_{t,t'\in T} b_{tt'}^2 - \xi^2 \sum_{t\in T} u_t^2 \sum_{t'\in T} u_{t'}^2 = S(B) - \xi^2 \left(\mathbf{u}'\mathbf{u}\right)^2,$$

where $S(B) = \sum_{t,t'\in T} b_{tt'}^2$ is the similarity data scatter.

Let us denote the last item by

$$G(\mathbf{u}) = \xi^2 \left(\mathbf{u}'\mathbf{u}\right)^2 = \left(\frac{\mathbf{u}'B\mathbf{u}}{\mathbf{u}'\mathbf{u}}\right)^2, \tag{5}$$

so that the similarity data scatter is the sum:

$$S(B) = G(\mathbf{u}) + E \tag{6}$$

of two parts, $G(\mathbf{u})$, which is explained by cluster $(\mu, \mathbf{u})$, and E, which remains unexplained.

An optimal cluster, according to (6), is to maximize the explained part $G(\mathbf{u})$ in (5) or its square root

$$g(\mathbf{u}) = \xi\mathbf{u}'\mathbf{u} = \frac{\mathbf{u}'B\mathbf{u}}{\mathbf{u}'\mathbf{u}}, \tag{7}$$

which is the celebrated Rayleigh quotient: its maximum value is the maximum eigenvalue of matrix B, which is reached at its corresponding eigenvector, in the unconstrained problem.

This shows that the spectral clustering approach is appropriate for our problem. According to this approach, one should find the maximum eigenvalue λ and corresponding normed eigenvector z for B, $[\lambda, z] = \Lambda(B)$, and take its projection to the set of admissible fuzzy membership vectors.

Our clustering approach involves a number of model-based criteria for halting the process of sequential extraction of fuzzy clusters. The process stops if either is true:

1. The optimal value of ξ (4) for the spectral fuzzy cluster becomes negative.
2. The contribution of a single extracted cluster to the data scatter becomes too low, less than a pre-specified $\tau > 0$ value.
3. The residual data scatter becomes smaller than a pre-specified ε value, say less than 5% of the original similarity data scatter.

The described one-by-one Fuzzy ADDItive-Spectral thematic cluster extraction algorithm is referred to as FADDI-S in [35]. It combines three different approaches: additive clustering [44, 31, 41], spectral clustering [45, 26, 55], and relational fuzzy clustering [5, 6] and adds an edge to each. In the context of additive clustering, fuzzy approaches were considered only by [41], yet in a very restricted setting: (a) the clusters intensities were assumed constant there, (b) the number of clusters was pre-specified, and (c) the fitting method was very local and computationally intensive - these all restrictions are overcome in FADDI-S. The spectral clustering approach is overtly heuristic, whereas FADDI-S is model-based. The criteria used in relational fuzzy clustering are ad hoc whereas that of FADDI-S is model-based, and, moreover, its combined belongingness function values $\mu\mathbf{u}$ are not constrained by unity as is the case in relational clustering, but rather follow the scales of the similarity relation under investigation, which is in line with the original approach by L. Zadeh [54].

3.3 Experimental Verification of FADDI-S

We describe here results of two of the experiments reported in [35].

3.3.1 Fuzzy Clustering Affinity Data

The affinity data is a relational similarity data obtained from a feature based dataset using a semi-positive definite kernel, usually the Gaussian one. Specifically, given an $N \times V$ matrix $Y = (y_{tv})$, $t \in T$ and $v = 1, 2, ..., V$, non-diagonal elements of the similarity matrix W are defined by equation

$$w_{tt'} = exp(-\frac{\sum_{v=1}^{V}(y_{tv} - y_{t'v})^2}{2\sigma^2}),$$

with the diagonal elements made equal to zero, starting from founding papers [45, 38]. The value $ss = 2\sigma^2$ is a user-defined parameter, that is pre-specified to make the resulting similarities $w_{tt'}$ spread over interval [0,1].

To compare our approach with other methods for fuzzy clustering of affinity data, we pick up an example from a recent paper by Brouwer [6]. This example concerns a two-dimensional data set, that we refer to as Bivariate4, comprising four clusters generated from bivariate spherical normal distributions with the same standard deviation 950 at centers (1000, 1000), (1000,4000), (4000, 1000), and (4000, 4000), respectively. The data forms a cloud presented in Fig. 6.

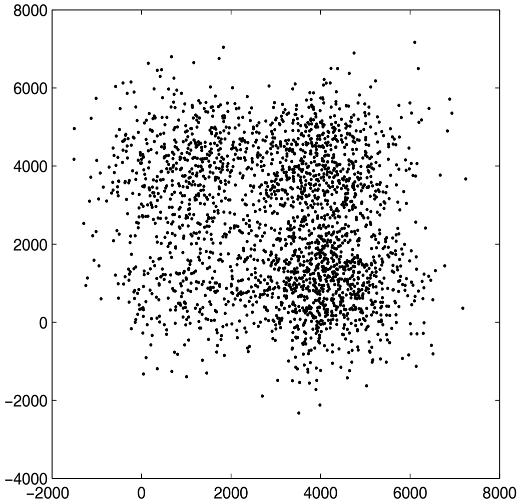


Fig. 6: Bivariate4: the data of four bivariate clusters generated from Gaussian distributions according to [6].

This data was analyzed in [6] by using the matrix D of Euclidean distances between the generated points. Five different fuzzy clustering methods have been compared, three of them relational, by Roubens [40], Windham [49] and NERFCFM [22], and two of fuzzy c-means (FCM) with different preliminary pre-processing options of the similarity data into the entity-to-feature format, FastMap and SMACOF [6]. Of these five different fuzzy clustering methods, by far the best results have been obtained with method FCM applied to a five-feature set extracted from D with FastMap method [6]. The Adjusted Rand index [24] of the correspondence between the generated clusters and those found with the FCM over FastMap method is equal on average, of 10 trials, 0.67 (no standard deviation is reported in [6]).

To compare FADDI-S with these, we apply Gaussian kernel to the data generated according to the Bivariate4 scheme and pre-processed by the z-score standardization so that similarities, after z-scoring, are defined as $a_{ij} = exp(-d^2(y_i, y_j)/0.5)$ where d is Euclidean distance. This matrix then is Lapin transformed to the matrix W to which FADDI-S is applied.

To be able to perform the computation using a PC MatLab, we reduce the respective sizes of the clusters, 500, 1000, 2000, and 1500 totaling to 5000 entities altogether in [6], tenfold to 50, 100, 200 and 150 totaling to 500 entities. The issue is of doing a full spectral analysis of the square similarity matrices of the entity set sizes, which we failed to do with our PC MatLab versions at a 5000 strong dataset. We also experimented with fivefold and twofold size reductions. This should not much change the results because of the properties of smoothness of the spectral decompositions [23].

Indeed, one may look at a 5000 strong random sample as a combination of two 2500 strong random samples from the same population. Consider a randomly generated $N \times 2$ data matrix X of N bivariate rows, thus leading to Lapin transformed $N \times N$ similarity matrix W. If one doubles the data matrix by replicating X as $XX = [X;X]$, in MatLab notation, which is just a $2N \times 2$ data matrix consisting of a replica of X under X, then its Lapin transformed similarity matrix will be obviously equal to

$$WW = \begin{bmatrix} W & W \\ W & W \end{bmatrix}$$

whose eigenvectors are just doubles (z,z) of eigenvectors z of W. If the second part of the double data matrix XX slightly differs from X, due to sampling errors, then the corresponding parts of the doubled similarity matrix and eigenvectors also will slightly differ from those of WW and (z,z). Therefore, the property of stability of spectral clustering results [23] will hold for thus changed parts. This argument equally applies to the case when the original sample is supplemented by four or nine samples from the same population.

In our computations, five consecutive FADDI-S clusters have been extracted for each of randomly generated ten Bivariate4 datasets. The very first cluster has been discarded as reflecting just the general connectivity information, and the remaining four were defuzzified into partitions so that every entity is assigned to its maximum membership class. The average values of the Adjusted Rand index, along with the standard deviations at Bivariate4 dataset versions of 500, 1000, and 2500 generated bivariate points are presented in Table 1 for FADDI-S. The results support our view that the data set size is not important if the proportions of the cluster structure are maintained. According to the table, FADDI-S method achieves better results than the ones obtained by the five fuzzy clustering methods reported in [6].

Table 1: Adjusted Rand Index values for FADDI-S at different sizes of Bivariate4 dataset

Size	Adjusted Rand Index	Standard deviation
500	0.70	0.04
1000	0.70	0.03
2500	0.73	0.01

A remark:

The entity-to-feature format of the Bivariate4 data suggests that relational cluster analysis is not necessarily the best way to analyze it; a genuine data clustering method such as K-Means may bring better results. Indeed, an application of the ”intelligent” K-Means method from [30] to the original data size of $N = 5000$ has brought results with the average adjusted Rand index of 0.75 (the standard deviation 0.045), which is both higher and more consistent than the relational methods applied here and in [6].

3.3.2 Finding Community Structure

The research in finding community structure in ordinary graphs has been a subject of intense research (see, for example, [37, 36, 50, 26]). The graph with a set of vertices T is represented by the similarity matrix $A = (a_{tt'})$ between graph vertices such that $a_{tt'} = 1$ if t and t' are connected by an edge, and $a_{tt'} = 0$, otherwise. Then matrix A is symmetrized by the transformation $(A + A')/2$ after which all diagonal elements are made zero, $a_{tt} = 0$ for all $t \in T$. We assume that the graph is connected; otherwise, its connected components are to be treated separately.

The spectral relaxation involves subtraction of the “background” random interactions from similarity matrix $A = (a_{tt'})$. The random interactions are defined with the same within-row summary values $d_t = \sum_{t' \in T} a_{tt'}$ as those used in Laplace matrices. The random interaction between t and t' is defined as the product $d_t d_{t'}$ divided by the total number of edges [36]. The modularity criterion is defined as a usual, non-normalized cut, that is the summary similarity between clusters to be minimized, with thus transformed similarity data [36]. The modularity criterion has proven good in crisp clustering. This approach was extended to fuzzy clustering in the space of the first eigenvectors in [55].

Our approach allows for a straightforward application of FADDI-S algorithm to the network similarity matrix A. It also involves a transformation of the similarity data which is akin to the subtraction of background interactions in the modularity criterion [36]. Indeed we find initially the eigenvector z_1 corresponding to the maximum eigenvalue λ_1 of A itself. As is well known, this vector is positive because the graph is connected. Thus z_1 forms a fuzzy cluster itself, because it is conventionally normed. We do not count it as part of the cluster solution, though, because it expresses just the fact that all the entities are part of the same network. Thus, we proceed to the residual matrix with elements $a_{tt'} - \lambda_1 z_{1t} z_{1t'}$. We expect the matrix A to be rather “thin” with respect to the number of positive eigenvalues, which should allow for a natural halting the cluster extracting process when there are no positive eigenvalues at the residual matrix W.

We apply the FADDI-S algorithm to Zachary karate club network data, which serves as a prime test bench for community finding algorithms. This ordinary graph consists of 34 vertices, corresponding to members of the club and 78 edges between them - the data and references can be found, for example, in [37, 55]. The members of the club are divided according to their loyalties toward the club’s two prominent individuals: the administrator and instructor. Thus the network is claimed to consist of two communities, with 18 and 16 differently loyal members respectively.

Applied to this data, FADDI-S leads to three fuzzy clusters to be taken into account. Indeed, the fourth cluster accounts for just 2.4% of the data scatter, which is less than the inverse of the number of entities $\tau = 1/34$, reasonably suggested as a natural threshold value. Some characteristics of the found solution are presented in Table 2. All the membership values of the first cluster are positive - as mentioned above, this is just the first eigenvector; the positivity means that the network is well connected. The second and third FADDI-S clusters match the claimed structure of the network: they have 16 and 18 positive components, respectively, corresponding to the two observed groupings.

Table 2: Characteristics of Karate club clusters found with FADDI-S.

Cluster	Contribution, %	λ_1	Weight	Intensity
I	29.00	3.36	3.36	1.83
II	4.34	2.49	1.30	1.14
III	4.19	2.00	0.97	0.98

Let us compare our results with those of a recent spectral fuzzy clustering method developed in [55]. The latter method finds three fuzzy clusters, two of them representing the groupings, though with a substantial overlap between them, and the third, smaller cluster consisting of members 5,6,7,11,17 of just one of the groupings – see [55], p. 487. We think that this latter cluster may have come up from an eigenvector embracing the members with the largest numbers of connections in the network. It seems for certain that FADDI-S outperforms the method of [55] on Zachary club data.

4 Parsimonious Lifting Method

To generalize the contents of a thematic cluster, we propose a method for lifting it to higher ranks of the taxonomy so that 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 at this higher level of the ACM-CCS taxonomy (see Fig. 7). Depending on the extent of inconsistency between the cluster and the taxonomy, such lifting can be done differently, leading to different portrayals of the cluster on ACM-CCS tree depending on the relative weights of the events taken into account. A major event is the so-called “head subject”, a taxonomy node covering (some of) leaves belonging to the cluster, so that the cluster is represented by a set of head subjects. The penalty of the representation to be minimized is proportional to the number of head subjects so that the smaller that number the better. Yet the head subjects cannot be lifted too high in the tree because of the penalties for associated events, the cluster “gaps” and "offshoots", where their number depends on the extent of inconsistency of the cluster versus the taxonomy.

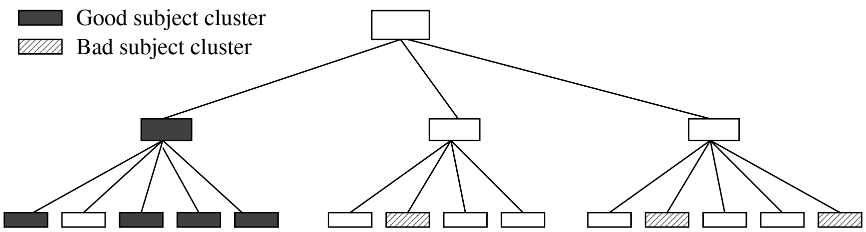


Fig. 7: Two clusters of second-layer topics, presented with checked and diagonal-lined boxes, respectively. The checked box cluster fits within one first-level category (with one gap only), whereas the diagonal line box cluster is dispersed among two categories on the right. The former fits the classification well; the latter does not.

The gaps are head subject's children topics that are not included in the cluster. An offshoot is a taxonomy leaf node that is a head subject (not lifted). It is not difficult to see that the gaps and offshoots are determined by the head subjects specified in a lifting (see Fig. 8).

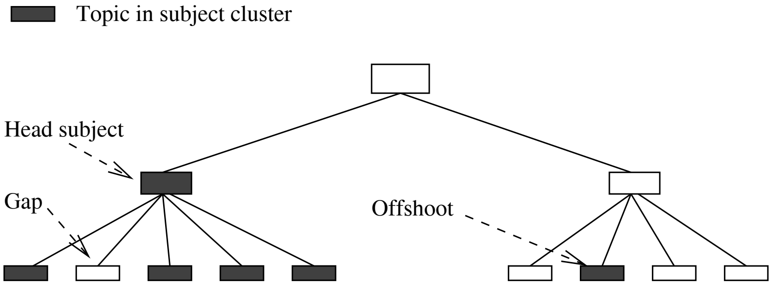


Fig. 8: Three types of features in lifting a subject cluster within taxonomy.

The total count of head subjects, gaps and offshoots, each weighted by both the penalties and leaf memberships, is used for scoring the extent of the cluster misfit needed for lifting a grouping of research topics over the classification tree. The smaller the score, the more parsimonious the lifting and the better the fit. Depending on the relative weighting of gaps, offshoots and multiple head subjects, different liftings can minimize the total misfit, as illustrated in Fig. 10 and Fig. 11 later.

Altogether, the set of topic clusters together with their optimal head subjects, offshoots and gaps constitute a parsimonious representation of the organization. Such a representation can be easily accessed and expressed. 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 awards) or distinguished by a special feature (i.e., industrial use or inclusion in a teaching program). Multiple head subjects and offshoots, when they 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 lift of a subject cluster can be achieved by recursively building a parsimonious representation for each node of the ACM-CCS tree based on parsimonious representations for its children as described in [34]. In this, we assume that any head subject is automatically present at each of the nodes it covers, unless they are gaps (as presented in Fig. 8). Our algorithm is set as a recursive procedure over the tree starting at leaf nodes.

The procedure [34] determines, at each node of the tree, sets of head subject gain and gap 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. 9, with each node assigned with sets of gap and head subject gain events under the two inheritance of head subject assumptions.

Let us denote the total penalty, to be minimized, under the inheritance and non-inheritance assumptions by p_i and p_n, respectively. A lifting result at a given node is defined by a pair of sets (H, G), representing the tree nodes at which events of head subject gains and gaps, respectively, have occurred in the subtree rooted at the node. We use (H_i, G_i) and (H_n, G_n) to denote lifting results under the inheritance and non-inheritance assumptions, respectively. The algorithm computes parsimonious representations 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 [32] for a mathematical problem in bioinformatics.

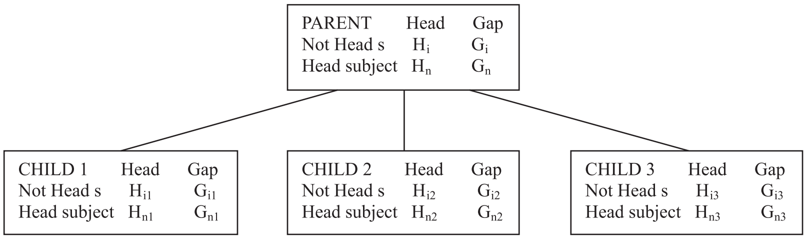

PARENT Head Gap
Not Head s H_i G_i
Head subject H_n G_n
CHILD 1 Head Gap
Not Head s H_{i1} G_{i1}
Head subject H_{n1} G_{n1}
CHILD 2 Head Gap
Not Head s H_{i2} G_{i2}
Head subject H_{n2} G_{n2}
CHILD 3 Head Gap
Not Head s H_{i3} G_{i3}
Head subject H_{n3} G_{n3}

Fig. 9: Events in a parent-children system according to a parsimonious lift scenario.

We present here only a version of the algorithm for crisp clusters obtained by a defuzzification step. Given a crisp topic cluster S, and penalties h, o and g for being a head subject, offshoot and gap, respectively, the algorithm is initialized as follows.

At each leaf l of the tree, either $H_n = \{l\}$, if $l \in S$, or $G_i = \{l\}$, otherwise. The other three sets are empty. The penalties associated are $p_i = 0$, $p_n = o$ if H_n is not empty, that is, if $l \in S$, and $p_i = g$, $p_n = 0$, otherwise. This is obviously a parsimonious arrangement at the leaf level.

The recursive step applies to any node t whose children $v \in V$ have been assigned with the two couples of H and G sets already (see Fig. 9 at which V consists of three children): $(H_i(v), L_i(v);\ H_n(v), L_n(v))$ along with associated penalties $p_i(v)$ and $p_n(v)$.

(I) Deriving the pair $H_i(t)$ and $G_i(t)$, under the inheritance assumption, the one of the following two cases is to be chosen depending on the cost:

(a) The head subject has been lost at t, so that $H_i(t) = \cup_{v\in V} H_n(v)$ and $G_i(t) = \cup_{v\in V} G_n(v) \cup \{t\}$. (Note different indexes, i and n in the latter expression.) The penalty in this case is $p_i = \Sigma_{v\in V}\, p_n(v) + g$;

or

(b) The head subject has not been lost at t, so that $H_i(t) = \emptyset$ (under the assumption that no gain can happen after a loss) and $G_i = \cup_{v\in V} G_i(v)$ with $p_i = \Sigma_{v\in V}\, p_i(v)$.

The case that corresponds to the minimum of the two p_i values is returned then.

(II) Deriving the pair $H_n(t)$ and $G_n(t)$, under the non-inheritance assumption, the one of the following two cases is to be chosen that minimizes the penalty p_n:

(a) The head subject has been gained at t, so that $H_n(t) = \cup_{v\in V} H_i(v) \cup \{t\}$ and $G_n(t) = \cup_{v\in V} G_i(s)$ with $p_n = \Sigma_{v\in V}\, p_i(v) + h$;

or (b) The head subject has not been gained at t, so that $H_n(t) = \cup_{v\in V} H_n(v)$ and $G_n = \cup_{v\in V} G_n(v)$ with $p_n = \Sigma_{v\in V}\, p_n(v)$.

After all tree nodes t have been assigned with the two pairs of sets, accept the H_n, L_n and p_n at the root. This gives a full account of the events in the tree.

This algorithm leads indeed to an optimal representation; its extension to a fuzzy cluster is achieved through using the cluster memberships in computing the penalty values at tree nodes [34].

5 Case Study

In order to illustrate our cluster-lift&visualization approach we are going to use data from two surveys of research activities conducted in two Computer Science organizations: (A) the research Centre of Artificial Intelligence, Faculty of Science & Technology, New University of Lisboa and (B) Department of Computer Science and Information Systems, Birkbeck, University of London. The ESSA survey tool was applied for data collection and maintenance (see Sect. 2.1).

Because one of the organizations, A, is a research center whereas the other, B, is a university department, one should expect that the total number of research topics in A is smaller than that in B, and, similarly, the number of clusters in A should be less than that in B. In fact, research centers are usually created for a limited set of research goals, whereas university departments must cover a wide range of topics in teaching, which relates to research efforts. These appear to be true: the number of

ACM-CCS third layer topics scored in A is 46 (out of 318) versus 54 in B. With the algorithm FADDI-S applied to the 46×46 and 54×54 topic-to-topic similarity matrices (see equation (1)), two fuzzy clusters (in case of center A) and four fuzzy clusters (in case of department B) have been sequentially extracted, after which the residual similarity matrix has become definite negative (stopping condition (1) of FADDI-S algorithm).

Let us focus our attention on the analysis of department B's research activities. On the clustering stage, as a result of the FADDI-S algorithm, four fuzzy clusters are obtained which are presented in Tables 3 and 4. Each of the topics in the tables is denoted by its ACM-CCS code and the corresponding string. The topics are sorted in the descending order of their cluster membership values (left columns of Tables 3 and 4). For each cluster, it is also presented its contribution to the data scatter, $G(\mathbf{u})$ (equation (5)), its intensity μ, and its weight ξ (equation (4)). Notice that the sum of clusters' contributions total to about 60%, which is a good result for clustering [2].

On the lifting stage, each of the found four clusters is mapped to and lifted in the ACM-CCS tree by applying the parsimonious lifting method with penalties for "head subjects" (h), "offshoots" (o) and "gaps" (g) of: $h = 1$, $o = 0.8$, and $g = 0.15$. We have chosen the gap penalty value considering that the numbers of children in ACM-CCS are typically around 10 so that two children belonging in the query would not be lifted to the parental node because the total gap penalty 8*0.15=1.2 would be greater than the decrease of head subject penalty 2-1=1. Yet if 3 of the children belong to the query, then it would be better to lift them to the parental node because the total gap penalty in this case, 7*0.15=1.05 would be smaller than the decrease of head subject penalty 3-1=2.

The parsimonious representation of the clusters in terms of the "head subjects", "offshoots", and "gaps" are described in Tables 5-8. Specifically, cluster 1 has as "head subject" *'D.2 SOFTWARE ENGINEERING'* with "offshoots" including *'C.2.4 Distributed Systems'*, *'D.1.6 Logic Programming'* and *'I.2.11 Distributed Artificial Intelligence'*. Cluster 2 is of *'J. Computer Applications'* with "offshoots" including *'G.2.2 Graph Theory'*, *'I.5.3 Clustering'*, *'K.6.0 General in K.6 - MANAGEMENT OF COMPUTING AND INFORMATION SYSTEMS'*, *'K.6.1 Project and People Management'*. Cluster 3 is described by the subjects (not lifted) *'E.2 DATA STORAGE REPRESENTATIONS'*, *'H.0 GENERAL in H. - Information Systems'*, *'I.0 GENERAL in I. - Computing Methodologies'*. Finally, cluster 4, with a more broad representation, has as "head subject" *'F. Theory of Computation'*, *'I.2 ARTIFICIAL INTELLIGENCE'*, and *'I.5 PATTERN RECOGNITION'*; its "offshoots" include *'E.1 DATA STRUCTURES'*, *'H.2.8 Database Applications'*, *'J.3 LIFE AND MEDICAL SCIENCES'* and *'K.3.1 Computer Uses in Education'*.

Let us illustrate the influence of the penalty parameters, more specifically the cost of gaps, g, on the parsimonious representation of cluster's research activities. Consider the scenario represented in Fig. 10 resulting from the lifting method with penalties of $h = 1$, $o = 0.8$, and $g = 0.3$. Due to the value of the gap penalty the cluster's topics (see Table 3) hold on as "leaf head subjects" as they are stated in the

[2] A 50% sum of clusters' contributions was obtained in the case of center A.

Table 3: Two clusters of research topics in department B

Cluster 1		
Contribution	26.7%	
Eigenvalue	37.44	
Intensity	5.26	
Weight	27.68	
Membership	**Code**	**Topic**
0.43055	K.2	HISTORY OF COMPUTING
0.39255	D.2.11	Software Architectures
0.35207	C.2.4	Distributed Systems
0.3412	I.2.11	Distributed Artificial Intelligence
0.3335	K.7.3	Testing, Certification, and Licensing
0.30491	D.2.1	Requirements/Specifications in D.2 Software Engineering
0.27437	D.2.2	Design Tools and Techniques in D.2 Software Engineering
0.24126	C.3	SPECIAL-PURPOSE AND APPLICATION-BASED SYSTEMS
0.19525	D.1.6	Logic Programming
0.19525	D.2.7	Distribution, Maintenance, and Enhancement in D.2 Software Engineering
Cluster 2		
Contribution	13.4%	
Eigenvalue	26.65	
Intensity	4.43	
Weight	19.60	
Membership	**Code**	**Topic**
0.66114	J.1	ADMINISTRATIVE DATA PROCESSING
0.29567	K.6.1	Project and People Management in K.6
0.29567	K.6.0	General in K.6 MANAGEMENT OF COMPUTING AND INF. SYSTEMS
0.29567	H.4.m	Miscellaneous in H.4 INF. SYSTEMS APPLICATIONS
0.29567	J.7	COMPUTERS IN OTHER SYSTEMS
0.2696	J.4	SOCIAL AND BEHAVIORAL SCIENCES
0.16271	J.3	LIFE AND MEDICAL SCIENCES
0.14985	G.2.2	Graph Theory
0.14593	I.5.3	Clustering
0.12307	I.6.4	Model Validation and Analysis
0.10485	I.6.5	Model Development

initialization of the lifting algorithm, being not lifted to higher ranks of the taxonomy (which would imply the appearance of some gaps). However, when decreasing the gap penalty from $g = 0.3$ to $g = 0.15$, it would lead to a different parsimonious generalization with subjects D.2.1, D.2.2, D.2.7 and D.2.11 generalized to "head subject" D.2, and the consequent assignment of the other subjects as "offshoots", as well as the occurrence of a set of gaps (i.e. the children of D.2 not present in the cluster). This scenario, described in Table 5, is visualized in Fig. 11.

Additionally, Fig. 11 illustrates the present visualization stage of our approach. Each cluster is individually visualized on the ACM-CCS subtree that covers the clusters' topics, represented as a tree plot with nodes labeled with the corresponding ACM-CCS subjects's code. The "head subjects", "gaps" and "offshoots" are marked with distinct graphical symbols: black circle for "head subjects" (or leaf

Table 4: Two other clusters of research topics in department B

Cluster 3		
Contribution	18.9%	
Eigenvalue	24.31	
Intensity	4.83	
Weight	23.31	
Membership	**Code**	**Topic**
0.613	E.2	DATA STORAGE REPRESENTATIONS
0.55728	I.0	GENERAL in I. Computing Methodologies
0.55728	H.0	GENERAL in H. Information Systems
Cluster 4		
Contribution	3.7%	
Eigenvalue	19.05	
Intensity	3.20	
Weight	10.26	
Membership	**Code**	**Topic**
0.35713	I.2.4	Knowledge Representation Formalisms and Methods
0.35636	F.4.1	Mathematical Logic
0.29495	F.2.0	General in F.2 ANAL. OF ALGORITHMS AND PROBLEM COMPLEXITY
0.28713	I.5.0	General in I.5 PATTERN RECOGNITION
0.28169	I.2.6	Learning
0.25649	K.3.1	Computer Uses in Education
0.24848	I.4.0	General in I.4 IMAGE PROCESSING AND COMPUTER VISION
0.24083	F.4.0	General in F.4 MATHEMATICAL LOGIC AND FORMAL LANGUAGES
0.18644	H.2.8	Database Applications
0.17707	H.2.1	Logical Design
0.17029	I.2.3	Deduction and Theorem Proving
0.15727	E.1	DATA STRUCTURES
0.15306	I.5.3	Clustering
0.14976	F.2.2	Nonnumerical Algorithms and Problems
0.14809	I.2.8	Problem Solving, Control Methods, and Search
0.14809	I.2.0	General in I.2 ARTIFICIAL INTELLIGENCE

head subjects), open circle for “gaps”, and dark grey square in case of “offshoots”. Also, the children of an “head subjects” that were “head subjects” before the current lifting stage are marked with grey circle.

A similar analysis had been performed concerning the representation of research activities in center A. The parsimonious representations of the two clusters found correspond to cluster 1 having as “head subject” *‘H. Information Systems’* and *‘I.5 PATTERN RECOGNITION’* with offshoots including *‘I.2.6 Learning’*, *‘I.2.6 Natural Language Processing’*, *‘I.4.9 Applications’*, *‘J.2 PHYSICAL SCIENCES AND ENGINEERING’*. Cluster 2 has as head subject *‘G. Mathematics of Computing’* and its “offshoots” include *‘F.4.1 Mathematical Logics’*, *‘I.2.0 General in I.2 - ARTIFICIAL INTELLIGENCE’*, *‘I.2.3 Deduction and Theorem Proving’* as well as *‘J.3 LIFE AND MEDICAL SCIENCES’*.

Overall, the surveys’ results analyzed in this study are consistent with the informal assessment of the research conducted in each of the research organizations.

Table 5: Parsimonious representation of department B cluster 1

	HEAD SUBJECT
D.2	SOFTWARE ENGINEERING
	OFFSHOTS
C.2.4	Distributed Systems
C.3	SPECIAL-PURPOSE AND APPLICATION-BASED SYSTEMS
D.1.6	Logic Programming
I.2.11	Distributed Artificial Intelligence
K.2	HISTORY OF COMPUTING
K.7.3	Testing, Certification, and Licensing
	GAPS
D.2.0	General in D.2 - SOFTWARE ENGINEERING
D.2.3	Coding Tools and Techniques
D.2.4	Software/Program Verification
D.2.5	Testing and Debugging
D.2.6	Programming Environments
D.2.8	Metrics
D.2.9	Management
D.2.10	Design
D.2.12	Interoperability
D.2.13	Reusable Software
D.2.m	Miscellaneous in D.2 - SOFTWARE ENGINEERING

Moreover, the sets of research topics that have been chosen by individual members at the ESSA survey follow the cluster structure rather closely, falling mostly within one of them.

6 Conclusion

We have proposed a novel method for knowledge generalization that employs a taxonomy tree. The method constructs fuzzy membership profiles of the entities constituting the system under consideration in terms of the taxonomys leaves, and then it generalizes them in two steps. These steps are:

(i) fuzzy clustering research topics according to their thematic similarities, ignoring the topology of the taxonomy, and

(ii) lifting clusters mapped to the taxonomy to higher ranked categories in the tree.

Table 6: Parsimonious representation of department B cluster 2

	HEAD SUBJECT
J.	Computer Applications
	OFFSHOTS
G.2.2	Graph Theory
H.4.m	Miscellaneous in H.4 - INFORMATION SYSTEMS APPLICATIONS
I.5.3	Clustering
I.6.4	Model Validation and Analysis
I.6.5	Model Development
K.6.0	General in K.6 - MANAGEMENT OF COMPUTING AND INFORMATION SYSTEMS
K.6.1	Project and People Management
	GAPS
J.0	GENERAL in J. - Computer Applications
J.2	PHYSICAL SCIENCES AND ENGINEERING
J.5	ARTS AND HUMANITIES
J.6	COMPUTER-AIDED ENGINEERING
J.m	MISCELLANEOUS in J. - Computer Applications

Table 7: Parsimonious representation of department B cluster 3

	SUBJECTS
E.2	DATA STORAGE REPRESENTATIONS
H.0	GENERAL in H. - Information Systems
I.0 -	GENERAL in I. - Computing Methodologies

These generalization steps thus cover both sides of the representation process: the empirical – related to the structure under consideration – and the conceptual – related to the taxonomy hierarchy.

Potentially, this approach could lead to a useful instrument for comprehensive visual representation of developments in any field of organized human activities.

However, there are a number of issues remaining to be tackled. They relate to all main aspects of the project: (a) data collection, (b) thematic clustering and (c) lifting. On the data collection side, the mainly manual e-survey ESSA tool should be supported by an automated analysis and rating of relevant research documents including those on the internet. The FADDI-S method, although already experimentally proven competitive to a number of existing methods, should be further explored and more thoroughly investigated. The issue of defining right penalty weights for parsimonious cluster lifting should be addressed.

Table 8: Parsimonious representation of department B cluster 4

	HEAD SUBJECTS
F.	Theory of Computation
I.2	ARTIFICIAL INTELLIGENCE
I.5	PATTERN RECOGNITION
	OFFSHOTS
D.2.8	Metrics
E.1	DATA STRUCTURES
G.2.2	Graph Theory
H.2.1	Logical Design
H.2.8	Database Applications
I.4.0	General in I.4 - IMAGE PROCESSING AND COMPUTER VISION
J.3	LIFE AND MEDICAL SCIENCES
K.3.1	Computer Uses in Education
	GAPS
F.0	GENERAL in F. - Theory of Computation
F.1	COMPUTATION BY ABSTRACT DEVICES
F.2.1	Numerical Algorithms and Problems
F.2.3	Tradeoffs between Complexity Measures
F.2.m	Miscellaneous in F.2 - ANAL. OF ALGORITHMS AND PROBLEM COMPLEXITY
F.3	LOGICS AND MEANINGS OF PROGRAMS
F.4.2	Grammars and Other Rewriting Systems
F.4.3	Formal Languages
F.4.m	Miscellaneous in F.4 - MATHEMATICAL LOGIC AND FORMAL LANGUAGES
F.m	MISCELLANEOUS in F. - Theory of Computation
I.2.1	Applications and Expert Systems
I.2.2	Automatic Programming
I.2.5	Programming Languages and Software
I.2.7	Natural Language Processing
I.2.9	Robotics
I.2.10	Vision and Scene Understanding
I.2.11	Distributed Artificial Intelligence
I.2.m	Miscellaneous in I.2 - ARTIFICIAL INTELLIGENCE
I.5.1	Models
I.5.4	Applications
I.5.5	Implementation
I.5.m	Miscellaneous in I.5 - PATTERN RECOGNITION

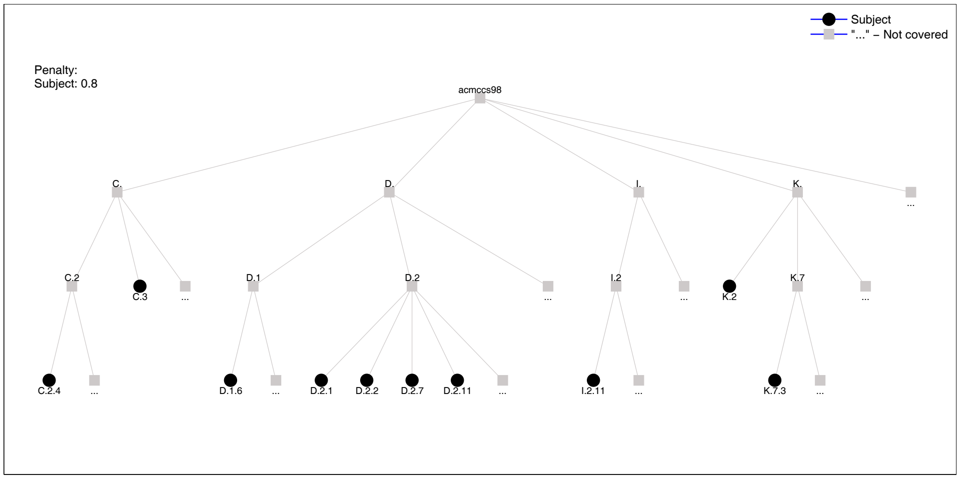


Fig. 10: Parsimonious representation lift of department B cluster 1 within the ACM-CCS tree with penalties of $h = 1$, $o = 0.8$, and $g = 0.3$

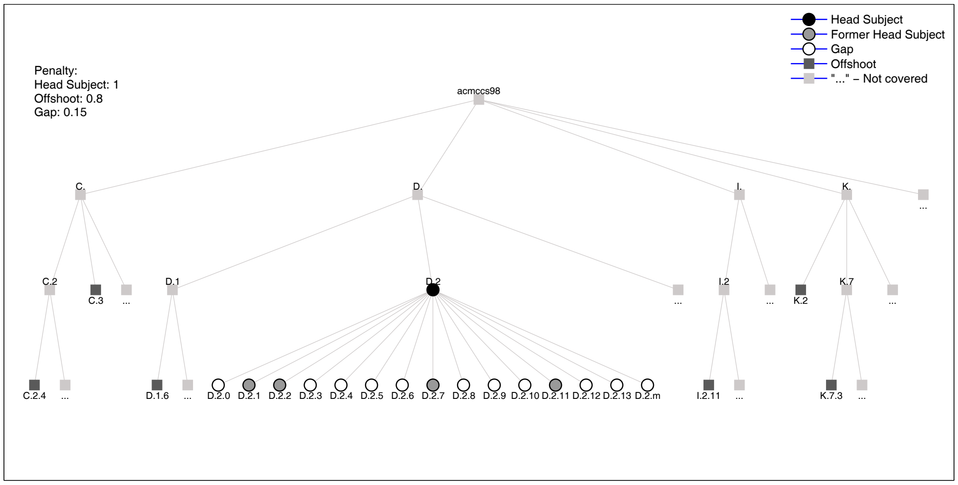


Fig. 11: Parsimonious representation lift of department B cluster 1 within the ACM-CCS tree with penalties of $h = 1$, $o = 0.8$, and $g = 0.15$

Moreover, further investigation should be carried out with respect to the extension of this approach to more complex structures than the hierarchical tree taxonomy, ontology structures. Finally, there remains to be explored the usage of the cluster and head subjects information in query answering, and its visualization; as well as the updating of taxonomies (or other structures) on the basis of the empirical data found, possibly involving aggregating data from multiple organizations.

Acknowledgements The authors are grateful to the staff members of the department of Computer Science, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Centre for Artificial Intelligence, Universidade Nova de Lisboa, Lisboa, Portugal, and department of Computer Science, Birkbeck University of London, London, UK, that participated in the surveys. Igor Guerreiro is acknowledged for developing software for the ESSA tool. Rui Felizardo is acknowledged for developing software for the lifting algorithm with interface shown in Fig. 10 and Fig. 11. This work has been supported by grant PTDC/EIA/69988/2006 from the Portuguese Foundation for Science & Technology. The partial financial support of the Laboratory of Choice and Analysis of Decisions at the State University – Higher School of Economics, Moscow RF, is also acknowledged.

References

1. ACM Computing Classification System (1998) http://www.acm.org/about/class/1998. Cited 9 Sep 2008
2. Advanced Visual Systems (AVS) http://www.avs.com/solutions/avs-powerviz/utility_distribution.html. Cited 27 Nov 2010
3. Beneventano, D., Dahlem, N., El Haoum, S., Hahn, A., Montanari, D., and Reinelt, M.: Ontology-driven semantic mapping. Enterprise Interoperability III, Part IV, Springer, 329-341 (2008)
4. Bezdek, J., Hathaway, R.J., Windham, M.P.: Numerical comparisons of the RFCM and AP algorithms for clustering relational data. Pattern Recognition **24**, 783-791 (1991)
5. Bezdek, J., Keller, J., Krishnapuram, R., Pal, T.: Fuzzy Models and Algorithms for Pattern Recognition and Image Processing. Kluwer Academic Publishers (1999)
6. Brouwer, R.: A method of relational fuzzy clustering based on producing feature vectors using FastMap. Information Sciences 179, 3561-3582 (2009)
7. Buche, P., Dibie-Barthelemy, J., and Ibanescu, L.: Ontology mapping using fuzzy conceptual graphs and rules. In: ICCS Supplement., 1724 (2008)
8. Cali, A., Gottlob, G., Pieris, A.: Advanced processing for ontological queries. Proceedings of the VLDB Endowment Vol. 3, no. 1, 554-565 (2010)
9. Davé, R.N., Sen, S.: Robust fuzzy clustering of relational data. IEEE Transactions on Fuzzy Systems, **10**, 713-727 (2002)
10. Ding, Y., Foo, S.: Ontology research and development. Journal of Information Science **28**(5), 375-388 (2002)
11. Dotan-Cohen D., Kasif S., Melkman A.: Seeing the forest for the trees: using the gene ontology to restructure hierarchical clustering. Bioinformatics **25**(14), 1789-1795 (2009)
12. Eick S.G.: Visualizing online activity. Communications of the ACM **44**(8), 45-50 (2001)
13. 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, 6 (2003)
14. Freudenberg, J. M., Joshi V.K., Hu Z., Medvedovic M.: CLEAN: CLustering Enrichment ANalysis. BMC Bioinformatics 10:234 (2009)
15. Gahegan, M., Agrawal, R., Jaiswal, A., Luo, J., and Soon, K.-H.: A platform for visualizing and experimenting with measures of semantic similarity in ontologies and concept maps. Transactions in GIS **12**(6), 713-732 (2008)
16. Gaevic, D., Hatala, M.: Ontology mappings to improve learning resource search. British Journal of Educational Technology **37**(3), 375 - 389 (2006)
17. Georgeon, O.L., Mille, A., Bellet, T., Mathern, B., Ritter F.: Supporting activity modeling from activity traces, Expert Systems: The Journal of Knowledge Engineering (submitted) (2010)

18. The Gene Ontology Consortium. The Gene Ontology project in 2008. Nucleic Acids Research **36** (Database issue): D4404. doi:10.1093/nar/gkm883. PMID 17984083 (2008)
19. Ghazvinian, A., Noy, N., Musen, M.: Creating mappings for ontologies in Biomedicine: simple methods work. AMIA 2009 Symposium Proceedings, 198-202 (2009)
20. Guh, Y.Y., Yang, M.S., Po, R.W., Lee, E.S.: Establishing performance evaluation structures by fuzzy relation-based cluster analysis. Computers and Mathematics with Applications **56**, 572-582 (2008)
21. Hathaway, R.J., Davenport, J.W., Bezdek, J.C.: Relational duals of the c-means algorithms. Pattern Recognition **22**, 205-212 (1989)
22. Hathaway, R.J., Bezdek, J.C.: NERF c-means: Non-Euclidean relational fuzzy clustering. Pattern Recognition 27, 429-437 (1994)
23. Huang, L., Yan, D., Jordan, M.I., Taft, N.: Spectral clustering with perturbed data. In: Koller, D., Schuurmans, D., Bengio, Y., Bottou, L. (Eds.): Advances in Neural Information Processing Systems 21, Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems (Vancouver), MIT Press, 705-712 (2009)
24. Hubert, L.J., Arabie, P.: Comparing partitions. Journal of Classification **2**, 193-218 (1985)
25. Liu, J., Wang, W., Yang, J.: Gene ontology friendly biclustering of expression profiles. Proc. of the IEEE Computational Systems Bioinformatics Conference, IEEE, 436-447 (2004)
26. von Luxburg, U.: A tutorial on spectral clustering. Statistics and Computing 17, 395-416 (2007)
27. Marinica, C., Guillet, F.: Improving post-mining of association rules with ontologies, The XIII International Conference Applied Stochastic Models and Data Analysis (ASMDA), ISBN 978-9955-28-463-5, 76-80 (2009)
28. Mazza, R.: Introduction to Information Visualization. Springer, ISBN: 978-1-84800-218-0 (2009)
29. McLachlan G.J., Khan N.: On a resampling approach for tests on the number of clusters with mixture model based clustering of tissue samples. J. Multivariate Anal. **90**, 90105 (2004)
30. 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, 242-245 (2005)
31. Mirkin, B.: Additive clustering and qualitative factor analysis methods for similarity matrices. Journal of Classification **4**(1), 7-31 (1987)
32. 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)
33. Mirkin, B., Nascimento, S., Pereira, L.M.: Cluster-lift method for mapping research activities over a concept tree. In: Koronacki, J., Wierzchon, S.T., Ras, Z.W., Kacprzyk, J. (eds.), Recent Advances in Machine Learning II, Computational Intelligence Series Vol. 263, Springer, pp. 245-258 (2010)
34. Mirkin, B., Nascimento, S., Fenner, T., Pereira, L.M.: Constructing and Mapping Fuzzy Thematic Clusters to Higher Ranks in a Taxonomy. In: Bi, Y., Williams, M.A. (eds.), 4th Intl. Conf. on Knowledge Science, Engineering & Management (KSEM 2010), Springer LNAI 6291, pp. 329-340 (2010).
35. Mirkin, B., Nascimento, S.: Additive Spectral Method for Fuzzy Cluster Analysis of Similarity Data Including Community Structure and Affinity Matrices. Information Sciences, Springer, (to appear).
36. Newman, M.: Finding community structure in networks using the eigenvectors of matrices. Phys. Rev. E, 74 036104 (2006)
37. Newman, M., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E, 69 026113 (2004)
38. Ng, A., Jordan, M. Weiss, Y.: On spectral clustering: analysis and an algorithm. In: Ditterich, T.G., Becker, S., Ghahramani, Z. (Eds.), Advances in Neural Information Processing Systems, 14, MIT Press, Cambridge Ma., 849-856 (2002)

39. OWL 2 Web Ontology Language Overview (2009) http://www.w3.org/TR/2009/RECowl2-overview20091027/. Cited 27 Nov 2010
40. Roubens, M.: Pattern classification problems and fuzzy sets. Fuzzy Sets and Systems 1, 239-253 (1978)
41. Sato, M., Sato, Y., Jain, L.C.: Fuzzy Clustering Models and Applications. Physica-Verlag, Heidelberg (1997)
42. Schattkowsky T., and Frster A.: (2007) On the pitfalls of UML-2 activity modeling, International Workshop on Modeling in Software Engineering (MISE'07), 1-6.
43. Skarman, A., Jiang, L., Hornshoj, H., Buitenhuis, B., Hedegaard, J., Conley, L., Sorensen, P.: Gene set analysis methods applied to chicken microarray expression data. BMC Proceedings 3(Suppl 4) (2009)
44. Shepard, R.N., Arabie, P.: Additive clustering: representation of similarities as combinations of overlapping properties. Psychological Review 86, 87-123 (1979)
45. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **22**(8), 888-905 (2000)
46. SNOMED Clinical Terms (2010) http://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html. Cited 27 Nov 2010
47. Sosnovsky, S., Mitrovic, A., Lee, D., Prusilovsky, P., Yudelson, M., Brusilovsky, V., and Sharma, D.: Towards integration of adaptive educational systems: mapping domain models to ontologies. In Dicheva, D., Harrer, A., Mizoguchi, R. (eds.), Procs. of 6th International Workshop on Ontologies and Semantic Web for ELearning (SWEL'2008) at ITS2008 (2008) (Available at: urlhttp://compsci.wssu.edu/iis/swel/SWEL08/Papers/Sosnovsky.pdf).
48. Thomas, H., O'Sullivan, D., and Brennan, R.: Evaluation of ontology mapping representation. Proceedings of the Workshop on Matching and Meaning, 64-68 (2009)
49. Windham, M.P.: Numerical classification of proximity data with assignment measures. Journal of Classification **2**, 157-172 (1985)
50. White, S., Smyth, P.: A spectral clustering approach to finding communities in graphs. SIAM International Conference on Data Mining, (2005)
51. Thorne, C., Zhu, J., Uren, V.: Extracting domain ontologies with CORDER. Tech. Reportkmi-05-14. Open University, 1-15 (2005)
52. Yang, M.S., Shih, H.M.: Cluster analysis based on fuzzy relations. Fuzzy Sets and Systems **120**, 197-212 (2001)
53. 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), 42-52 (2005)
54. Zadeh, L.A.: Fuzzy sets. Information and Control **8**, 338-353 (1965)
55. Zhang, S., Wang, R.-S., Zhang, X.-S.: Identification of overlapping community structure in complex networks using fuzzy c-means clustering. Physica A 374, 483-490 (2007)

Contents