

Hybrid Clustering of Text Mining and Bibliometrics Applied to Journal Sets

Xinhai Liu ^{*†‡} Shi Yu ^{*†} Yves Moreau[†] Bart De Moor[†] Wolfgang Glänzel [§]
Frizo Janssens^{†¶}

Abstract

To obtain correlated and complementary information contained in text mining and bibliometrics, hybrid clustering to incorporate textual content and citation information has become a popular strategy. In this paper, we propose a new computational framework of integrating text mining and bibliometrics to provide a mapping of journal sets. Two different approaches of hybrid clustering methods are applied in this paper. The first category is ensemble clustering, which combines different clustering results obtained from individual data into a consolidated clustering result. The second category is kernel fusion, which maps heterogeneous data sets into the kernel space and combines the kernel matrices for clustering. Kernels can be combined either averagely, or by an optimized weighted linear combination model. In this paper, we propose a novel adaptive kernel K-means clustering algorithm to combine textual content and citation information for clustering. The proposed algorithm is systematically compared with other methods on a clustering problem of 1869 journals published in 2002-2006. Based on several validation indices, the experimental results demonstrate that our hybrid clustering strategy is able to provide clustering result as well as the best individual data source.

1 Introduction

In information science studies, unsupervised learning methods such as clustering are helpful to get the structure mapping of science or technology fields. It is also useful to detect new emerging fields or hot topic in the long term. In previous research, two types of data sources, text mining data and citation data [15], were often used separately to obtain structure mapping. On one hand, text mining data is useful to indicate similarities at the textual level, but is also affected by the

ambiguities of vocabularies. On the other hand, citation data is able to measure the relevance of journals by their citation links, while it lacks the information about document similarities at the semantic level. Since the information contained in these two data sources is highly correlated and complementary, the combination of them seems to be a promising approach for clustering analysis (hybrid clustering).

Hybrid clustering is a technique to integrate multiple information sources for clustering. A single dataset can be regarded as a description of a problem sliced by a specific conceptual view; combining multiple views might be helpful to obtain a comprehensive understanding of the problem. In particular, if the information contained in multiple views is correlated and complementary, clustering with the integrated information might improve the effectiveness of data partitions. In this paper, we try various hybrid clustering algorithms and these algorithms can be roughly divided into two main categories. The first category is ensemble clustering, which combines the *partitions* of different data sources into a new consolidate clustering. The second approach is kernel fusion, which combines the *similarity matrices* (or *distance matrices*) of multiple sources as a new individual data for clustering. Although many methodologies have been proposed in both of these approaches, unfortunately, there has been relatively few effort invested to investigate and compare them in a general framework. Thus in this paper, we investigate different hybrid clustering methods in a unified framework. The comparison is addressed on a real application of integrating text analysis and citation analysis to obtain the structure mapping of journal sets. We also discuss the problem of evaluation for hybrid clustering, which is important and useful in the sense of extending clustering model comparison and prediction from single data to multiple data sets.

The organization of this paper is as follows. Section 2 is a brief review of previous approaches. In Section 3 and Section 4, we introduce hybrid clustering methodologies from two main categories: clustering ensemble and kernel fusion. Section 5 presents the clustering evaluation methods applied in this paper. The description

^{*}Xinhai Liu and Shi Yu are equally contributed authors for this paper

[†]K.U. Leuven, Dept. of Electrical Engineering ESAT-SCD, Kasteelpark Arenberg 10, B-3001 Leuven (Belgium)

[‡]Wuhan University of Science and Technology, College of Information Science and Engineering, 430081, Wuhan, China

[§]K.U. Leuven, Steunpunt O&O Indicatoren, Dekenstraat 2, B-3000 Leuven (Belgium)

[¶]Attentio, Studio TROPE, Bloemenstraat 32, B-1000, Brussels, Belgium

of experimental data and analysis of experimental results are presented in Section 6 and Section 7, respectively. In section 8, we address some caveats and ongoing topics in hybrid clustering. The final conclusion is made in section 9.

2 Related Research

In the literature, the idea of combining bibliometric or citation information with text mining data has been reported in different applications. In information retrieval, Plachouras [21] presented a query-based interface for web ranking. It combines the rank lists obtained from text-content and from citation analysis on the basis of Dempster-Shafers evidence theory. By assigning each source an uncertainty measure between the evidence and the query, it provides a hybrid ranking mechanism for web information. In bibliometric mapping, Braam and his colleagues combined co-citation analysis with word analysis to improve the efficiency of co-citation based clustering [2]. Kostoff made a survey about the integration of full-text based techniques with bibliometric methodologies [16]. In document clustering analysis, Modha and Spangler [19] introduced a toric k-means algorithm to cluster web documents using terms, out-links and in-links, which actually combines text and citation information together. Zhang and his colleagues [31] use genetic programming to optimize the document classification model which integrates citation-based information and structural content.

Recently, hybrid clustering has also been applied to the structure mapping of journal sets or paper sets. Janssens [15] adopts a clustering method based on weighted linear combination of distance matrices (WLCDM) which combines the distance measure of documents obtained from text and citations. Since the linear combination of distances might neglect the differences of distributions of various data sources, an algorithm based on Fisher’s inverse chi-square (FICSM) [12] was further proposed to combine p-values instead of distances from various data sources.

3 Ensemble Clustering

3.1 Definition Ensemble clustering, also known as clustering aggregation or consensus clustering, combines different clustering partitions into a consolidated partition. The consolidated partition is usually obtained by some consensus functions, for example, to maximize the average mutual information, or, to minimize the average squared distance between the consolidated partition with the individual partitions. Ensemble clustering was originally proposed for a single type of data only, thus various individual partitions are usually generated in two scenarios: 1) choice of data representa-

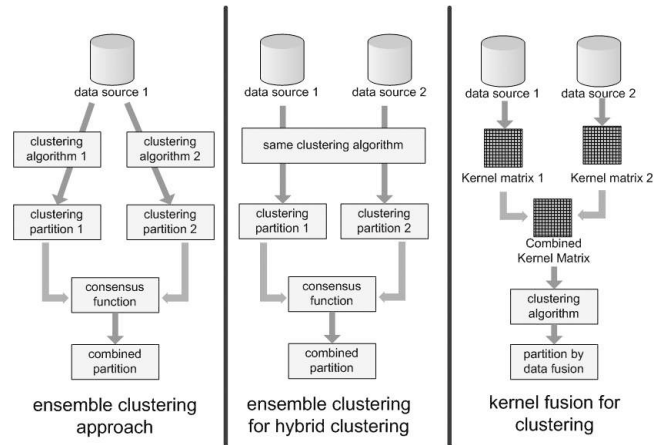


Figure 1: A conceptual overview of ensemble clustering and kernel fusion methods for hybrid clustering analysis

tion and 2) choice of clustering algorithms or algorithmic parameters. In the first scenario, different representations of data may be produced by: a) employing different pre-processing or feature extraction methods, which result in different pattern representations (vectors, strings, graphs, etc.); b) exploring subspaces of the same data representations, such as using subsets of features or applying different dimensionality reduction methods; c) randomly perturbing the data, such as bootstrapping or sampling. In the second scenario, the multiple data partitions may be obtained by: a) applying different clustering algorithms; b) applying the same school of algorithms but with different algorithmic parameters; c) keeping the algorithm and parameter the same, but using different dissimilarity measures (for example, different distance measure) for evaluating inter-pattern relationships.

The strategy of ensemble clustering can be straightforwardly extended to the hybrid clustering problem, where the main difference is that various individual partitions are now obtained from different data sources. Within the ensemble framework, if the information contained in multiple sources is highly correlated, partitions obtained from multiple data sources should also contain some “common agreement”, thus a consolidated partition can also be obtained.

3.2 Algorithms In this paper, we extend and apply several well-known ensemble algorithms proposed in the literature to hybrid clustering. A conceptual overview of these algorithms is shown in Figure 1 and various algorithms mainly vary on the choice of different consensus functions.

HGPA, CSPA, MCLA Strehl and Ghosh [23] for-

mulate the optimal consensus as the partition that shares the most information with the partitions to combine, as measured by the Average Normalized Mutual Information. They use three heuristic consensus algorithms based on graph partitioning, called Cluster-based Similarity Partitioning Algorithm (CSPA), Hyper Graph Partitioning Algorithm (HGPA) and Meta Clustering Algorithm (MCLA) to obtain the combined partition.

QMI Topchy [25] formulated the combination of partitions as a categorical clustering problem. His method adopts a category utility function [18] that evaluates the quality of a “median partition” as a summary of the ensemble. He proves that maximizing this category utility function implies the same ensemble clustering criterion as maximizing the generalized mutual information based on quadratic entropy. Furthermore, the maximization of the category utility function is equivalent to the square-error based clustering criterion when the number of clusters is fixed. The final consensus partition is obtained by applying the K-Means algorithm on the feature space transformed by the category utility function.

EACAL Fred and Jain [8] introduce the concept of evidence accumulation clustering (EAC) that maps the individual data partitions as an ensemble clustering by constructing a co-association matrix. The entries of the co-association matrix are interpreted as votes on the pairwise co-occurrence of objects, which is computed as the number of times each pair of objects appears in the same cluster of a individual partition. Then the final consensus partition is obtained by applying single-link (SL) and average-link (AL) methods on the co-association matrix. According to their experiments, average linkage performs better than single linkage, so in this paper we apply EAC-AL for comparison.

adacVote Ayad and Kamel proposed [1] a “cumulative vote weighting method” to compute an empirical probability distribution summarizing the ensemble. The goal of this ensemble is to minimize the average squared distance between the mapped partitions and the combined partition. The cumulative voting method seeks an adaptive reference partition and incrementally updates it by averaging other partitions to relax the dependence of the combined partition on the selected reference. In the adaptive cumulative voting (ACV) algorithm they proposed, the partitions are combined in decreasing order of the entropy.

The software containing HGPA, CSPA and MCLA algorithms was downloaded from the author’s website. We implement QMI, EAC-AL and adacVote algorithms in MATLAB. In our experiments, the individual partitions are all obtained by K-means clustering. All the experimental results presented in this paper, if without special note, are obtained from 50 random repetitions.

4 Kernel Fusion Algorithms

The main difference of the kernel fusion approach is that the integration is carried out in kernel space before the clustering algorithm is applied (early integration) while ensemble clustering fuses partitions after clustering (late integration). Kernel methods provide an elegant way to combine data because kernel mapping resolves the heterogeneities of data sources and represents them as same-size kernel matrices. Moreover, if we assume that the importance of each data source is equivalent, we can combine the kernels in an average manner, thus the issue of data integration is then transparent to the pattern analysis problem. The averagely combined kernel can be regarded as a new individual data source and the partition can be obtained by standard clustering algorithms in kernel space. A more machine-intelligent approach is to couple the optimization problem of kernel learning with the objective function of pattern analysis so that the weights assigned to each data source can be adjusted adaptively during the clustering procedure [4]. In this section, we propose a novel adaptive kernel K-means algorithm to do clustering and weights learning simultaneously. We also propose algorithms based on average combination of kernels.

4.1 Adaptive Kernel K-means Clustering (AKKC)

4.1.1 Objective Function The standard K-means hard clustering algorithm adopt the squared Euclidean distance to measure the dissimilarity between vectors x_i and cluster representatives θ_j . The membership coefficient u_{ij} , which indicates whether the i -th sample belongs to the j -th cluster, is either 1 or 0. Then the cost function of K-means clustering becomes

$$(4.1) \quad J(\theta, U) = \sum_{i=1}^n \sum_{j=1}^k u_{ij} \|x_i - \theta_j\|^2.$$

For clustering based on data integration, the *Mahalanobis* distance is preferred because it is invariant to any nonsingular linear transformation [26]. It scales the distance between two objects by the inverse covariance matrix,

$$(4.2) \quad d_M(x_i, x_j) = [(x_i - x_j)^T C^{-1} (x_i - x_j)]^{\frac{1}{2}},$$

where C is the covariance matrix defined as follows,

$$(4.3) \quad C = \frac{1}{n} \sum_{i=1}^n (x_i - \mu)(x_i - \mu)^T,$$

and $\mu = \frac{1}{n} \sum_{i=1}^n x_i$ is the mean of $\{x_i\}_{i=1}^n$.

To avoid the singularity of the covariance matrix, a regularized covariance matrix is often used as

$$(4.4) \quad C = \frac{1}{n} \sum_{i=1}^n (x_i - \mu)(x_i - \mu)^T + \lambda I,$$

where I is the identity matrix and $\lambda > 0$ is the regularization parameter.

Using the distance measure defined above, K -means-like clustering can be regarded as partitioning the data $\{x_i\}_{i=1}^n$ into k disjoint clusters, $\{l_1, l_2, \dots, l_k\}$, which minimize the *Within Clusters Sum of Square Error* (WSSE),

$$(4.5) \quad \text{WSSE}(\{l_j\}_{j=1}^k) = \sum_{j=1}^k \sum_{x_i \in l_j} d_M(x_i, \mu_j)^2,$$

where $d_M(\cdot, \cdot)$ is the *Mahalanobis* distance defined in (4.2) and μ_j is the mean of the j -th cluster l_j . For a given data set, the summation of all pairwise distances is a constant value hence the minimization of WSSE is equal to the maximization of *Between Clusters Sum of Square Error* (BSSE) defined as follows,

$$(4.6) \quad \text{BSSE}(\{l_j\}_{j=1}^k) = \sum_{j=1}^k |l_j| d_M(\mu_j, \hat{\mu})^2,$$

where $|l_j|$ is the cardinality of samples in cluster j , μ_j is the mean of the j -th cluster l_j , and $\hat{\mu}$ is the global mean of $\{x_i\}_{i=1}^n$.

The BSSE can be expressed in a compact matrix form as

$$(4.7) \quad \text{BSSE}(\{l_j\}_{j=1}^k) = \text{trace}(L^T X^T C^{-1} X L),$$

where X is the data matrix, L is the *weighted cluster indicator matrix* $L = [l_1, l_2, \dots, l_k]$ defined as

$$(4.8) \quad L = F(F^T F)^{-\frac{1}{2}},$$

where F is the $n \times k$ cluster indicator matrix defined as follows,

$$(4.9) \quad F = f_{i,j}{}_{n \times k}, \text{ where } f_{i,j} = \begin{cases} 1 & \text{if } x_i \in l_j \\ 0 & \text{if } x_i \notin l_j \end{cases}.$$

4.1.2 Kernel Extension The objective function defined in (4.7) can be extended into kernel space by a mapping implicitly specified by a symmetric kernel function Ω , which computes the inner product of the pairwise data in kernel space, that is

$$(4.10) \quad \Omega(x_i, x_j) = (\phi(x_i), \phi(x_j)),$$

where x_i, x_j are data points in the original space, $\phi(\cdot)$ is the kernel mapping. Let $\phi_\Omega(X)$ denotes the data matrix in kernel space defined by kernel mapping Ω , therefore in kernel space the objective function for clustering can be formulated as the following trace maximization problem:

$$(4.11) \quad \max_{\Omega, L} \text{trace}(L^T \phi_\Omega(X)^T C_\Omega^{-1} \phi_\Omega(X) L).$$

We assume that the data in feature space has been centered, so the regularized covariance matrix has the form as,

$$(4.12) \quad C_{\hat{\Omega}} = \phi_{\hat{\Omega}}(x) \phi_{\hat{\Omega}}(x)^T + \lambda I = \hat{\Omega} + \lambda I,$$

where $\hat{\Omega}$ is the centered kernel matrix on X . For simplicity, from now on we denote Ω as the centered kernel matrix on X . The problem of clustering can also be addressed in the framework of kernel fusion. Given a set of p centered kernel matrices, the optimal kernel matrix Ω^* that optimizes the objective function

$$(4.13) \quad \max_{\Omega, L} \text{trace} \left(L^T \phi_\Omega(x)^T (\phi_\Omega(x) \phi_\Omega(x)^T + \lambda I)^{-1} \phi_\Omega(x) L \right),$$

is defined as the convex linear combination of p centered kernel matrices

$$\Omega = \left\{ \sum_{i=1}^p \mu_i \Omega_i \mid \sum_{i=1}^p \mu_i r_i = 1, \mu_i > 0, r_i = \text{trace}(\Omega_i) \right\},$$

where μ_i is the weight assigned on each data source.

4.1.3 Algorithm Now the task of clustering by data fusion is to find the optimal partition of data L^* and the optimal combination of kernels Ω^* that maximizes the objective function defined in (4.13). Solving L^* and Ω^* simultaneously is very difficult, so an alternative minimization framework [5] is applied to solve L and Ω iteratively.

Algorithm 4.1: ADAPTIVEKMEANS($\Omega_1, \Omega_2, \dots, \Omega_p, K, \lambda$)

$L_{(0)} \leftarrow$ KERNEL K-MEANS CLUSTERING(Ω_s), $s \in 1, \dots, p$

comment: obtain an initial partition

while <!convergence >

do $\left\{ \begin{array}{l} \text{step1: } \Omega \leftarrow \text{ADAPTIVE WEIGHING}(L_{(i)}, \Omega_1, \Omega_2, \dots, \Omega_p, \lambda) \\ \text{step2: } L_{(i+1)} \leftarrow \text{KERNEL K-MEANS CLUSTERING}(\Omega, K) \end{array} \right.$

comment: i is the counter of iteration

return (L_{i+1})

It can be proved that the proposed algorithm converges locally because the *step 1* and *step 2* are optimizing towards the same objective function. The *adaptive weighing* procedure is related to many existing methods in supervised learning since the label information is contained in L [17, 27]. The solution is represented in the following theorem and it is formulated as a QCQP problem and solved in MOSEK toolbox.

THEOREM 4.1. *Let L be the weighted cluster indicator matrix of multiple clusters as mentioned in (8), $\phi(x)$ as the data matrix in kernel space, Ω as the kernel Gram matrix defined as $\Omega_{i,j} = \phi(x_i)\phi(x_j)^T$, $\lambda > 0$ as the regularization parameter on covariance matrix, given a set of n centered kernel matrices $\Omega_1, \dots, \Omega_n$, the optimal kernel matrix Ω as the convex linear combination of n matrices which optimize the objective function F_1 of clustering*

$$(4.14) \quad F_4 = \max_{\Omega} \text{trace} \left(L^T \phi(x)^T (\phi(x)\phi(x)^T + \lambda I)^{-1} \phi(x)L \right),$$

can be found by solving the following convex QCQP problem:

$$(4.15) \quad \begin{aligned} \max_{\beta_j, t} \quad & - \sum_{j=1}^k \frac{1}{4} \beta_j^T \beta_j - \frac{1}{4\lambda} t + \sum_{j=1}^k \beta_j^T l_j \\ \text{s.t.} \quad & t \geq \frac{1}{r_i} \sum_{j=1}^k \beta_j^T \Omega_i \beta_j, \quad i = 1, \dots, n, \\ & \beta_j \geq 0, \quad j = 1, \dots, k. \end{aligned}$$

Due to the length, we omit the proof of this theorem in this paper. The detailed proof of the relevant problem is available in [27].

The clustering procedure (*step 2*) can be achieved by standard kernel K-means clustering proposed in [9]. If the objective function defined in (4.13) stops increasing, the iteration stops. The complexity of the adaptive K-means algorithm is determined by the QCQP problem where complexity is $O(pk^3n^3)$. Since the adaptive K-means algorithm is locally optimized, the performance is strongly dependent on the starting point. Practically, we use multiple starting points from all the individual kernel $\Omega_1, \dots, \Omega_p$ to obtain the initial partition $L_{(0)}$ and then run the overall algorithm from different initial partitions and select the best result with the maximum object function value. The overall complexity of the total algorithm is then $O(p^2k^3n^3)$. The regularization parameter λ of the covariance matrix is selected empirically, in our approach we set λ to 0.01. In some cases, it is necessary to regularize the weights of data sources to avoid overfitting. In our application, we also benchmark the regularization effect by setting different minimal boundaries of weights on data sources. However, the influence of the regularization on data sources was not significant and thus, we do not want to mention the regularization in the discussion of results.

4.2 Average Combination of Kernels Instead of the complicated approach of tuning the weights in kernel fusion, one can also combine the kernels averagely as,

$$\Omega = \sum_{i=1}^p \frac{1}{p} \Omega_i.$$

Regarding Ω as a new individual data which equally combines information of multiple data sources, one could apply standard clustering algorithms on this new combined data in kernel space. In this paper, we apply 6 standard clustering algorithms on the averagely combined kernel. Since these methods have been well studied in the literature, we omit the discussion of their formulations here.

4.2.1 Kernel K-means The kernel K-means algorithm applied on the average kernel can be regarded as a simplified version of the AKKC algorithm, which only contains the kernel K-means clustering step.

4.2.2 Hierarchical Clustering In order to apply hierarchical clustering methods in feature space, we first transform the kernel matrix into a distance matrix by calculating the distances between feature vectors [24],

$$(4.16) \quad \|\phi(x) - \phi(z)\|^2 = \langle \phi(x), \phi(x) \rangle - 2\langle \phi(x), \phi(z) \rangle + \langle \phi(z), \phi(z) \rangle.$$

Then we apply standard linkage clustering methods (single linkage, average linkage, complete linkage and ward linkage) on the transformed distance matrix and obtain the partitions by hierarchical clustering. In particular, the ward linkage clustering algorithm based on average combination of kernels can be regarded as a special case of the WLCDM method [13]. In WLCDM, the weights assigned on data sources are determined empirically, while in our paper, the weights are set as equal.

4.2.3 Spectral Clustering The spectral clustering algorithm we apply in this paper is proposed by Jordan and Weiss [14]. In our experiment, the Laplacian is constructed on the averaged kernel matrix.

5 Clustering Evaluation

The quality of clustering result is evaluated by different indices. These indices can be categorized as two groups: internal validation and external validation. In the context of hybrid clustering, we highlight the main differences as following. Internal validation usually requires two inputs: the clustering partitions obtained by the algorithm, and the original data set. Since internal validation is calculated on data set as a “goodness” of partitions, it is often data dependent. In other words,

internal validation can be affected by the data structure, the dimensionalities, and the scale of the data set. So it is often difficult to compare internal validations across heterogeneous data sets. On some data, such as gene sequence data, internal validation is also difficult to be computed directly. On the other hand, external validation compares the clustering partitions obtained by the algorithm with a reference partition (usually assumed as ground-truth labels), so it is independent of the structure, dimensionality and scale of the data source. For hybrid clustering, performing model prediction and comparison based on external validation is easier because it gives a unique score, while when using internal validations one has to consider the affect of data heterogeneities.

5.1 Internal Validations Being aware of the data dependency problem, we apply different internal validations for different data sets separately and only compare internal validations on the same data set.

Mean Silhouette Value (MSV) The Silhouette value of a clustered object (e.g., journals) measures its similarities with objects within the cluster versus the objects outside of the cluster [22]. The Silhouette value is defined as follows:

$$(5.17) \quad S(i) = \frac{\min(B(i, C_j) - W(i))}{\max[\min(B(i, C_j)), W(i)]},$$

where $W(i)$ is the average distance from object i to all other objects within its cluster, and $B(i, C_j)$ is the average distance from object i to all objects in another cluster C_j . The mean Silhouette value (MSV) for all objects is an intrinsic measurement of the overall quality of a clustering solution and it varies with the number of clusters, which can also be used to find the optimal cluster number. In this paper, we have two different data sources so correspondingly we need two MSV indices. The MSV calculated on text data is denoted as TMSV while the one calculated on link based citation data is called LMSV.

Modularity Modularity [20] is a graph based evaluation of clustering. Up to a multiplicative constant, modularity calculates the number of intra-cluster citations minus the expected number in an equivalent network with the same clusters but with citations given at random. Intuitively, in a good clustering there are more edges within (and fewer citations between) clusters than could be expected from random edges. The modularity Q_q of a clus-

tering into q partitions is defined as

$$(5.18) \quad Q_q = \sum_{s=1}^q \left(e_{ss} - a_s^2 \right),$$

where $a_s = \sum_{r=1}^q e_{rs}$. Here, e_{rs} is the fraction of edges between nodes in partition r and s . Obviously, the larger the distortion between internal edges e_{ss} and expected internal edges a_s^2 , the more modular the graph or network is (higher Q). By design, $Q < 1$.

5.2 External Validations

Normalized Mutual Information(NMI) Mutual information is a symmetric measure to quantify the statistical information shared between two distributions. Let $\{c_i\}_{i=1}^n$ and $\{l_i\}_{i=1}^n$ be the set of indicators and the ground truth labels, respectively. The normalized mutual information is defined as:

$$(5.19) \quad NMI = \frac{2 \times H(\{c_i\}, \{l_i\})}{H(\{c_i\})H(\{l_i\})},$$

where $H(\{c_i\}, \{l_i\})$ is the mutual information between $\{c_i\}_{i=1}^n$ and $\{l_i\}_{i=1}^n$, $H(\{c_i\})$ and $H(\{l_i\})$ are the entropy of indicators and labels. For a balanced clustering problem, if the indicators and the labels are independent, the mutual information approaches 0.

Rand Index, for the samples $\{x_i\}_{i=1}^n$, let the vectors $\{c_i\}_{i=1}^n$ (denoted as \mathcal{C}) and $\{l_i\}_{i=1}^n$ (denoted as \mathcal{P}) be the corresponding cluster indicators and ground truth labels, respectively. Consider a pair of vectors (c_i, l_i) . We refer to it as (1) a the number of pairs if both vectors belong to the same cluster in \mathcal{C} and to the same group in \mathcal{P} , (2) b the number of pairs if both vectors belong to different clusters in \mathcal{C} and to the different groups in \mathcal{P} . (3) c the number of pairs if the vectors belong to the same cluster in \mathcal{C} and to different groups in \mathcal{P} , and (4) d if the vectors belong to different clusters in \mathcal{C} and to the same group in \mathcal{P} . Rand Index is defined as [11]:

$$(5.20) \quad RI = \frac{a + b}{a + b + c + d}.$$

6 Dataset

6.1 Data Sources and Data Processing The main data set contains more than 6,000,000 publications (articles, letters, notes and reviews) indexed by the Web of Science (WoS) database of *ThomsonScientific* (Philadelphia,PA,USA) from the year 2002 till 2006. In

Field #	ESI Field
1.	Molecular Biology and Genetics
2.	Multidisciplinary
3.	Neuroscience
4.	Pharmacology Toxicology
5.	Physics
6.	Plant and Animal Science
7.	Psychology/Psychiatry

Table 1: 7 science categories of journals labeled by Essential Science Indicator(ESI)

the pre-processing step, the ambiguities in the spelling of the journal names, changes of the journal names, spelling of the author names, bibliographic data, and citations are resolved. We only keep the journals with more than 50 papers and more than 30 references or citations. After that pre-processing, we got 8,305 journals, spanning 22 field categories according to the ESI classification [7]. From these 22 categories, we selected 7 categories (1869 journals) randomly as the journal set data used in this paper.

6.2 Text Mining Analysis The titles, abstracts and keywords of these 1869 journals are indexed by a text mining program using Jakarta Lucene API without controlled vocabulary. The index result contains 9,473,061 terms and we cut the Zipf curve of terms at the head and the tail to remove the rare terms, stop-words and common words. After Zipf cut, 669,860 meaningful terms are used to represent the journal in the vector space model (text data) and the weights of terms are calculated by TF-IDF weighting scheme.

6.3 Citation Analysis We only considered citations between papers from 2002 till 2006 and aggregated all paper-level citations into journal-by-journal citations. The direction of citations is ignored and symmetric citation data is obtained.

6.4 Labels of Standard Categories We referred the Essential Science Indicators (ESI) classification created by Thomson Scientific as the ground-truth labels of journal assignment[7]. The ESI labels are also used in the calculation of external validations. The 7 ESI labels of 1869 types of journals in our dataset are presented in Table 1.

7 Experimental Results

7.1 Clustering by Fusing 2 Datasets: Text Mining and Bibliometrics Data We first clustered the

text mining data and bibliometrics data separately in their original dimensions. Then we applied ensemble clustering methods to combine the partitions obtained from text and bibliometrics data into a consensus partition. We also took the kernel fusion approach by mapping the data into kernel space and applied adaptive kernel fusion and average fusion clustering algorithms to obtain the partition. We first fixed the number of clustering to 7, which is the same number as the categories defined by ESI labels. The clustering results were evaluated by 5 different evaluations and compared across different hybrid clustering strategies in Figure 2.

When using single data set for clustering, text mining data provides more accurate journal partitions than citation data. The RI and NMI scores of text data (0.8618, 0.6627) are much higher than those of citation data (0.7383, 0.4844). Ensemble clustering algorithms do not perform well when combining 2 partitions. After hybrid approach, their RI and NMI scores are compromised between the individual performance of text data and citation data. On the contrary, kernel fusion methods obtain satisfying performance, where hybrid approach performs as same as the best performance obtained on individual data set. For the proposed algorithm (AKKC), the mean values of weights learned in 50 random repetitions are: 0.6139 on text data and 0.3861 on citation data. The adaptive algorithms automatically bias towards text dataset (the “useful” data or “relevant” data). Other kernel fusion results based on averagely combined kernel are also good, this is probably because the kernel created on text data is dense while the kernel constructed on the citation data is very sparse. When these two kernels are averagely combined, the effect of the sparse kernel matrix (citation data which has low performance) is overwhelmed by the dense kernel matrix (text data which has good performance).

According to the results of three internal validations (TMSV is applied on text data, LMSV and MOD are applied on citation data), the trend is consistent with external validations. Ensemble clustering methods do not perform well since their TMSV scores are even lower than text data alone. Kernel fusion methods again get satisfying results: the TMSV obtained by AKCC partition is 0.1974, which is almost the same as text data alone. In particular, spectral clustering applied on averagely combined kernel gets the highest TMSV score (0.2082). The results on citation data show the same picture, especially, the LMSV and MOD indices are significantly improved by hybrid clustering.

7.2 Clustering by Fusing 4 Data sets: Text Mining and Bibliometrics Data and their Pro-

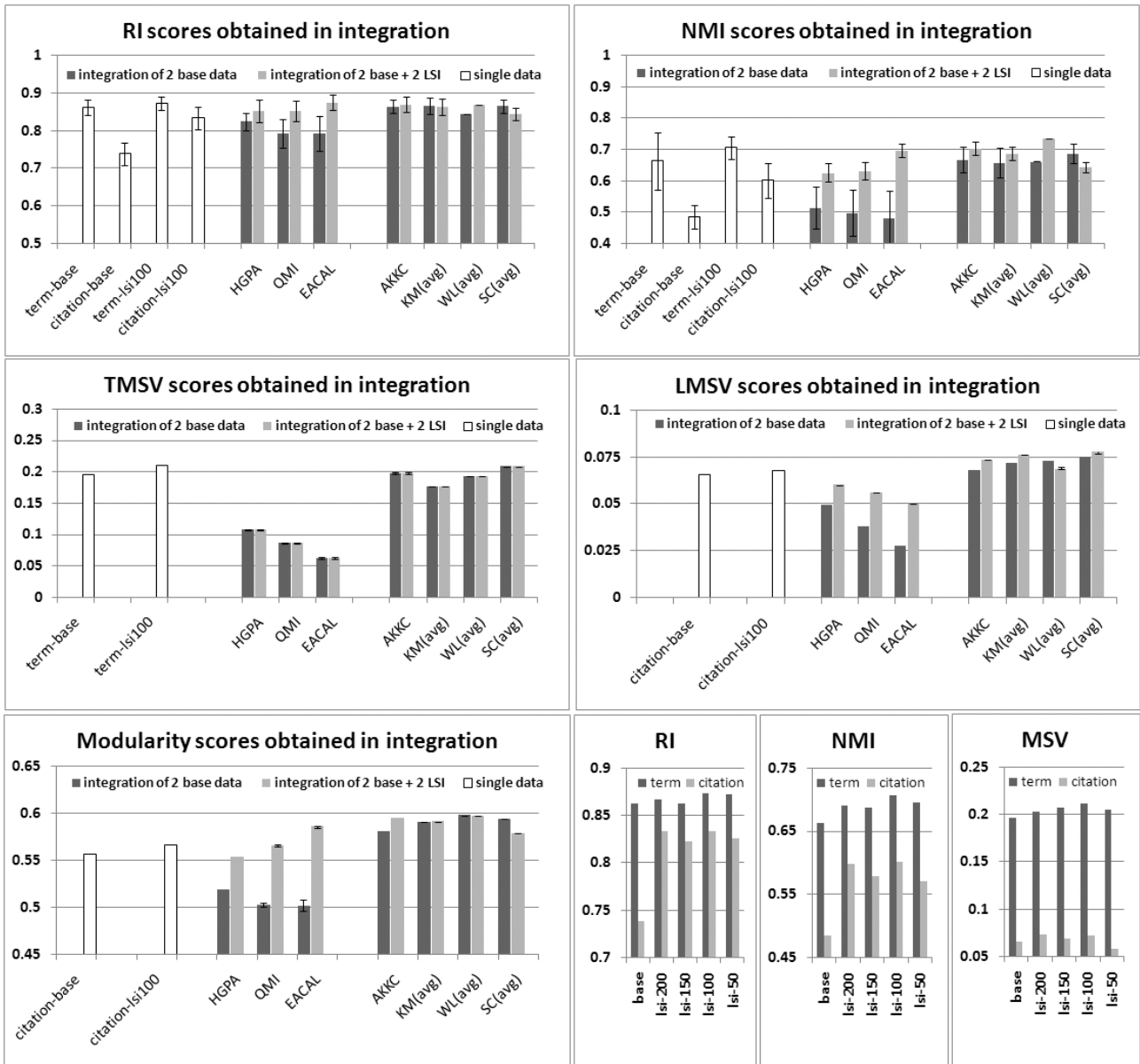


Figure 2: Comparison of 2 external validations and 3 internal validations by hybrid clustering. In the figure, *term-base* and *citation-based* represent term and citation data without dimensionality reduction. *term-lsi100* and *citation-lsi100* represent the projection of data in 100-dimensional space obtained by LSI. The white bars show results of individual data, the dark grey bars show results obtained by integration of 2 base data sources (term-base and link-base), the light grey bars show results obtained by integration of 4 data sources. For clarification, only 3 best ensemble clustering approaches and 4 best kernel fusion approaches are shown. KM(avg), WL(avg) and SC(avg) represent the kernel K-means, Ward linkage and spectral clustering methods applied on average kernel respectively. The right bottom figure shows RI, NMI, and MSV indices on clustering in different dimensions by LSI.

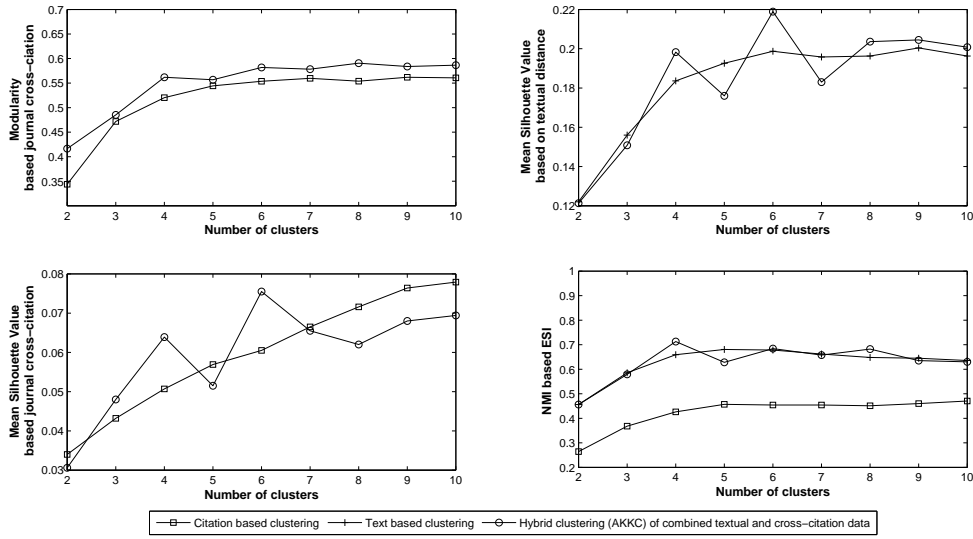


Figure 3: Clustering comparison across various numbers of clusters

jections after Dimensionality Reduction We notice the fact that ensemble clustering methods do not perform well when combining 2 datasets for clustering. This is probably because ensemble clustering was originally proposed to combine various partitions derived from one data set. So it expects and relies on the “agreement” among various partitions to find the optimal consensus partition. In previous experiment, according to evaluations, one data set is relevant and another one is comparably less relevant, so the insufficient number and inconsistency of partitions probably prevent ensemble clustering approach to find the optimal partition. In this experiment 2 other new data sets were employed. The new data was obtained by applying latent semantic indexing (LSI) [6] on text and citation data. We traced the eigenvalues during dimensionality reduction and found in the 100-dimensional space spanned by principal components, about 95% of the variance contained in text and citation data is preserved. So, we reduced text and citation data to the 100-dimensional space and constructed 2 new data sets (text-LSI100 and citation-LSI100). All 4 data sets were then combined for hybrid clustering. We notice that in literature, the empirical optimal number of partitions for ensemble clustering is around 20, however, in order to keep the problem concise and clear, in this paper we used 4 data sets to address the problem.

By dimensionality reduction, the performance of citation data is significantly improved (RI score increases from 0.7383 to 0.8332, NMI score increases from 0.4844 to 0.6013). There is also considerable improvement on

text data but it is not significant. The important discovery is, when combining these 4 data sets for clustering, the performances obtained by ensemble clustering are significantly improved. For example, EACAL algorithm obtains RI score of 0.8746 and NMI score of 0.6969, which is much better than the results obtained by combining 2 data sets (RI 0.7923, NMI 0.4811). This situation is probably because among the 4 data sets combined, 3 of them have good quality (text, text-LSI100 and citation-LSI100), so that ensemble clustering algorithms are able to find “agreement” among individual partitions and obtain stable consensus partitions. On the comparison, kernel fusion methods do not affect much by the 2 new data sets and their results are almost same with previous experiments. The weights learned by proposed algorithm on 4 data sets are: 0 on text, 0.5080 on text-LSI100, 0 on citation, 0.4920 on citation-LSI100.

The internal validation results are also consistent with external ones. Ensemble clustering methods get significantly improved on citation data: for instance, on EACAL, LMSV score increases from 0.0276 to 0.0501 and MOD score increases from 0.5016 to 0.5854.

7.3 Comparison of Performance across Various Number of Clusters

We also benchmark the optimal number of clusters by different validations. Figure 3 presents the evolution of 4 different indices with the number of clusters. We first consider the indices obtained by single data clustering. Figure 3.a plots the modularity index applied on citation data and it clearly

Cluster1(ESI #4)	Cluster 2(ESI #7)
1. J. Pharma.& Exp. Therap.	1. J. Person. & Soc. Psych.
2. E. J. Pharma.	2. Perso.& Soc. Psych. Bull.
3. I. J. Pharma.	3. Psych. Sci.
4. B. J. Pharma.	4. Psych. B. R.
5. Pharam. Research	5. Memory & Cognition
6. M. Pharma.	6. Psych. B.
Cluster 3(ESI #1)	Cluster 4(ESI #3)
1. Hydrobiologia	1. PNAS
2. J. Fish Biology	2. J. Neuroscience
3. Theriogenology	3. Nature
4. M. Biology	4. Science
5. Limnology & Oceanography	5. Neuron
6. J. E. M. Biology & Ecology	6. E. J. Neuroscience
Cluster 5 (ESI #6)	Cluster 6(ESI #5)
1. Plant Physiology	1. Physical Rev. L.
2. Plant Cell	2. Physical Rev. B
3. Plant Journal	3. Physical Rev. D.
4. J. E. Botany	4. A. Physics L. B
5. Plant M. B.	5. Physical R. E
6. Planta	6. Physical R. A
Cluster 7(ESI #7)	
1. A. J. Psychia.	
2. J. Clinical Psychia.	
3. Bio. Psychia.	
4. A. G. Psychia.	
5. B. J. Psychia.	
6. Biplor Disorders	

Table 2: Partitions of journals obtained by AKKC. In each cluster, six most important journals are shown as the example members of that cluster. The mappings of cluster numbers to ESI label numbers are obtained by manual interpretation.

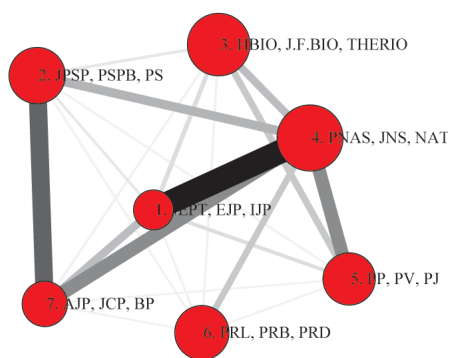


Figure 4: Visualization of journal sets structural mapping. Big vertex represents for large number of journals. Thick edge represents for large number of citations. The clustering result in this figure is obtained by AKKC.

shows that the index becomes stable when the cluster number is larger than 6. The TMSV index plotted in Figure 3.b also suggests an optimal cluster number between 6 to 8. The LMSV index plotted in Figure 3.c cannot give any clue about the optimal cluster number because it grows monotonically from 2 to 10. We also benchmark the partitions of hybrid clustering obtained by AKKC algorithm over various cluster numbers. In modularity index, it also suggests the optimal number as 6 to 8. Especially, the index value obtained by hybrid clustering is always higher than single data clustering, which means the quality of clustering obtained by hybrid clustering is better. The TMSV index and LMSV index obtained by AKKC based on hybrid clustering also indicate that the optimal clustering number should be around 6. Though the optimal cluster number suggested by data does not exactly match with the number of ESI fields, it is still acceptable because the ESI labels as given in Table 1 contains “multidisciplinary”, which may be quite similar to some other journal sets at data level. Also, “Molecular Biology and Genetics” has strong relation with “Plant and Animal Science” and can also be merged as one category.

We also use the standard ESI labels to evaluate the NMI score of clustering labels across cluster numbers and the results are shown in Figure 3.d. The NMI score of hybrid clustering partition is always better than citation data only, and quite close to the performance of text data. This is consistent to our previous results obtained by fixing the cluster number as 7.

7.4 Mapping of Journal Sets We visualize the hybrid clustering results of journal sets obtained by AKKC in Pajek [3]. The visualization of structure mapping of 1869 journals (7 journal sets) is shown in Figure 4. Without hybrid clustering, text data and citation data may generate different networks while in our approach they are completely combined to obtain a consolidate partition. Table 2 provides a list of the 6 most important journals within each cluster. The importance of journals is determined by measuring the journal cross-citations within each cluster. By comparing the ESI fields in Table 1 with the content of journal clusters in Table 2, most of the clusters can be matched to the corresponding ESI fields. Furthermore, we manually interpret and assign the most appropriate ESI field after each cluster number as shown in Table 2. Then by interpreting the network structure of journal clusters, we can easily have a high level view about the importance and relationships of journal sets.

8 Discussion

An open question of combining heterogeneous data sources for clustering analysis is to determine the “relevant” or “useful” data source w.r.t. to the problem. Furthermore, if an algorithm is capable of finding “relevant” or “useful” data sources, maybe we can expect a lower bound about the performance of data fusion approach which should be not worse than the best individual data source. In supervised learning, the “relevance” or “usefulness” can be determined by validations. However, for unsupervised learning such as clustering, it is difficult to split the data for training and validation and the whole data set should be isolated from the label information, so extending model prediction techniques of clustering analysis to multiple data sources, is a difficult and ongoing problem.

In this paper, we extend the strategy of ensemble clustering to multiple data sources. However, we should be aware of multiple caveats when applying them. If the number of data sources is insufficient, in order to obtain variants of partitions for ensemble method to find consolidate partition, we can also apply multiple distance measures, different subsets of features or various dimensionality reduction techniques on each individual data source to generate more partitions. For ensemble clustering methods, it is also possible to use different clustering algorithms to generate partitions for combination. However, it would be hard to explain the combinatorial affect of algorithm heterogeneities with data heterogeneities, so we suggest to apply the same clustering algorithms here.

For kernel based data fusion approach, there is no suggested minimum number of data sources (or partitions) to be combined. As a matter of fact, it works quite well with 2 data sets in our problem. However, the assumption of kernel K-means based clustering is that the data is normally distributed in kernel space. The advantage of kernel methods is that by kernel mapping, non-Gaussian data can be transformed into Gaussian data in kernel space. So, the performance of kernel based clustering is also determined by the choice of kernel function and kernel parameters. Therefore, in kernel based data fusion, we should also consider the combinatorial affect of kernel function (parameters) and data heterogeneities. In this paper, since the focus is combining heterogeneous data rather than tuning optimal kernel parameter, we only use linear kernel function to construct the kernels. In other words, all our results are obtained by combining data in linear space. The issue of combining heterogeneous data in nonlinear space is a very interesting problem and it will be the main topic of our future research.

In this paper we compared the scores based on

internal validations (mean Silhouette value, modularity) across different cluster numbers in order to find the optimal cluster number. Moreover, we presented a benchmark of cluster number on two datasets and they show consistent trends about the optimal cluster number. However, finding the optimal cluster number in hybrid clustering is a difficult problem because the trend of validation indices may behave differently across data sources, thus when fusing a large number of data sources, the interpretation of optimal number might be hard. In that case, one might need to find the “agreement” among multiple indices.

9 Conclusion

The main contribution of this paper can be concluded as following.

First, we provided a framework of hybrid clustering methods to combine text mining and bibliometrics data for journal sets analysis. This framework can be generalized to other heterogeneous data sets for clustering analysis as well, for example, to combine multi-model digit recognition datasets [28, 29], to combine various genomic data for multi-view gene clustering [29, 30], etc. In this framework, we reviewed and extended the methodologies of ensemble clustering to hybrid clustering problems. We also proposed kernel fusion methods for hybrid clustering in a unified view.

Second, to solve the main obstacle in hybrid clustering, we highlighted the problem of how to automatically determine the “relevance” or “usefulness” among data sources. We proposed a novel AKKC algorithm to learn optimal weights of data sources together with the clustering procedure. This algorithm extends the optimal kernel learning approach from supervised learning context to unsupervised learning context, in particular, with an application of heterogeneous data fusion.

We applied hybrid clustering to combine text mining and bibliometrics for journal sets clustering. According to our experimental results, the performance obtained by hybrid clustering is better than that the best single data.

Based on the consistent partition obtained by hybrid clustering, we visualized a network of journal sets containing fields mapping and citation links.

10 Acknowledgement

Research supported by China Scholarship Council(CSC) and by grants and projects for the Research Council K.U.Leuven (GOA-Mefisto 666, GOA-Ambiorics, several PhD / Postdocs & fellow grants), the Flemish Government FWO: PhD / Postdocs grants, projects G.0240.99, G.0211.05, G.0407.02, G.0197.02, G.0080.01, G.0141.03, G.0491.03, G.0120.03,

G.0452.04, G. 0499.04, G.0226.06, G.0302.07, ICCoS, ANMMM; AWI;IWT:PhD grants, GBOU (McKnow) Soft4s, the Belgian Federal Government (Belgian Federal Science Policy Office: IUAP V-22; PODO-II (CP/01/40), the EU(FP5-Quprodix, ERNSI, Eureka 2063-Impact;Eureka 2419-FLiTE) and Contracts Research/Agreements (ISMC/IPCOS, Data4s, TML, Elia, LMS, IPCOS, Mastercard). Bart De Moor is a full professor at the K.U.Leuven, Belgium. The scientific responsibility is assumed by its authors.

References

- [1] H. G. Ayad, M.S. Kamel. *Cumulative Voting Consensus Method for Partitions with a Variable Number of Clusters*. IEEE Trans. PAMI, 30, 2008, pp. 160-173.
- [2] R. R. Braam, H. F. Moed, and A. F. J. van Raan. *Mapping of science by combined co-citation and word analysis*. Journal of the American Society for Information Science, 42(4),1991, pp. 252266.
- [3] V. Batagelj, A. Mrvar. *Pajek analysis and visualization of large networks*. Graph Drawing, 2265, 2002, pp. 477478, ISSN: 0302-9743.
- [4] J. Chen, Z. Zhao, J. Ye, and H. Liu. *Nonlinear Adaptive Distance Metric Learning for Clustering*. ACM KDD, 2007.
- [5] I. Csiszar, and G. Tusnady. *Information geometry and alternating minimization procedures*, Statistics and Decisions, Supplementary Issue 1, 205-237, 1984.
- [6] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R. Harshman. *Indexing by latent semantic analysis*. JASIS, 41, 6(1990),pp.391407.
- [7] ESI. *Essential Science Indicators (accessible via: <http://www.esi-topics.com/fields/index.html>)*.
- [8] A.L.N. Fred,A.K. Jain. *Combining Multiple Clusterings Using Evidence Accumulation*. IEEE Trans. PAMI, 27, 2005, pp. 835-850.
- [9] M. Girolami. *Mercer Kernel-Based Clustering in Feature Space*. IEEE Tran. NN., 13(3), 2002.
- [10] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. *On clustering validation techniques*. Journal of Intelligent Information Systems, 17(2-3),2001, pp.107145.
- [11] A.K. Jain and R.C. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., 1988.
- [12] F. Janssens. *Clustering of scientific fields by integrating text mining and bibliometrics*. PhD thesis, Faculty of Engineering, K.U.Leuven (Leuven, Belgium), May 2007, 261 p.
- [13] F. Janssens, W. Glanzel, B. De Moor. *A hybrid mapping of information science*. Scientometrics, 3 (2008), pp. 607-631.
- [14] A.Ng,M. Jordan, and Y. Weiss. *On spectral clustering: Analysis and an algorithm*. NIPs, 14, 2002.
- [15] F. Janssens, L. Zhang, W. Glanzel. *Hybrid Clustering for Validation and Improvement of Subject-Classification Schemes*. Internal Report 08-141, 2008, ESAT-SISTA, K.U.Leuven, Leuven, Belgium.
- [16] R. N. Kostoff, H. A. Buchtel, J. Andrews, and K. M. Pfeil. *The hidden structure of neuropsychology: Text mining of the journal Cortex: 1991-2001*. Cortex, 41(2),2005, pp. 103115.
- [17] G. Lanckriet, N. Cristianini, P. Bartlett,L.E. Ghaoui, and M.I. Jordan. *Learning the Kernel Matrix with Semidefinite Programming*. J. Mach. Learn. Res., 5, 27-72, 2004.
- [18] B. Mirkin. *Reinterpreting the Category Utility Function*. Machine Learning, 45(2), 219-228, 2001.
- [19] D. S. Modha and W. S. Spangler. *Clustering hypertext with applications to web searching*. Proceedings of the 7th ACM on Hypertext and Hypermedia, 2000, pp. 143152.
- [20] M.E.J. Newman. *Modularity and community structure in networks*. PNAS US, 103(23), 2006, ISSN: 0027-8424.
- [21] V. Plachouras. *Dempster-Shafer Theory for a Query-Biased Combination of Evidence on the Web*. Information Retrieval, 8,2005,pp. 197-218.
- [22] P. Rousseeuw,*Silhouettes: a graphical aid to the interpretation and validation of cluster analysis*. Journal of Computational and Applied Mathematics, 20(1),1987, pp.5365.
- [23] A. Strehl, J. Ghosh. *Cluster Ensembles-A Knowledge Reuse Framework for Combining Multiple Partitions*. JMLR,3(2002),pp. 583-617.
- [24] J.S. Taylor, and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, Cambridge, England, 2004.
- [25] A. Topchy, A.K. Jain,W. Punch. *Clustering Ensembles: Models of Consensus and Weak Partitions*. IEEE Trans. PAMI,27(12),2005, pp. 1866-1881..
- [26] R. Xu,and D. Wunsch II. *Survey of Clustering Algorithms*. IEEE TRAN. on Neural Networks, 16(3), 645-678, 2005.
- [27] J. Ye, S. Ji, and J. Chen. *Multi-class Discriminant Kernel Learning via Convex Programming*. J. Mach. Learn. Res., 9, 2008, 719-758.
- [28] S. Yu, B. De Moor, Y. Moreau. *Clustering by heterogeneous data fusion: framework and applications*. NIPs workshop on Learning with Multiple Sources, Whistler, Canada, 2008.
- [29] S. Yu, L.C. Tranchevent, X.H. Liu, B. De Moor, Y. Moreau. *Integrating heterogeneous data sets for clustering*. Internal Report, SCD-SISTA, ESAT, K.U.Leuven, submitted to IEEE Trans. on Pattern Analysis and Machine Intelligence, 2008.
- [30] S. Yu, L.C. Tranchevent, B. De Moor, Y. Moreau. *Gene prioritization and clustering by multi-view text mining*. Internal Report, SCD-SISTA, ESAT, K.U.Leuven, submitted to Bioinformatics.
- [31] B. Zhang, Y. Chen, W. Fan, E. A. Fox, M. A. Goncalves, M. Cristo, and P. Calado. *Intelligent fusion of structural and citation-based evidence for text classification*. Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, 2005, pp. 667-668.