Clustering and Identifying Temporal Trends in Document Databases

Alexandrin Popescul[1,2], Gary William Flake[2], Steve Lawrence[2], Lyle H. Ungar[1], C. Lee Giles[2]

[1] Department of Computer and Information Sciences, University of Pennsylvania, PA
[2] NEC Research Institute, 4 Independence Way, Princeton, NJ 08540

{sasha,flake,lawrence,giles}@research.nj.nec.com
{popescul,ungar}@unagi.cis.upenn.edu

Abstract

We introduce a simple and efficient method for clustering and identifying temporal trends in hyper-linked document databases. Our method can scale to large datasets because it exploits the underlying regularity often found in hyper-linked document databases. Because of this scalability, we can use our method to study the temporal trends of individual clusters in a statistically meaningful manner. As an example of our approach, we give a summary of the temporal trends found in a scientific literature database with thousands of documents.

1 Introduction

Over the past decade, the World Wide Web has become an increasingly popular medium for publishing scientific literature. Since many researchers release preprints on the web, the scientific literature on the web is often far more timely than a similar snapshot of paper journals and proceedings, especially when one considers review and publication delays. As such, the scientific literature on the web may represent one of the more up-to-date characterizations of the state of a scientific discipline.

While the web is rich with information about the progress of science, gathering and making sense of this data is difficult because publications on the web are largely unorganized, they are not indexed as many paper publications are, citation and impact counts are not readily available, and differences in language and terminology make text-based approaches problematic.

In this paper we consider clustering applied to online scientific literature. Clustering scientific literature is an important problem because it enables tasks such as estimating the amount of activity, growth, and decay in different scientific areas, identifying the fragmentation or merging of disciplines, and assisting a user in navigating through a database.

Our approach to clustering uses the citation patterns of a database to form soft clusters about the most frequently cited papers. The soft clusters, in turn, can be compared to one another in terms of the papers that they have in common. Similar soft clusters are merged by a secondary clustering algorithm. In the end, we find the collections of documents that are all related to one another by their citation patterns, with the cluster centroid being the most often cited papers in the cluster that were centroids in the first phase clustering. By approaching the problem in this manner, we can rapidly calculate clusters for datasets with tens of thousands of documents. Because our dataset is relatively large, we are able to measure relative growth trends within clusters in a statistically meaningful manner.

This paper is organized into six sections. In Section 2, we discuss previous work related to citation analysis and give reasons why other approaches are inadequate for our task. Section 3 describes CiteSeer, the largest database of full-text scientific literature that is freely available, which we used for our study. In Section 4, we describe our clustering algorithm with an emphasis on how the computational complexity of the algorithm is reduced by exploiting the regularity in citation patterns. Section 5 contains a summary of the results obtained by running the proposed clustering algorithm on the CiteSeer database. Finally, Section 6 gives our conclusions and discusses future work.

2 Previous Work

Co-citation analysis has been used extensively to map and analyze scientific disciplines since the introduction of the first such systematic computerized method by Small and Griffith [13]. These systems gave insight into the structure of disciplines and the interrelationships among them. The data for the study described in [13] was from the first quarter of the 1972 Science Citation Index (SCI), published by the Institute for Scientific Information, including citation information for scientific papers from the physical, biological and medical literature. Documents that received less than 10 citations were filtered out, resulting in a total of 1,832 documents.

The co-citation count is a similarity measure between two documents based on how many other documents cite the two documents, i.e. in how many reference lists the two co-occur. In [13], co-citation counts were computed for each pair of documents, and single linkage clustering was performed. The study tested and provided support for two hypotheses: disciplines of science exhibit structure and such structure can be discovered by employing co-citation analysis. Co-citation analysis provides advantages over bibliographic coupling (a measure based on the number of documents co-referenced by the documents for which the measure is computed). For example, bibliographic coupling cannot identify related documents based on a given cited article if one of the related documents was written prior to the date of the cited article (and hence the candidate cited article was unavailable for citing no matter how relevant). Co-citation information becomes richer over time as more papers are published that cite given documents.

Garfield [2] describes the work behind the creation of SCI and additional co-citation studies revealing the structure of scientific disciplines.

McCain [8] uses authors rather than documents as a unit of study, selecting a set of 58 authors from the field of population genetics which are analyzed using clustering with a correlation matrix derived from co-citation counts, and multidimensional scaling to visualize the results. Both Ward's clustering [14] and complete linkage hierarchical clustering methods were used. The study noted a problem where authors with more recent publications are discriminated against because the filtering phase excludes all documents having citation counts below a fixed threshold, irrespective of how long it has been since the document was published and available for citation.

Chen and Carr [1] used ACM publication data to analyze the structure of hypertext literature, filtering out authors that were cited less than five times during the period of 1987 – 1998, resulting in 367 authors. An author co-citation matrix was constructed and converted into a correlation matrix. Principal Component Analysis (PCA) was used to extract factors to be plotted for three sub-periods separately and for the entire period. Visualization methods employed the use of colors that identify the age of corresponding papers thus allowing identification of emerging research directions. The authors analyzed were ranked according to their loadings to the factors produced by PCA (factors accounting for most of the variance were considered).

Raghupathi and Nerur [10] analyzed 155 authors in the field of artificial intelligence with data extracted from the Science Citation Index for the period of 1980 – 1995. They used a similar technique to [1], finding 14 factors that were labelled manually.

Pitkow and Pirolli [9] used the method of [13] applied towards sets of hypertext documents on the World Wide Web, transferring the concept of scientific publication citations to hypertext links on the web. 5,582 HTML and 15,139 non-HTML documents were considered and clustered using complete linkage hierarchical clustering at different citation frequency thresholds (one, three, five and ten).

In comparison with these studies, the method we introduce here facilitates the analysis of much larger datasets (the dataset we analyze has close to an order of magnitude more documents than the largest dataset from these studies), and addresses the issue of discriminating against newer publications.

3 CiteSeer Data

We use the database of scientific literature created by CiteSeer [5, 6], which is available at `http://csindex.com/`. CiteSeer is currently the largest free full-text index of scientific literature in the world, indexing over 250,000 articles focusing on computer science. The CiteSeer software is available at no cost for non-commercial use.

CiteSeer indexes articles found on the publicly indexable web, on the homepages of researchers or on institutional technical report archives, and thus our analysis relates to the body of computer science literature that is available online, which is likely to differ from all computer science literature.

CiteSeer differs from many online scientific literature archives in that it performs Autonomous Citation Indexing (ACI), autonomously extracting and matching references in bibliographies. As a result, CiteSeer contains an implicit graph where papers are represented by vertices and directed edges represent citations between papers. In this way, relationships between papers can be identified independently of the textual content of the papers.

For the results reported in this paper we started with 150,000 documents from CiteSeer and then narrowed

the dataset to highly cited papers and papers that were co-cited with highly cited ones. The final dataset analyzed includes 31,428 papers in total.

Because we view the CiteSeer database as a graph, the task of clustering is closely related to the graph partitioning and k-centroids problems, both of which are NP-hard in the most general case. Given the size of the dataset, and the difficulty of graph clustering, it is essential to use an efficient method such as the approach that we discuss in the next section.

4 Efficient Graph Clustering

In the general case, graph clustering is a time consuming process because of the temptation to perform the clustering in a way that requires similarities to be calculated for all vertices in a graph. The citations in scientific literature are, however, far from random and very non-uniform. As such, we make the reasonable assumption that scientific disciplines form about influential papers, and that citations to key papers are indicative of the community in which a paper should be naturally classified. Thus, our approach is to reduce the dimensionality of the problem by first identifying key papers that are cited above some threshold.

One problem with using the raw citation count as a measure of importance is that the older a paper is, the heavier the bias for the paper simply because there has been more opportunity for the paper to have been cited. We account for this fact by using a normalized citation count for each paper, where the normalization factor is equal to the number of papers in our database that have been published since the paper under consideration has been published. We also note that our database is more up-to-date than traditional databases of scientific literature since it includes conference papers and technical reports that may have only been made available on the web very recently. In this way, new influential papers can be upwardly adjusted while older papers with fewer citations over time are downwardly adjusted.

The next step of our algorithm is to create a soft cluster around each influential paper (where an "influential paper" is a paper that has a normalized citation count in excess of the threshold). Other papers are assigned to the soft cluster if they are co-cited along with the influential paper. Intuitively, the soft clusters contain any paper that any author deemed related to the seed paper.

After creating the soft clusters, we then calculate a similarity measure between the clusters. The similarity measure we use between two soft clusters, A and B, is

$$\frac{|A \cap B|}{|A| + |B| - |A \cap B|}.$$

Thus, in set terminology, the similarity is defined by the number of elements in common divided by the number of disjoint elements.

Once a similarity matrix is calculated with respect to the soft clusters, a traditional clustering algorithm is used on the reduced dataset to cluster the papers into an even smaller set. Readers may wish to consult Table 1 which contains the complete algorithm.

For our experiments, we used a threshold of 100 citations to decide if a paper should serve as a soft cluster centroid. This value was chosen *ad hoc* and we found that other values gave similar results. This choice produced 475 soft cluster centroids. Once the soft clusters were chosen, we then assigned all other papers in the dataset to one or more soft clusters, as shown in lines 12–15 of our algorithm. Thus, the citation graph is reduced from a 150,000 squared adjacency matrix to a 475 by 150,000 soft cluster matrix. We further reduce the problem by calculating the similarity measure between all soft clusters, which results in a 475 by 475 symmetric matrix.

At this point, the 475 by 475 matrix essentially contains similarity information of the research threads around the most influential papers in the dataset. We argue that this form of dimensionality reduction retains much of the information of how the most influential papers relate to one another and, as a result, contains most of the information for how scientific disciplines are organized. Moreover, the similarity measure implicitly contains information about rarely cited papers, since any paper that co-cites two influential papers will make the corresponding soft clusters have a higher degree of similarity.

Regarding the final step of the algorithm, any standard clustering algorithm can be used. We chose Ward's hierarchical clustering [14, 3] as it seems, based on dendrogram inspection, to better represent clusters for this data set, and does not exhibit the problem of singletons as much as single linkage does.

Ward's clustering is an agglomerative hierarchical clustering technique that tends to locate compact and spherical clusters. It is one of the variance minimization techniques, such as k-means [7]. While k-means requires the desired number of clusters to be specified in advance, Ward's technique allows the posterior choice of the desired level of cluster generality. A "successful" cut-off-level to cut the hierarchy can be chosen by visually inspecting the hierarchy dendrogram.

We propose a new way to naturally decide the cut level for the hierarchy; the cut-off level is determined by examining the "heights of merges" plot (see [3]) where we can see that the merge levels grow very slowly until a certain point where the levels start growing very fast, which indicates a good point to cut. For our experiments the cut at this point produced 15 clusters. One of these

```
1  procedure EFFICIENT-GRAPH-CLUSTER(graph: G = (V, E))
2      { Find most highly cited papers normalized by publications for year. }
3      C ← {}
4      for all {v ∈ V} do
5          norm ← number of papers published after v
6          norm-cite(v) ← |{(u, v) ∈ E}|/norm
7          if norm-cite(v) > threshold then
8              C ← C ∪ {v}
9          end if
10     end for
11
12     { Assign papers to soft cluster if they are co−cited. }
13     for all {v ∈ C} do
14         S_v ← {x : ∃_y (y, x) ∈ E ∧ (y, v) ∈ E}
15     end for
16
17     { Calculate similarity measure for all pairs c ∈ C . }
18     for all {x ∈ C} do
19         for all {y ∈ C} do
20             M_xy ← |S_x ∩ S_y|/(|S_x| + |S_y| − |S_x ∩ S_y|)
21         end for
22     end for
23
24     { Cluster reduced similarity matrix. }
25     CLUSTER(M)
26 end procedure
```

Table 1. The Efficient Graph Clustering Algorithm.

clusters was large and appears to contain a combination of topics, while the remaining clusters appear to cover well-defined topics.

The titles of the papers in each of 15 clusters are stemmed and the highest frequency words are used to characterize the clusters. Stop words are excluded.

5 Regression Analysis and Experimental Results

After clustering is complete, we perform linear regression analysis on the frequency for each year of publication for the papers in each cluster, in order to analyze the rate of growth for each cluster.

Most of the 15 clusters obtained have small p-values for the regression fits, which indicate that the general trend of the clusters is not flat over time. (The p-value represents the probability that the slope coefficient of a fitted model is equal to zero, i.e., when there is no trend in a corresponding cluster relative to the growth of all of the dataset). Table 2 summarizes the sizes, regression results and aggregate indegree (number of citations) information for the clusters, while Figures 1 and 2 show the actual regression fits.

Clusters 10, 11, 12, and 14 have relatively large p-values due to the data points lying almost parallel to the x-axis, i.e. we can conclude that the slope coefficient is zero and there is no trend relative to the entire collection of the documents. The linearity assumption of linear regression is largely violated in clusters 1 and 2 due to the clusters' small sizes, but the trends can be still identified by visualizing the graphs.

Figure 3 shows an MDS (multidimensional scaling) plot of the clusters along with the most frequent terms in each cluster. MDS [4] creates a visualization that aims to display points in a lower dimensional space so that the proximity of points in the resulting space reflect as closely as possible the proximity of points in the original space.

Table 3 shows the top cited papers that were centroids in the first clustering phase.

Regression analysis was run with the normalized frequencies of the years. The normalization factor was computed based on all 150,000 documents, therefore the growth and decline coefficients are indicative of growth and decline relative to the whole data set, and not only to the papers that were chosen to participate in the experiment (the papers chosen are those that were co-cited at least once with the 475 most cited (adjusted) papers; there are 31,428 such papers). As a result, most of the

#	Size	avAdjIn	avAbsIn	RegSlope (10^{-6})	Interc (10^{-6})	p-value
1	17	47.5	5.2	10.1	- 25.48	0.021
2	43	120.8	38.1	4.92	0.84	0.055
3	277	42.7	16.8	2.11	9.84	0.006
4	309	34.1	11.3	3.62	2.14	0.000
5	1079	46.9	18.7	2.36	9.32	0.002
6	268	121.5	24.8	2.27	8.40	0.000
7	374	58.1	18.1	4.70	- 1.20	0.001
8	1323	38.1	15.7	0.90	14.0	0.042
9	1954	29.3	15.0	- 4.52	46.7	0.001
10	2810	29.4	11.4	0.28	18.5	0.131
11	2631	32.3	13.5	- 0.30	22.8	0.393
12	2968	42.5	12.7	- 0.24	21.8	0.153
13	5693	36.0	13.0	- 1.29	27.8	0.001
14	18841	26.8	9.7	- 0.61	24.6	0.087
15	5181	23.5	11.3	- 3.04	36.1	0.000

Table 2. Cluster summaries. avAdjIn is the average adjusted indegree of the cluster (average number of citations to papers in the cluster), avAbsIn is the average absolute indegree of the cluster, RegSlope is the coefficient of the Year in the linear regression model, and Interc is the intercept in the linear regression model.

clusters show relative growth, and just 2 clusters exhibit significant declines—clusters 9 and 15.

The decline of cluster 9 confirms previous prototype analysis based on another clustering technique; this cluster's theme is "languages, compilers, garbage collection".

The most rapidly growing cluster is cluster 7 (among significant fits). Cluster 7 contains 374 articles, and represents the themes of "machine learning, text classification, and web semistructured data querying." Clusters 3 and 10 (cluster 7's closest neighbors) also pertain to machine learning but lack the web orientation.

The second most rapidly growing field is represented by cluster 4; it includes papers mostly on wavelet estimators. The size of the cluster is 309 and the word "wavelet" appears in 173 of the titles.

We also noted that clusters of larger sizes often show trends (i.e. have slope sufficiently far from zero), e.g. clusters 9 and 15; however large clusters do tend to grow closer to the rate of the entire dataset in general. (Note the "cluster" of clusters in the upper left corner of the MDS plot, they all have growth rates close to that of the entire dataset, their themes are "neural networks, web querying, and association rule mining.")

On a more abstract level, we note that the MDS plot clearly separates software and hardware communities, with clusters 14, 9, and 6 serving as liaisons: 14 - due to the large size of the cluster, 9 - "languages, compilers and garbage collection", and 6 - "distributed networks, protocols, corba, middleware". The "cluster" of clusters 1, 4, and 2 are more mathematical than the rest.

Cluster 15 is very close by similarity metrics to cluster 5 with the latter exhibiting significant relative growth. This is an interesting finding that deserves further investigation — the clusters have a high similarity between them (clusters are soft) with one clearly declining and another clearly growing (relative to the entire dataset). At a first glance both clusters appear to cover the topic of network computing. The clusters have 250 papers in common (the sizes of the clusters are 1,079 and 5,184 respectively). To see what accounts for such a big difference in growth rates, we chose the 50 most frequent words from the titles of the papers of each cluster (with stop words removed), and we intersected these two sets and analyzed their overlap and disjoint members (Table 4).

While sharing the common topic of networking and parallel computing (the size of the intersection is 28 terms), there is clear difference in the orientation of the two clusters - the growing cluster (number 5) is heavily oriented toward web computing (with frequent terms such as www, mobile, proxy), whereas the declining cluster (number 15) represents research in parallel computing concerning more "local" issues such as machine clusters, multiprocessors, languages (compiling), and multithreading.

6 Summary and Future Work

We have introduced a method for clustering and identifying temporal trends in scientific literature. The method allows for the identification and description of

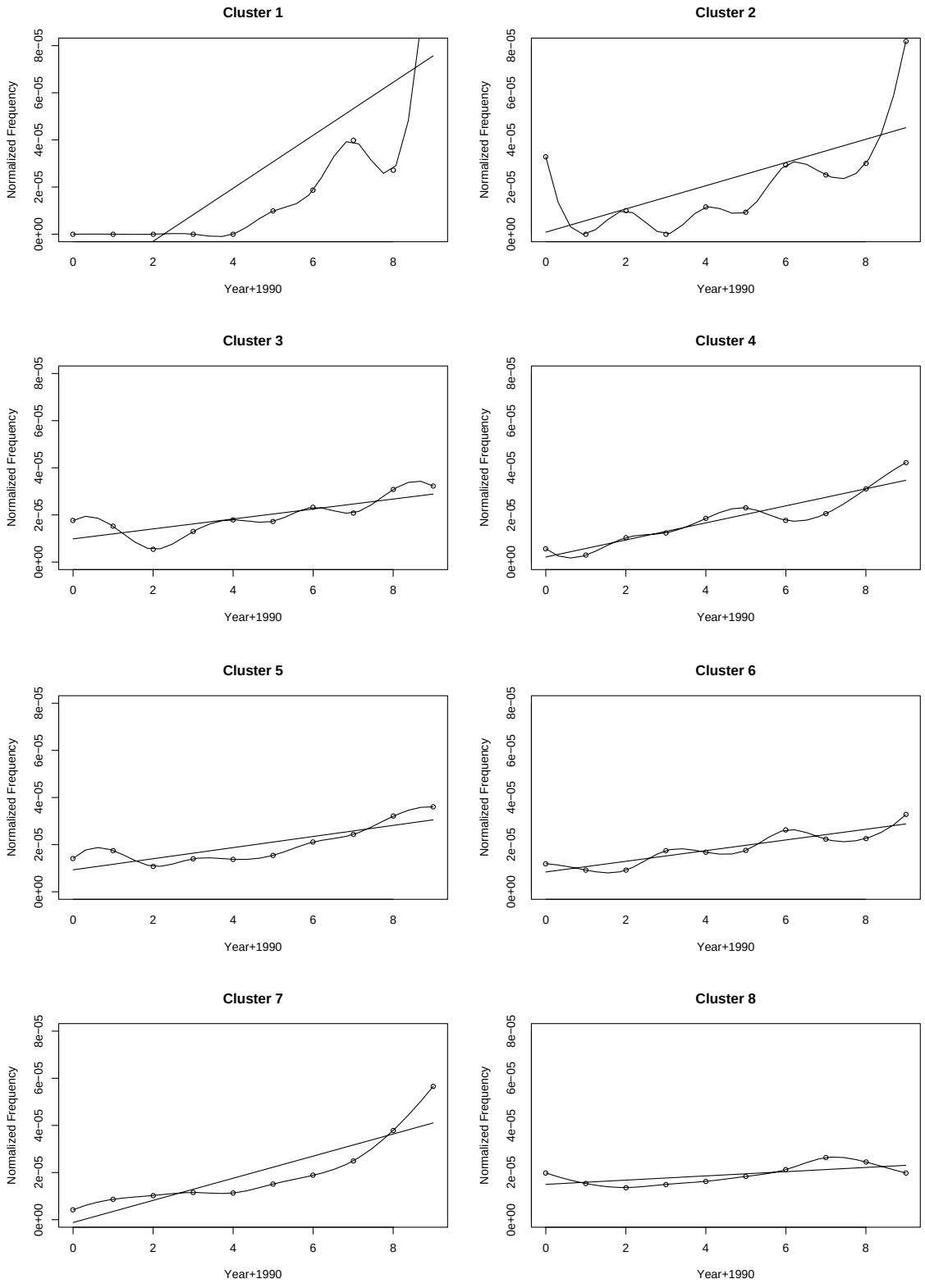

Figure 1. Regression fits for clusters 1–8.

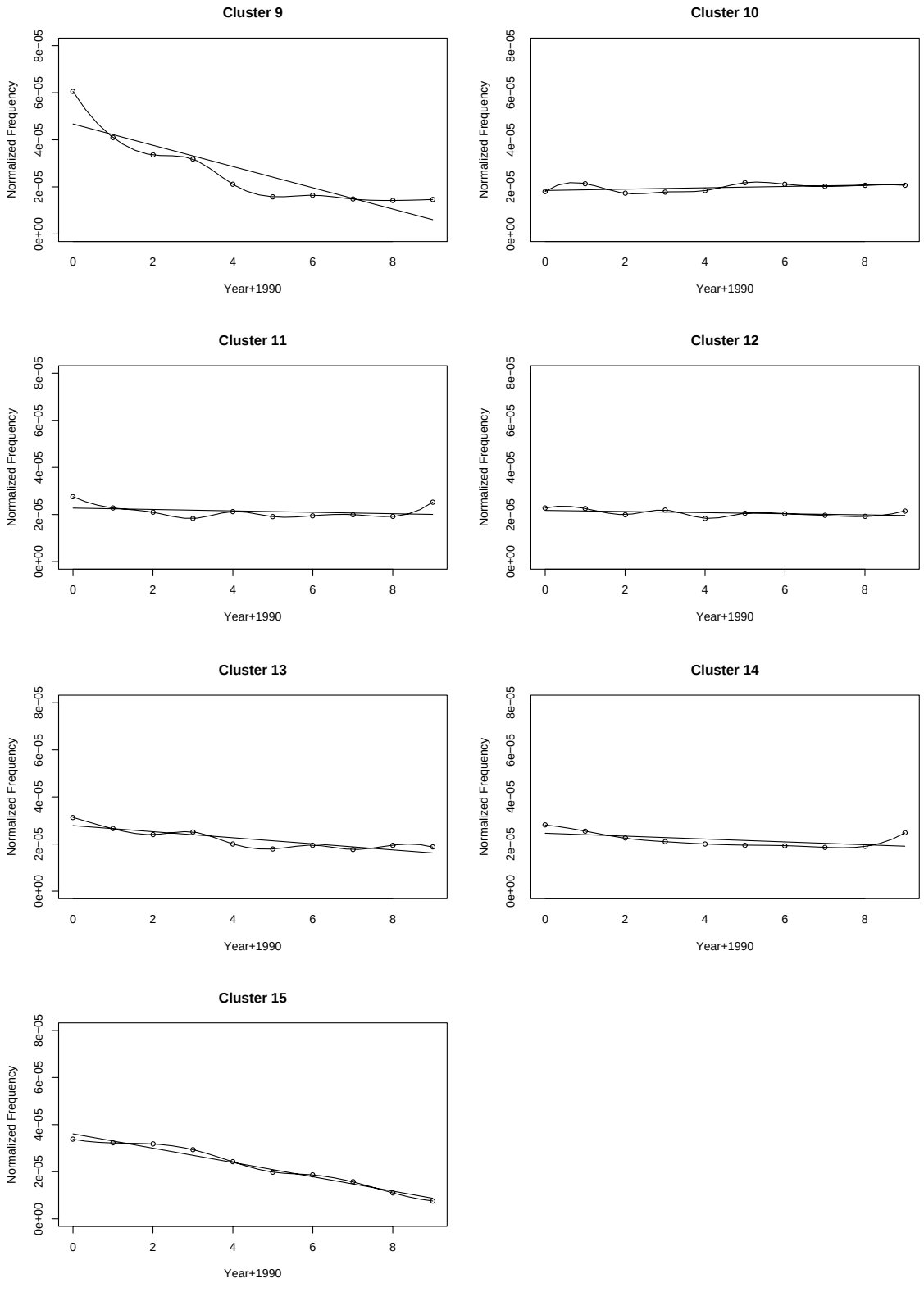

Figure 2. Regression fits for clusters 9–15.

Figure 3. MDS plot of the clusters showing the most frequent words (stemmed) in each cluster.

#	Titles of most cited (time adjusted) papers
1	Multistep Approximation Algorithms: Improved Convergence Rates through Postconditioning with Smoothing Kernels Numerical Solution of Variational Problems by Radial Basis Functions On Smoothing for Multilevel Approximation with Radial Basis Functions
2	Rewriting Aggregate Queries Using Views Rewriting of Regular Expressions and Regular Path Queries Tableau Techniques For Querying Information Sources Through Global Schemas
3	MINERVA: A Second-Generation Museum Tour-Guide Robot Map Learning and High-Speed Navigation in RHINO The Interactive Museum Tour-Guide Robot
4	Wavelet-Based Statistical Signal Processing Using Hidden Markov Models Nonlinear wavelet shrinkage with Bayes rules and Bayes factors Wavelet threshold estimators for data with correlated noise
5	Generating Representative Web Workloads for Network and Server Performance Evaluation Self-Similarity in World Wide Web Traffic Evidence and Possible Causes Maintaining Strong Cache Consistency in the World-Wide Web
6	Software Architectures for Reducing Priority Inversion and Non-determinism in Real-time Object Request Brokers The Design and Performance of a Real-Time CORBA Scheduling Service Techniques for Optimizing CORBA Middleware for Distributed Embedded Systems
7	An Evaluation of Statistical Approaches to Text Categorization Learning Information Extraction Rules for Semi-structured and Free Text Text Classification from Labeled and Unlabeled Documents using EM
8	Mining Association Rules between Sets of Items in Large Databases ROCK: A Robust Clustering Algorithm for Categorical Attributes Implementing Data Cubes Efficiently
9	Dependent Types in Practical Programming Type-Safe Linking and Modular Assembly Language Pizza into Java: Translating theory into practice
10	Improved Boosting Algorithms Using Confidence-rated Predictions Pattern Recognition and Neural Networks A decision-theoretic generalization of on-line learning and an application to boosting
11	The Lorel Query Language for Semistructured Data Querying Semi-Structured Data Querying the World Wide Web
12	RAP: An End-to-end Rate-based Congestion Control Mechanism for Realtime Streams in the Internet A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing RSVP: A New Resource ReSerVation Protocol
13	The Architectural Design of Globe: A Wide-Area Distributed System ANTS: A Toolkit for Building and Dynamically Deploying Network Protocols A Survey of Active Network Research
14	Tables Of Linear Congruential Generators Of Different Sizes And Good Lattice Structure Geometric Range Searching and Its Relatives A Method for Obtaining Digital Signatures and Public-Key Cryptosystems
15	High Performance Fortran Language Specification MPI: A Message-Passing Interface Standard Active Messages: a Mechanism for Integrated Communication and Computation

Table 3. The most cited papers in each final cluster that were centroids in the first phase clustering.

clusters in a database of scientific literature, and provides an indication of the rate of growth of different research areas. We have applied the method to a database of 150,000 computer science papers from the CiteSeer database.

The results of the proposed algorithm provide an overview of the CiteSeer database consisting of 15 clusters. These clusters were produced as the final phase with initial clustering performed around the 475 most highly cited papers, using normalized citation counts in order to avoid discriminating against newer influential papers. We have argued that scientific advancements evolve around influential papers and that a scientific discipline can be characterized by a collection of such influential papers together with the papers that they are co-cited with. The number of clusters chosen in practice would depend on the goals of a study. Several topics at a lower generalization level may be part of one larger cluster produced by the analysis. If a higher granularity is preferred a larger number of clusters should be chosen. Multidimensional scaling plots were used to produce the mapping of the disciplines, which were labeled with the most frequently occurring terms in the titles of member papers.

Intersection 5 and 15		5 and not 15		15 and not 5	
adapt	environ	tcp	file	gener	multithread
algorithm	memori	improv	mobil	techniqu	processor
analysi	model	inform	manag	interfac	high
applic	network	replic	control	runtim	softwar
architectur	object	widearea	prefetch	workstat	optim
cach	oper	www	proxi	virtual	evalu
commun	parallel	disk	traffic	approach	languag
comput	perform	multicast	internet	messag	implement
consist	protocol	resourc	servic	cluster	compil
data	scalabl	world	server	machin	multiprocessor
design	schedul	wide	web	simul	program
distribut	share				
dynam	support				
effici	system				

Table 4. High frequency words (stemmed) in combinations of clusters 5 and 15.

Our graph clustering algorithm scales extremely well by exploiting the underlying regularity of the citation database. If scientific disciplines truly coalesce around key papers, then our method of reducing the dimensionality of the problem should retain most of the important information in the citation database.

Nevertheless, it may be possible for our approach to fail to accurately characterize a hyper-linked database if the database does not naturally have clusters that tended around the most cited papers. For example, one could easily construct degenerate pathological cases in which citations are largely random and uniformly distributed. In such a case, our algorithm would be dominated by spurious graph vertices that happened to have a high in-degree due to statistical fluctuation.

We also note that our approach partially fails in that it forms one large cluster of items that appears to cover several topics. This artifact may be a side effect of the different policies that authors use for composing a bibliography.

Future work will explore alternate clustering approaches for handling the large disconnected cluster and pursue alternate similarity metrics for soft cluster similarity and for seeding the initial soft clusters. There are many ways of computing similarity between research articles including the citation based methods like co-citation and bibliographic coupling, and word-based methods such as computing TF-IDF scores [12, 11]. We are also planning to use a collection of refereed articles classified by humans to further test our approach.

References

[1] C. Chen and L. Carr. Trailblazing the literature of hypertext: author co-citation analysis (1989–1998). In *Proceedings of the 10th ACM Conference on Hypertext and hypermedia: returning to our diverse roots*, pages 51–60, 1999.

[2] E. Garfield. *Citation Indexing: Its Theory and Application in Science, Technology, and Humanities*. Wiley, New York, 1979. ISBN 089495024X.

[3] L. Kaufman and P. J. Rousseeuw. *Finding Groups In Data: An Introduction to Cluster Analysis*. Wiley-Interscience, 1990.

[4] J. B. Kruskal and M. Wish. *Multidimensional Scaling*. Sage Piblications, Beverly Hills, CA, 1978.

[5] S. Lawrence, K. Bollacker, and C. L. Giles. Indexing and retrieval of scientific literature. In *Eighth International Conference on Information and Knowledge Management, CIKM 99*, pages 139–146, Kansas Cite, Missouri, November 1999.

[6] S. Lawrence, C. L. Giles, and K. Bollacker. Digital libraries and autonomous citation indexing. *IEEE Computer*, 32(6):67–71, 1999.

[7] J. MacQueen. Some methods for classification and analysis of multivariate observations. In L. M. LeCam and J. Neyman, editors, *Proceedings Fifth Berkeley Symposium on Math. Stat. and Prob.*, pages 281–297. University of California Press, 1967.

[8] K. McCain. Mapping authors to intellectual space: Population genetics in the 1980s. pages 194–216, 1990.

[9] J. Pitkow and P. Pirolli. Life, death, and lawfulness on the electronic frontier. In *Proceedings of Human Factors in Computing Systems*, pages 383–390, 1997.

[10] W. Raghupathi and S. Nerur. Research themes and trends in artificial intelligence: An author co-citation analysis. *Intelligence*, Summer:18, 1990.

[11] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[12] G. Salton and M. McGill. *Introduction to Modern Information Retrieval*. McGraw Hill, 1983.

[13] H. Small and B. Griffith. The structure of scientific literatures: Identifying and graphing specialities. *Science Studies*, 4(17):17–40, 1974.

[14] J. Ward Jr. Hierarchical grouping to optimize an objective function. *J. Amer. Statist. Assoc.*, 58:236–244, 1963.