Abbreviated version of this report is published as
"Trends in Computer Science Research"
Apirak Hoonlor, Boleslaw K. Szymanski and M. Zaki,
Communications of the ACM, 56(10), Oct. 2013, pp.74-83

An Evolution of Computer Science Research*

Apirak Hoonlor, Boleslaw K. Szymanski, Mohammed J. Zaki, and James Thompson

Abstract

Over the past two decades, Computer Science (CS) has continued to grow as a research field. There are several studies that examine trends and emerging topics in CS research or the impact of papers on the field. In contrast, in this article, we take a closer look at the entire CS research in the past two decades by analyzing the data on publications in the ACM Digital Library and IEEE Xplore, and the grants awarded by the National Science Foundation (NSF). We identify trends, bursty topics, and interesting inter-relationships between NSF awards and CS publications, finding, for example, that if an uncommonly high frequency of a specific topic is observed in publications, the funding for this topic is usually increased. We also analyze CS researchers and communities, finding that only a small fraction of authors attribute their work to the same research area for a long period of time, reflecting for instance the emphasis on novelty (use of new keywords) and typical academic research teams (with core faculty and more rapid turnover of students and postdocs). Finally, our work highlights the dynamic research landscape in CS, with its focus constantly moving to new challenges arising from new technological developments. Computer science is atypical science in that its universe evolves quickly, with a speed that is unprecedented even for engineers. Naturally, researchers follow the evolution of their artifacts by adjusting their research interests. We want to capture this vibrant co-evolution in this paper.

1 Introduction

Computer science is a rapidly expanding research field fueled by emerging application domains and ever-improving hardware and software that eliminate old bottlenecks, but create new challenges and opportunities for CS research. Accordingly, the number of research papers published in CS conferences and journals has been rapidly increasing for the past two decades. With growing emphasis on externally funded research in most universities, scientific research is increasingly influenced by the funding opportunities. Although many funded programs are developed in close collaboration with leading researchers, we aimed to identify more precisely relationships between funding and publications related to new topics.

There are numerous papers already published that track research trends, analyze the impact of a particular paper on the development of the field or a topic, and study the

*First Report: 03/2012, Latest Revision: 09/2013 relations between different research fields. There have also been studies in social networks investigating the overlap and evolution of social communities around a field or a topic. In this paper, we are interested in learning about the evolution of Computer Science research communities, the trends in CS research, and the impact of funding on those trends. We collected data on proposals for grants supported by the National Science Foundation (NSF) and CS publications appearing in the ACM and IEEE publication databases. We used various methodologies to analyze research communities, research trends, and relation between awarded grants and changes in communities and trends. Within the Computer Science research communities, we also analyzed the connections between each research topics. We highlight the interesting trends discovered by our analysis.

1. While the number of CS publications continue to grow in every field, data from the ACM Digital Library and IEEE Xplore show that in the last decade the proportion of research done in mathematics of computing has decreased considerably. On the other hand, the proportion of publications on information system such as data mining, machine learning, and world wide web is increasing recently.

2. The term most used in an abstract is algorithm, which is not surprising as it is a fundamental CS topic. The next three topics in popularity are neural network, database, and Internet, indicating the recent major research interests.

3. Cloud computing, social media, and social network have strong upward trends within the last five years. However, we have found that two-year publication proportion trend is always followed by the reverse in the subsequent year.

4. A burst of new keywords in grants generally precedes their burst in publications; less than 1/3 of new keywords burst in publications first, reflecting the importance of funding for success of new CS fields.

5. While typical research community in Computer Science contains 5 to 6 members, its membership constantly changes. After four years, only one or two core people in the initial research group remain, which is consistent with the university setting in which one or two faculty members supervise a group of three to five postdocs and graduate students.

6. A typical scientist's research focus changes in roughly a 10-year cycle and often includes a once-in-a-career dramatic shift, likely in response to evolving technology creating new CS fields.

The rest of the paper is organized as follow. We discuss related work in Section 2. In Sections, 3 and 4, we introduce our datasets and the methods used in our analysis, respectively. We present and explain our observations in Section 5. Finally, we provide conclusions in Section 6.

2 Related Work

Trend analysis has been actively researched for a long time and applied to many types of datasets ranging from medical data [20], to weather information [18] and stock markets [7]. Many publications track research trends, analyze the impact of a particular paper on the development of a field or a topic, and study the relationships between different research fields. The Web of Science [21] collected data since 1900 on nearly 50 million publications in multiple scientific disciplines. It analyzed this data at various levels of detail by looking at the overall trends and patterns of emerging fields of research, and the influence of an individual paper on related research areas. Over the past decade, besides the Web of Science, there have also been studies in social networks investigating the overlap and evolution of social communities around a field or a topic. In [22, 23], the authors explore methods and visualizations for scientific research landscape and analyze the impact of each research area quantified by the collective cross-disciplinary citations of each paper. Porter and Rafols [19] analyze the citation information to find the evidence of collaboration across fields in scientific research. Other examples of such analysis are the network models for studying the structure of the social science collaboration network [17], and the analysis of women's authorship in CS publications in the ACM digital library [4].

Several studies have focused challenges, directions, and landscapes in specific CS fields [2, 11], and on specific CS topics [12, 25]. Chen [3] reported the studies of the international intellectual landscape based partly on the publication data in nanotechnology from Thomson Science Citation Index. The data was analyzed from various angles such as who the contributors of the paper were and from which country, what funding programs were active in such country and for those contributors, and what economic advantages each country offered for technology development. The studied found that researchers from US has published the most papers on nanotechnology, while China has largest increment in publications as it rose to the second place in contribution, even though the research in China did not begin until after 1991.

Other research related to our work focuses on social networks, especially on the topic of evolution and overlapping of social communities. Goldberg et al. [9] identify overlapping communities using a locally optimal algorithm. The algorithm can recover overlapping communities from a large network, such as LiveJournal network, without performing a global analysis on the network. Lancichinetti et al. [14] propose another locally optimal algorithm using a fitness function that discovers overlapping communities and their hidden hierarchical structures. Other related topics emerge from studies of overlapping of social communities. Sun et al. [26] present a Dirichlet process mixture model that can recover the evolution of communities over time. Goldberg et al. [10] introduced a dynamic algorithm that recovered chains of evolutionary communities.

3 Datasets

We used the ACM, IEEE, and NSF datasets from which we collected data on publications from 1990 to 2010. The National Science Foundation (NSF) records before 1990 were incomplete (such as lacking abstracts). Only 10% of publications in ACM and IEEE datasets were published before 1990. So, our time range covers nearly all publications in those datasets.

1. **ACM Dataset**: ACM Digital Library [1] contains the record of articles published with ACM and its affiliated organizations. For this dataset, we extracted the number of papers listed in top categories of the 1998 ACM Computing Classification System (CCS) (see the CSS at http://www.acm.org/about/class/1998/). We excluded the General Literature category because it includes too many non-research topics such as biography, reference, etc. The ACM dataset contains authors, title, abstract, year, publication venue, author-defined keywords, and ACM classification categories for each of the $116,003$ articles published between the year 1990 and 2010. We used ACM CCS and the author-defined keywords to respectively study the broader and static versus the finer and dynamic views of the CS landscape and trends. Only the author-defined keywords were used to identify the relationships between researchers, yielding smaller research groups than using ACM CCS would.

2. **IEEE Dataset**: For the IEEE dataset, the topics were extracted from 16 Wikipedia articles on CS research areas identified in the main Wikipedia CS article, since it does not have the same topic classification system as the ACM dataset. Over four hundred research topics in Computer Sciences are used as queries to extract paper abstracts from 1990 to 2010. The research topics included major research areas such as artificial intelligence, computer architecture, and computer engineering, as well as the branches of those major areas such as compilers, computer security, image processing and machine learning. The full list of queries is included in the supplementary material. We queried IEEE Xplore digital library [8] to retrieve all the conference papers whose abstracts contain at least one of the query terms. The title, paper id, the conference name, year of publications, list of authors, and the abstract are collected for each retrieved paper. Note that if the retrieved paper does not contain both its abstract and conference name, we ignore that paper. A total of $458,385$ papers were extracted.

3. **NSF Dataset**: NSF made the information on the awarded grants available online via its website `www.nsf.gov`. We collected the proposals of grants awarded by all directorates in NSF supporting CS research (the detailed list is provided in the supplementary material). From year 1990 to 2010, we collected the award number, title, start date, and abstract for each grant (records without abstracts were ignored). In total, $21,687$ awarded grants were retrieved.

For ACM and IEEE datasets, we created two data indexes: (i) authors and their publications venues, and (ii) papers and their keywords/topics.

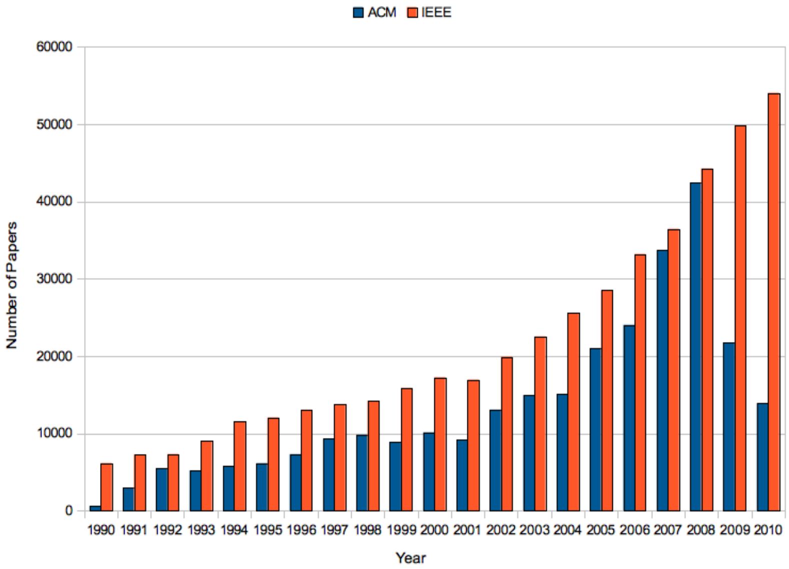


Figure 1: The number of records found each year between 1990 and 2010 in the ACM and IEEE datasets.

Fig. 1 shows that the IEEE and ACM datasets display about 11% yearly growth in the number of publication from 1990 to 2010, (the difference in the the last two years is caused by partial availability of data on non-ACM publications in the ACM dataset).

4 Methodologies

Using sequence mining [28], network extraction and visualization [22], bursty words detection [15], clustering with bursty keywords [13], and network evolution [10], we investigate: (i) changes over time in the computer science research landscape, (ii) interactions of CS research communities, (iii) similarities and dissimilarities between research topics, and (iv) the impact of funding on publications, and vice versa. The term "bursty keywords" in this context refers to keywords appearing with uncommonly high frequency during some intervals; such intervals may include multiple spikes of a keyword's frequency, as defined in Section 4.0.2. Note that such interval may include multiple spikes of a keyword frequency. The key software and methodologies used in this paper are Map Generator, Bursty Words, Trend Analysis, Sequence Mining, and Network Evolution.

4.0.1 Map Generator

For IEEE and ACM datasets, we created a weighted undirected graph to represent the inter-connectivity of research topics in Computer Science for every year from 1980 to 2010. The nodes of the graph are research topics. For IEEE dataset, the weight of the edge between nodes A and B is the number of abstracts that mention both topics. For

ACM dataset, the number of papers that contain both A and B as keywords was used as the weight of the link between them. To analyze the community structure in the network of Computer Science research, we used the map generator [6] which is a Flash applet using the map equation [22] to find the sub-networks of the given network. The map equation is a random walk based network clustering method. Essentially, nodes are clustered together if they are visited together in many walks. This allows us to detect (i) which topic areas are the bridges between major research fields, (ii) which topics receive the most attentions and from which fields, and (iii) how the clusters evolve from one year to the next.

4.0.2 Burstiness Score and Bursty Period

A bursty period is defined as the maximum sum segment – the period whose total burstiness score is greater than zero [15]. We used the burstiness score defined in equation 1 proposed by [15] to find the bursty score of each word at each time step.

$$Burst(w,t) = \left(\frac{|d_t : w \in d_t|}{|d : w \in d|} - \frac{1}{T}\right) \tag{1}$$

where w is the keyword/topic of interest, t is a time period, d_t is a document created during time t, d is any document, and T is the total time over which documents were created. The burstiness score measures how often w is in t compared to its occurrences in T. A positive score implies that w appears more often during the "bursty period" t than over the total time T. A negative score says otherwise. Finally, the maximal segments of burstiness scores in the sequence of documents are recovered using the linear-time maximum sum algorithm by Ruzzo and Tompa [24, 15]. We selected ten research topics with the highest number of publications. In other words, we tried to find the hottest research topics related to the top research topics at their peaks. We used these burstiness and bursty periods to find the time periods during which a keyword is bursty, i.e., when its burstiness score is greater than a predefined threshold.

We also used these notions to extract the following: "given a word a, what is its bursty period, and which keywords associated with it are also bursty in such period?". Essentially, the patterns that we want to extract are the correlated terms (a, B) where B is the set of bursty words in the bursty periods of a. To do that, we first need to find the bursty periods of a. Then, for each bursty period, we find words bursty in it.

4.0.3 Trend Analysis

To quantify the trends, we look at how fast each keyword grows and which direction it is heading using linear regression that measured the relationship between the number of publications and the time of publications. Then, we created linear trend lines for each keyword frequency and a linear model for the normalized data from the last 21 years and the last five years. We labeled the keyword as "up" trend, if its estimated trend line has the slope greater than zero and as "down" trend, otherwise. We extracted the up and down trends from the keywords with at least 100 document frequency from ACM and IEEE datasets.

4.0.4 Sequence Mining

Frequent sequences are mined using the cSpade program [28] that allows for multiple constraints: length and width limitations on the sequences, minimum and maximum gap constraints on consecutive sequence elements, time window on allowable sequences, and item constraints. For ACM dataset, we created two sets of data. First one contains the list of authors' publication venues from the list shown in supplementary material E. The second is the list of authors' major research field according to ACM Computing Classification System.

4.0.5 Network Evolution

Tracking evolution of such communities requires identifying all evolutionary sequences of communities in a dynamically changing social network. A Sub-network (cluster) discovered in the CS research network by graph clustering algorithm can be considered a community. For our datasets, there are two interesting questions related to the tracking of communities: (i) "how do the research communities in Computer Science evolve over time?", and (ii) "how do the research topics in Computer Science themselves evolve over time?". For the first question, we created the research-community network by looking at the connections between authors, and author-defined keywords, i.e., if two authors use the same author-defined keyword, then the link between them is of weight one. For the second question, we created the research-topic network by looking at the connections between author-defined keywords, and papers, i.e., if two keywords appear in the same paper, then they have a link of weight one between them. To track evolutions of these communities, we used the framework for analyzing the evolution of social communities developed by [10]. The framework searches for the link between communities in consecutive time-steps. A link is formed between two communities if their intersection is non-empty and the similarity between them is higher than a certain threshold.

5 Results and discussions

5.1 Landscapes of Computer Science research

We looked at the evolution of the landscape of Computer Science research from 1990 to 2010. Figure 2 shows the number of papers listed under each category from 1990 to 2010. With the exception of the last two years, the number of publications in each category increased each year. Many ACM records from 2009 to 2010, collected during the spring of 2011, did not have ACM classification categories, and thus were excluded from our study. This explains the drop in the number of records for the last two years seen for the ACM study. Figure 3 shows the ratio of publications listed under each category for the 1990 - 2010 period. We looked closer at individual research areas, by looking at

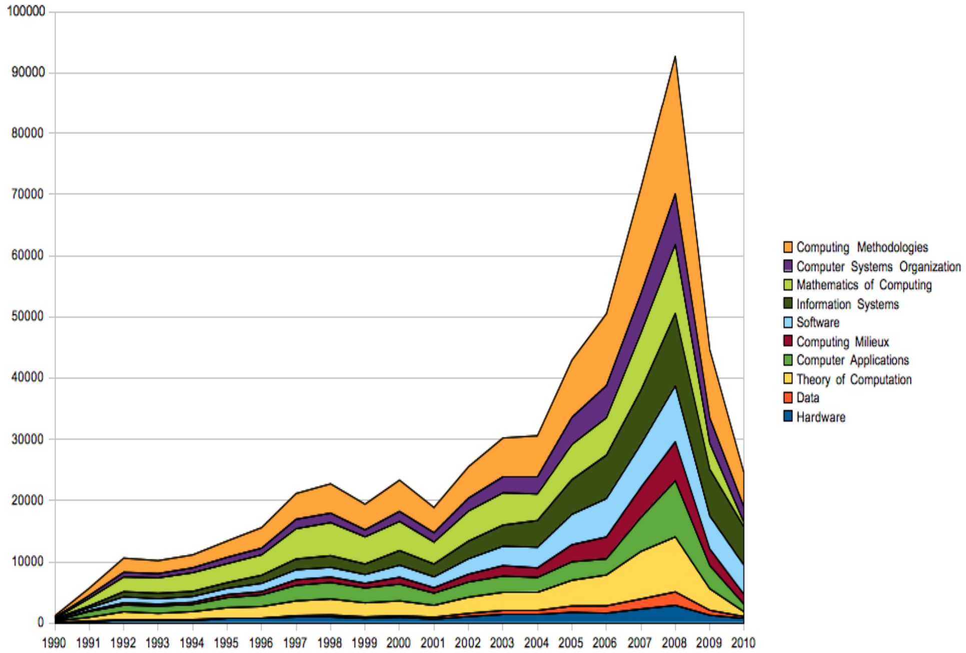


Figure 2: A landscape of Computer Science research between 1990 and 2010 from the ACM dataset.

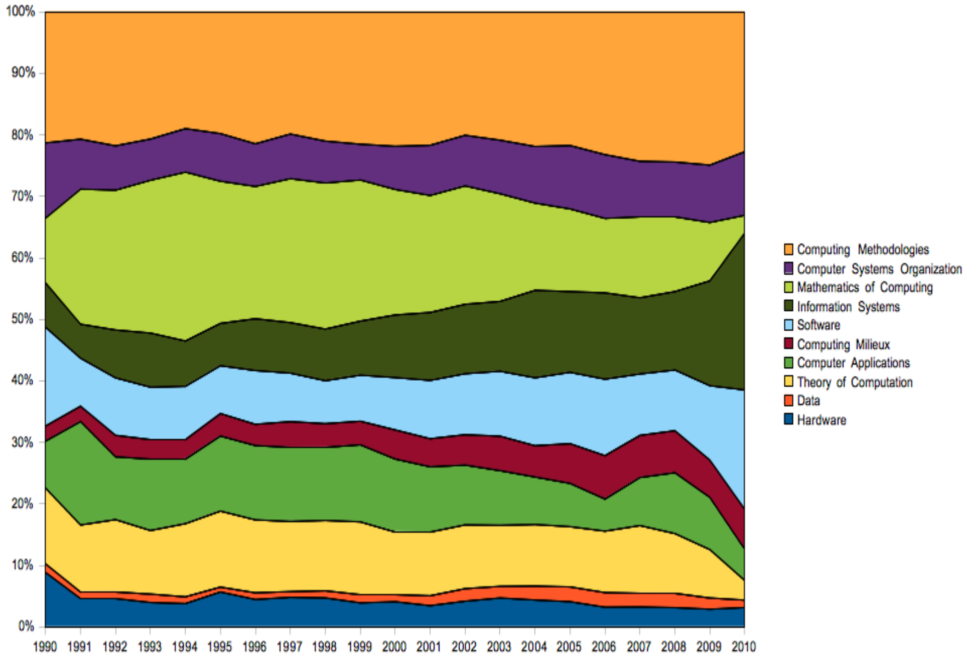


Figure 3: Another view of landscape of Computer Science research between 1990 and 2010 from the ACM dataset.

their occurrences in each decade. Table 1 and Table 2 show the author-defined keywords whose occurrence changed drastically in the past two decades. From Figure 3, after 1994 the number of publication in mathematics of computing category shrunk considerably compared to other categories. From the table the Table 1, the author-defined keywords that contributed to this drop were control theory and logic. We attributed this drop to shift of focus from general issues to challenges specific to an area with which such publications are increasingly associated. In contrast, publications in information systems continually accelerated their growth. Figure 3 shows that the growth of publications in information systems category continued to increase in comparison to other categories. Table 2 confirms that the author-defined keywords used increasingly frequently were Internet-related, such as XML, Internet, web services, and semantic web.

Table 1: The list of author-defined keywords in the papers in mathematics of computing category, whose occurrence dropped by at least half from the 1990s to 2000s.

Keyword	1990s	2000s
robust control	208	93
discrete-time systems	84	40
control theory	87	36
design of algorithms	83	29
singular perturbations	75	34
fuzzy topology	72	24
viscosity solutions	61	27
approximate reasoning	63	25
nonlinear control systems	69	9
membership functions	52	24
feedback control	53	22
expert systems	51	22
atm	52	19
calculus of variations	52	18
time-varying systems	45	20
linear complementarity problem	45	20
state feedback	52	13
algebra	41	18
fuzzy relations	40	17
quasi-newton methods	39	18

For IEEE dataset, Figure 4 contains the area plot of the number of papers, whose abstract mentioned the major Computer Science research topics from 1990 to 2010. Those topics and their corresponding conferences extracted from Wikipedia are listed in supplementary material E. For IEEE dataset, similar to the ACM dataset, the fastest growing research area was information science and information retrieval. Figure 5 contains the percentage of publications whose abstracts mentioned the major Computer

Table 2: The list of Author-defined Keywords in the papers in Information Systems category, whose occurrence at least double from the 1990s in 2000s.

Keyword	1990s	2000s
data mining	106	1847
information retrieval	243	1226
XML	22	889
evaluation	63	842
clustering	37	792
internet	197	609
web services	2	801
visualization	104	682
usability	73	672
semantic web	0	730
collaboration	101	594
virtual reality	147	539
design	61	545
ontology	16	582
machine learning	59	527
privacy	28	555
information visualization	92	469
classification	41	516
ubiquitous computing	40	508
security	58	480

Science research topics from 1990 to 2010.

To better see the impact of information systems, we extracted the top 25 research topics from the ACM and IEEE datasets, as shown in Table 3. We quantified the results in two ways: Document Frequency (DF) and Term Frequency - Inverse Document Frequency (TFIDF). DF of term/keyword k is the number of documents that contains it. TFIDF of term k is the sum of **tf-idf** weights of term/keyword k over all documents. The **tf-idf** weight of k in document d is defined as

$$\frac{n_{k,d}}{\sum_{w \in d} n_{w,d}} \cdot log \frac{|D|}{|j : k \in d_j|}$$

where $|D|$ is the number of documents, and $n_{k,d}$ is the number of times k appears in d For ACM dataset, Table 2 indicates that most publications in collaboration, data mining, information retrieval, machine learning, privacy, and XML appeared from 2000 to 2010. These research topics are also in both lists in Table 3, showing a remarkable research trend in Computer Science. The terms Internet and world wide web did not appear in any publication until 1995, but the related topics were present since early 1990. During the 1990 – 1997 period, 376 NSF grants and nine IEEE papers mentioned NSFNET in

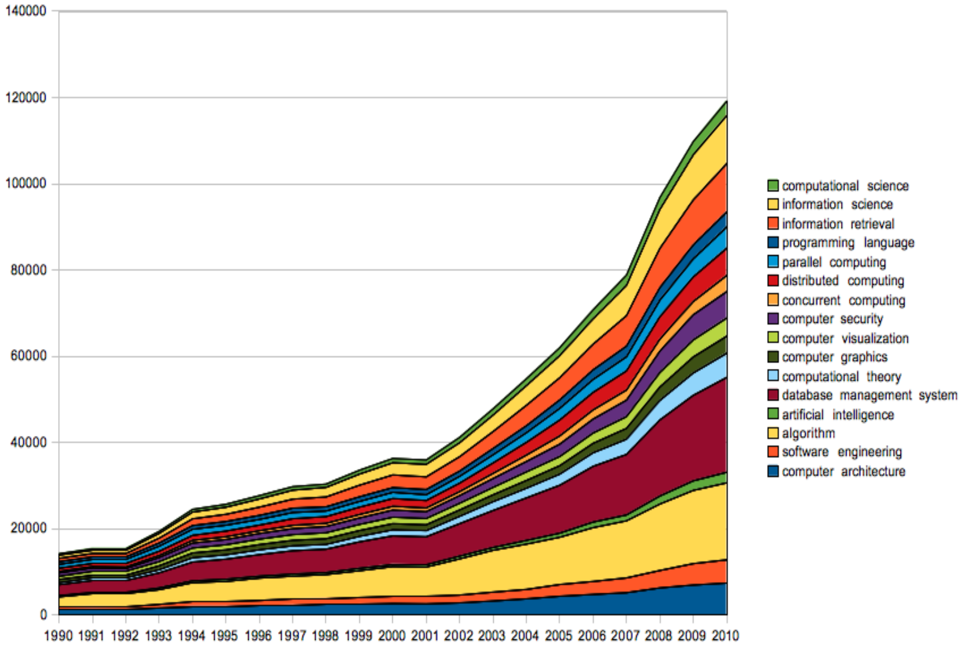


Figure 4: A landscape of Computer Science research between 1990 and 2010 from IEEE dataset.

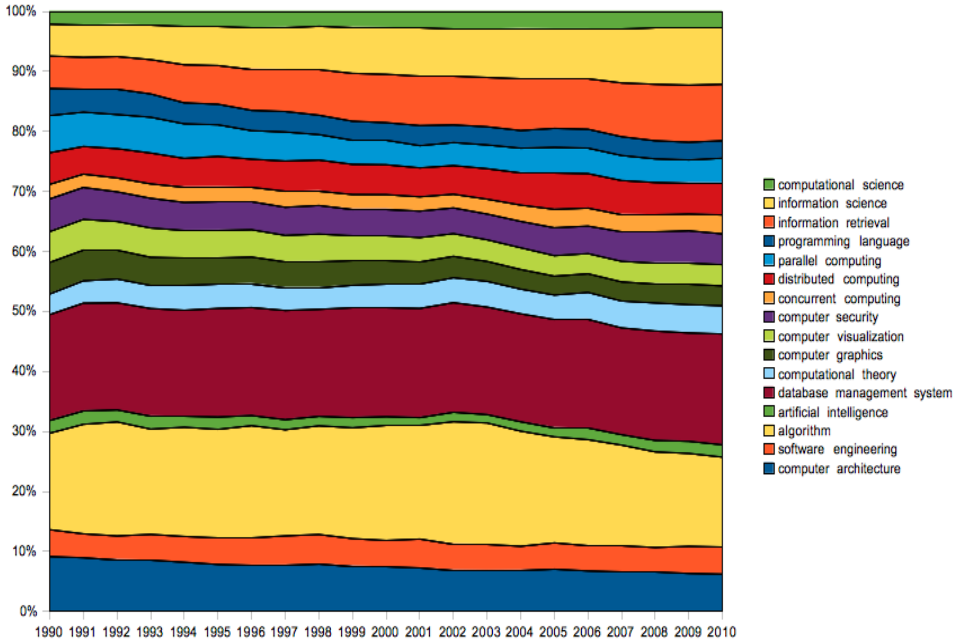
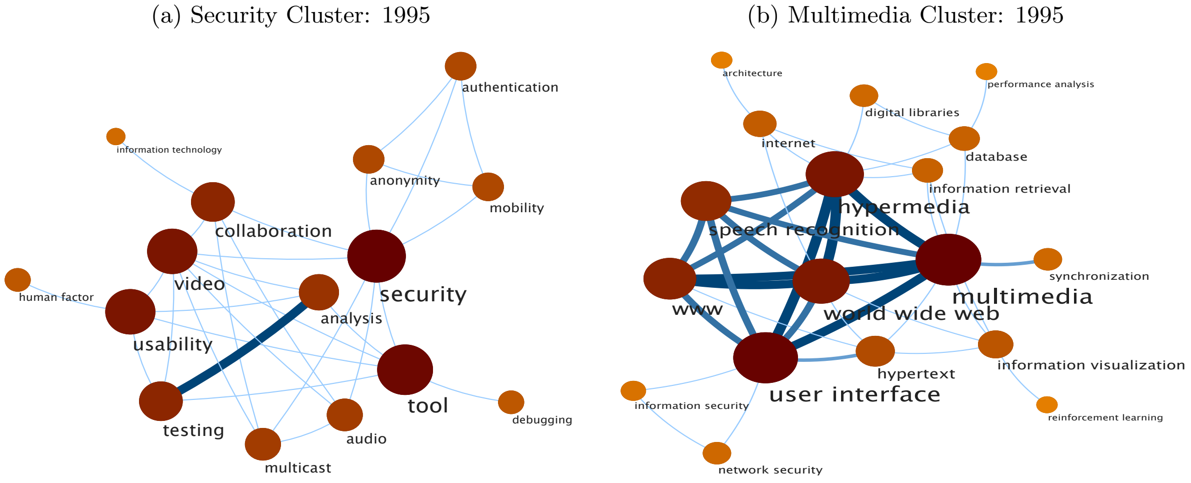


Figure 5: Another view of the landscape of Computer Science research between 1990 and 2010 from IEEE dataset.

Figure 6: The 1995 clusters of research network in (a) Security cluster, and (b) in Multimedia cluster (edge thickness represents strength of interaction).



(a) Security Cluster: 1995

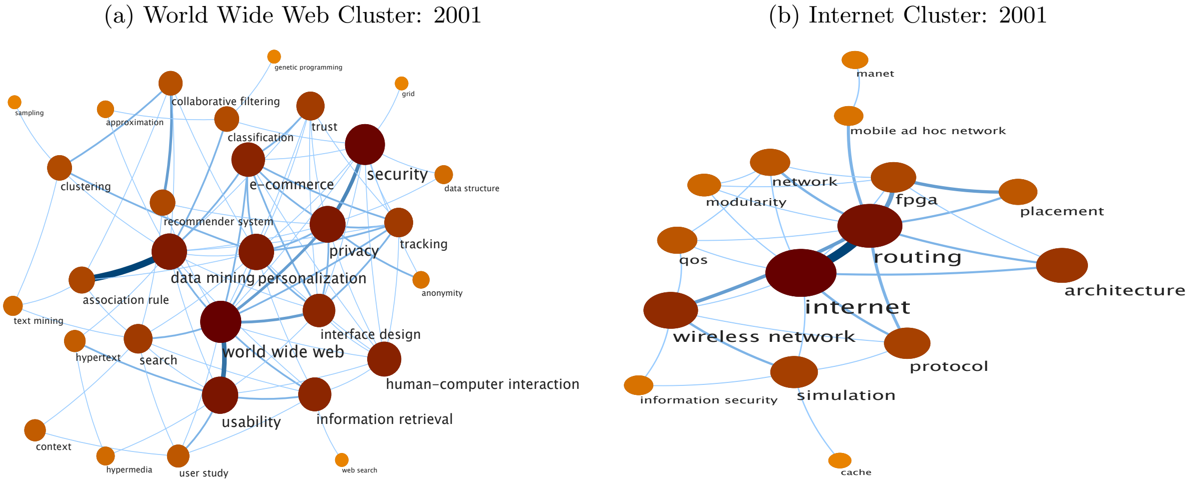
(b) Multimedia Cluster: 1995

their abstracts, but only two ACM papers used it as their keyword. Other terms such as net, prodigy, point-to-point, and inter-networking also appeared in the NSF dataset before 1995. Moreover, prodigy was bursty over the 1991 − 1992 period and TCP/IP over the 1990 − 1993 period. Figure 6 shows the research topic sub-networks created from ACM by Map Generator [6] for security and multimedia in 1995. Figure 7, shows the research topic sub-networks created from ACM by Map Generator for world wide web and Internet in 2001. Both figures show that, in 1995, world wide web was used as a keyword associated mostly with multimedia and information visualization, whereas information retrieval was used mostly with Internet. However, by the early 2000s, world wide web was used mostly with data mining and information retrieval, while Internet was mostly associated with network, protocol and routing. More recently, privacy and security have become important for world wide web, while semantic web, web 2.0, web service and XML have become major Internet topics In the IEEE dataset, database, Internet, information system, XML, telecommunications, data mining and HTML also appear in one or both of the lists in Table 3.

5.2 Bursty Period Analysis

To evaluate the influence of research funding on publications, or the reverse direction, we extracted bursty periods of author-defined keywords from ACM and NSF datasets as well as from IEEE and NSF datasets. We used the author-defined keywords because only the ACM records are classified using CCS. For each pair of datasets, we analyzed in which dataset a keyword's bursty period begins first, and how long it takes for the keyword to become bursty in the other dataset. In cases with more than one bursty period, we also looked at the keyword's burstiness score in each bursty period. We then

Figure 7: The 2001 clusters of research network in (a) World Wide Web cluster, and (b) Internet cluster (edge thickness represents strength of interaction).


(a) World Wide Web Cluster: 2001

(b) Internet Cluster: 2001

tabulated the percentage cases in which the later burstiness scores increase, decrease, or stay unchanged. We identified the changes if there were bursty periods in both datasets in a pair.

For the ACM-NSF pair, if a keyword became bursty in ACM, it became bursty in NSF 2.4 years later on average, but in the reverse case, the average delay was 4.8 years. This shows that if a new area is initiated by NSF, the increase in publications is delayed by the time researchers need to obtain grants and start research leading to a publication. If the keywords were bursty in both datasets, in 75% of such cases the keyword became bursty in the NSF dataset before it did in the ACM dataset, showing that NSF funding often increases interest in the supported areas. The reverse was true for about $16-17\%$ of the cases. Examples of bursts appearing first in the NSF dataset are data mining and search engine that became bursty in 1999 for NSF and in 2000 for ACM. The reverse cases include bioinformatics (2003 in ACM and 2004 in NSF) and semantic web (2004 in ACM and 2006 in NSF).

Tables 4 and 5 show the burst period comparison on the top 10 most frequent keywords that are bursty in NSF dataset before they are bursty in the ACM and IEEE datasets, respectively. It should be noted that Tables 4 and 5 contained results of bursty period analysis performed on the normalized data, while Tables 6 and 7 contain the raw data analysis. Since the number of publications increased every year, an increment in the publications in each area is positive, yet certain areas may lose their share of overall publication. Such discrepancy between two types of analysis can recover a period when a research topic is seemingly bursty in the raw data but only because of the overall publication increased.

For ACM-NSF pair, 20 words out of the top 25 most frequent words according to the document frequency became bursty first in NSF dataset. Algorithm and performance evaluation are two keywords which were not bursty in the NSF dataset, while web service and Internet were bursty in ACM dataset first (2004 and 1997, respectively), and in NSF later (2008, 2000). Computational complexity became bursty in both dataset in 2000.

For the IEEE-NSF pair, a keyword that is first bursty in IEEE becomes bursty in NSF 3.4 years later on average. In the reverse case, the average delay was 5.7 years. The difference between these two delays and its reason are the same as in the ACM dataset. Yet, both delays are by one year longer than in the ACM-NSF pair, which we conjecture result from a larger ratio of computer engineering topics in IEEE than in ACM, and presumably due to a larger fraction of support for IEEE publications coming from non-NSF source.

If a keyword was bursty in both datasets, 68% of the time the keyword became bursty in the NSF dataset first, again consistently with the ACM dataset. The reverse was true for 16% of the time. Table 5 has one extra column titled NSF-L that shows the last bursty year in NSF dataset for the keywords that were bursty in both datasets. Only internet (in 2000) and telecommunications (in 1995) became bursty at the same time in both dataset. A few keywords that became bursty in the IEEE dataset first are real-time database (1994 versus 1999 for NSF), procedural programming (1992 versus 1993), and neurobiological (1996 versus 2001). Interestingly, peer-to-peer network was bursty in IEEE dataset from 2003 to 2010 but never in the NSF dataset, which may indicate that the corresponding challenges were funded mostly from non-NSF sources. Other interesting keywords that did not appear on the top 10 keywords in the Table 5, but were bursty in the NSF dataset first are assembly language (1990 versus 1993), Bayesian network (2001 versus 2004) and computational geometry (1991 versus 1993).

We also analyzed the NSF dataset versus IEEE or ACM datasets and vice versa. For each such pair and each year from 1990 to 2010, we searched for the year in which the number of entries changed compared to any of the previous four years in the first database. For each such change, we searched in the other dataset for a change in any of the next four years. The relative change values ranged from -0.5 to 0.5, which we grouped into bins of size 0.1. We counted the frequency of the change in one dataset followed by a change in the other.

For the NSF dataset versus either ACM or IEEE dataset, a 10% or larger increase in the number of NSF grants awarded for a given topic from the previous few years was followed by an increase (with 75% probability) in the number of published papers on this topic of at least 10% in the next three years and 20% in the next four years. Topics with such an increase include data mining, information extraction, and wireless network. On the other hand, an increase of 10% in the number of published papers in a given topic in the ACM data set was followed with a 75% probability of increase (usually less than 10%) in the number of NSF grant awarded on the same topic. Examples are e-government, groupware, and knowledge management.

For a keyword in NSF with multiple bursty periods, the following bursty period had a higher/lower/equal burstiness score in 37%/51%/12% of the cases. For IEEE, it was 29%/64%/7%, respectively, while for ACM, it was 12%/85%/4%. However, for interleaved or overlapped bursty periods in the NSF and IEEE datasets, if the bursty period was first in the IEEE dataset, the following NSF bursty period had a higher/lower/equal burstiness score in 31%/22%/47% of the cases. In the reverse case, it was 36%/10%/55%. The same analysis of the NSF and ACM datasets shows that the following NSF bursty period had higher/lower/equal burstiness score for 38%/14%/48% of the cases while in the reverse case, for the following ACM bursty period those numbers were 8%/8%/84%.

The reason for a large percentage of equal burstiness scores is that a bursty period in one dataset was often a subset of the bursty period in another. Burstiness scores tend to decrease in the periods following a bursty period in the NSF dataset. Since novelty is highly valued in publications, authors tend to stress new aspects of their work in abstracts and keywords, contributing to the observed pattern. Yet during an NSF burstiness period, publication burstiness scores were more likely to increase than decrease, confirming that sustained NSF funding is essential for maintaining interest in the given topic.

The burstiest periods are shown in Table 6 for the ACM dataset and in Table 7 for the IEEE dataset. Further analysis identifies for each bursty period, associated keywords burst together. For example, in Table 6, wireless sensor networks (WSN) is temporally related to simulation, security and clustering in the order of bursty periods. This order corresponds to the temporal evolution of WSN research area that initially focused on simulations of networks, then on security issues and finally on clustering algorithms. Another conclusion from this table is that data mining is more broadly used than information retrieval since the former is used in computational science, web mining, time series mining and security, while the latter is used mainly in the web related topics. Text mining is temporally related to both information retrieval and data mining.

Multiple bursty periods for a keyword contain interesting temporally correlated terms. For example, there are three bursty periods for the keyword “scheduling”: $1990-1991$, $1999-1999$, and $2001-2006$. In 1999, scheduling correlated (list in the order of burstiness ranking) with genetic algorithms, parallel processing, performance evaluation, embedded systems, approximation algorithm, multimedia, quality of service, optimization, and heuristics. In the period $2001-2006$, such keywords, listed in the same order, were approximation algorithms, multimedia, online algorithms, real-time, embedded systems, fairness, multiprocessor, quality of service, and genetic algorithms. Hence, initially, both real-time systems and parallel processing were related to scheduling, later expanding to genetic algorithms and embedded systems. In the last few years of its bursty periods, scheduling correlated also with multimedia, online algorithm, and fairness. An alternative look at such links done via the co-reference document frequency instead of the burstiness score is shown in Table 8 for the ACM dataset and Table 9 for the IEEE dataset.

5.3 Trend Analysis

This section analyzes research trends using the linear regression trend line and changing popularity of topics based on fraction of papers containing a given keyword in each year. We generated a trend line for each keyword fraction and used its slope for ranking. We fitted the trend lines to data from the preceding two to six years in order to predict keyword fractions for the following year. For the IEEE, ACM and NSF datasets, we found that the more data we have, the better the prediction we got, as shown in Table 10.

In all datasets, we observed that if a trend based on two years of data has a positive slope, i.e., the fraction of publications increased from the previous to the current year, then the subsequent year fraction declines. We also used the trend line based on the NSF dataset to predict fractions for the following year in the ACM and IEEE datasets. The results show that this is a poor predictor, as is using the ACM and IEEE trends to predict the number of grants awarded by NSF. The accuracy on all these models was less than 50%.

The top 20 up and down trends for the last 21 years (1990-2010) and 5 years (2006-2010) are shown in Figures 8 and 9, respectively for the ACM dataset, and in Figures 10 and 11 for the IEEE datasets. In contrast to ACM dataset, IEEE dataset did not show significant decrease between the top and the bottom trends because research topics appeared in the abstract over a longer period of time than that for the author-defined keywords. Further, we used the list of Computer Science conferences (provided in the Supplementary Materials section) to categorize each paper in the IEEE and ACM datasets. The growth in different areas cannot be statistically compared because of vast differences in the number of conferences in each field, and the number of papers published in each conference. Nevertheless, Figures 14 and 12 show a growth of about 11% experienced by most CS publications. In the figure, each topic represents a set of CS conferences. This is in contrast to Figure 1 that uses the ACM classification or IEEE Xplore keywords. Also, we do not see the same drop in the number of records for the ACM dataset, since every record contains the publication venue. For instance, if a conference is on security and OS, we indexed all the papers published in that conference under both the security and OS topics.

5.4 Network of Computer Science Research

Since we looked back over the period $1990 - 2010$, we were able to monitor when connections between two fields occurred or changed. We extracted two sets of keywords, those that have never appeared in the same article, and those that have appeared in at least 5 articles every year. For IEEE dataset, we performed an analysis on the Algorithm topic first. Then, we removed the algorithm node from the network because this term is used in almost every CS research paper to describe how data are processed. Hence, keeping

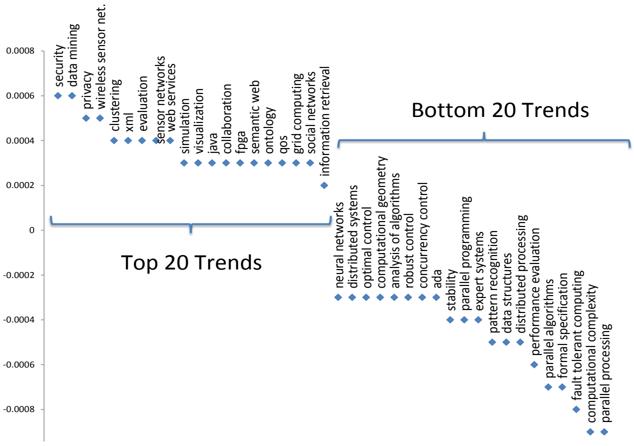


Figure 8: The top and bottom 20 trends 1990 - 2010 from the ACM dataset.

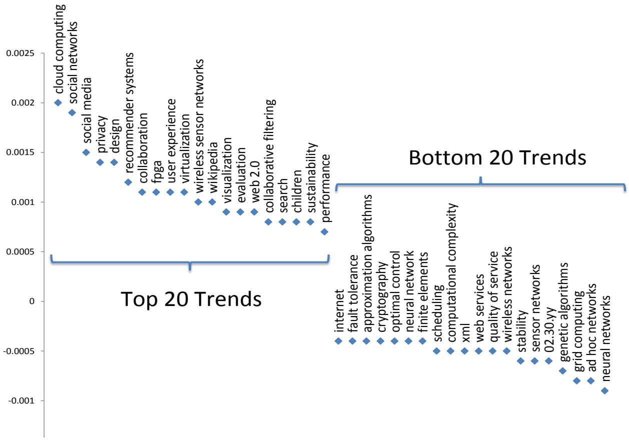


Figure 9: The top and bottom 20 trends 2006 - 2010 from the ACM dataset.

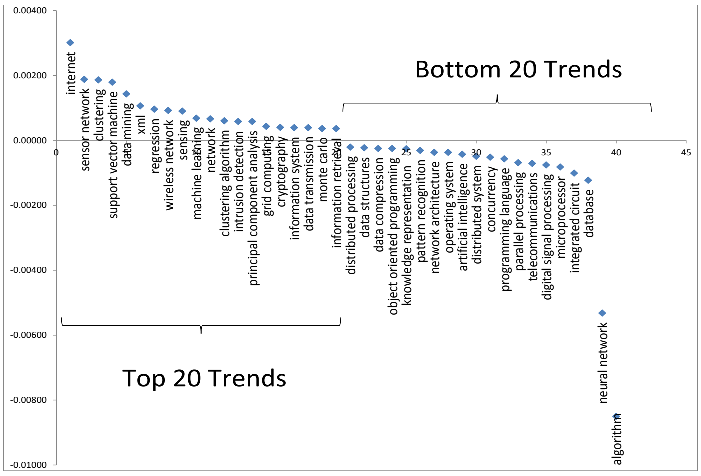


Figure 10: The top and bottom 20 trends 1990 - 2010 from the IEEE dataset.

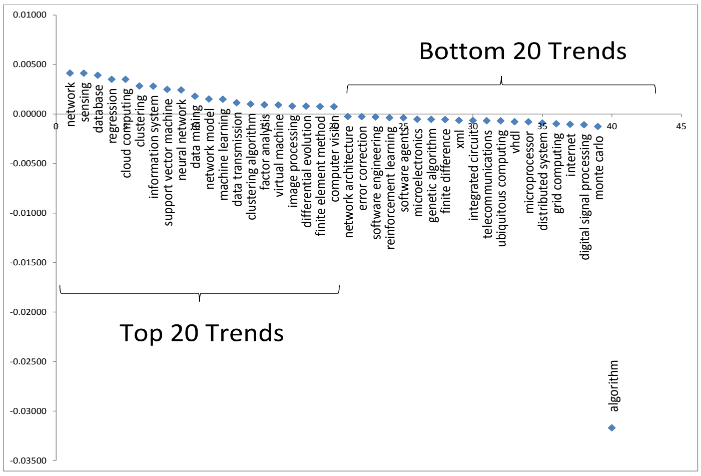


Figure 11: The top and bottom 20 trends 2006 - 2010 from the IEEE dataset.

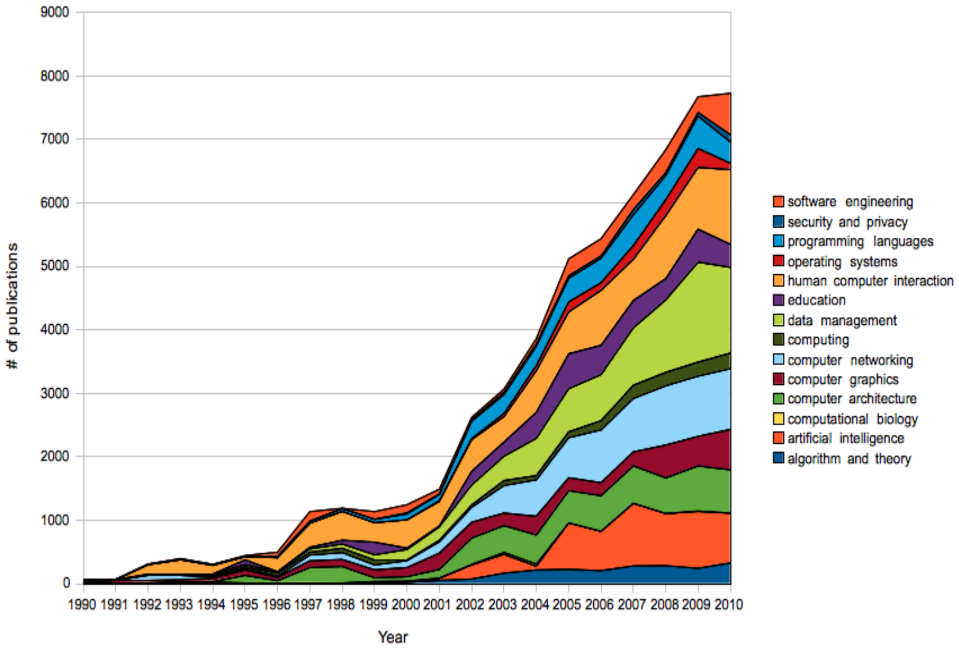


Figure 12: A landscape of Computer Science research fields 1990 - 2010 based on the raw number (frequencies) of publications for each keyword each year for the ACM dataset.

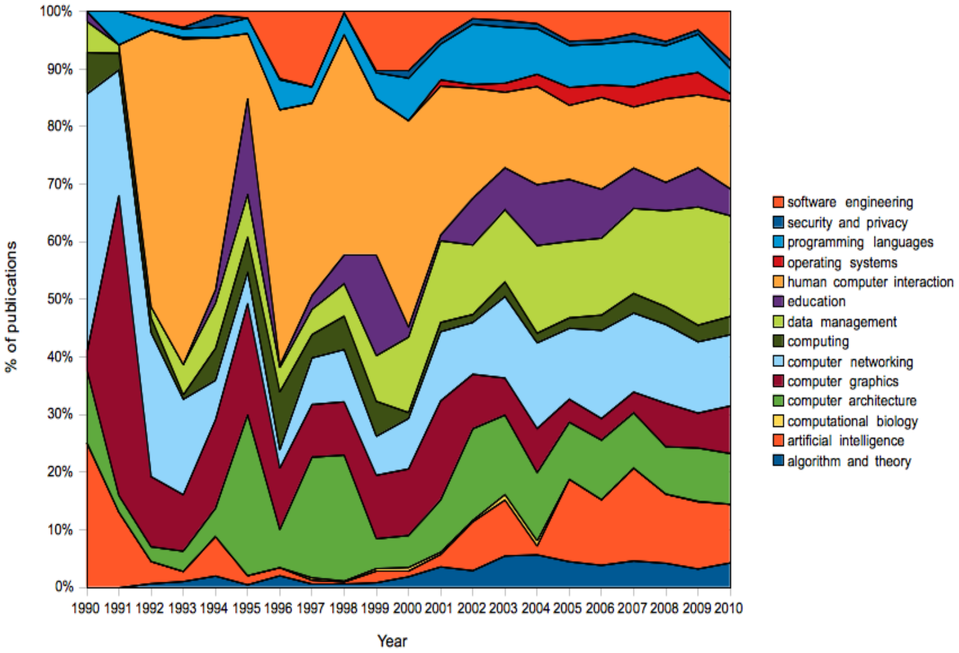


Figure 13: A landscape of Computer Science research fields from 1990 to 2010 based on the percentage of publications for each keyword each year for the ACM dataset.

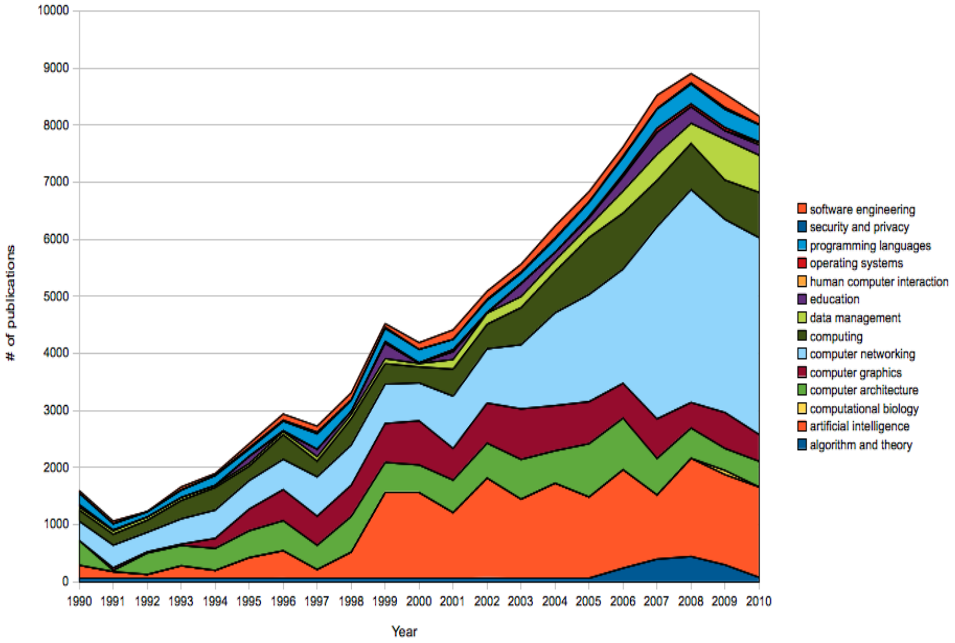


Figure 14: A landscape of Computer Science research fields from 1990 to 2010 based on the raw number (frequencies) of publications for each keyword each year for the IEEE dataset.

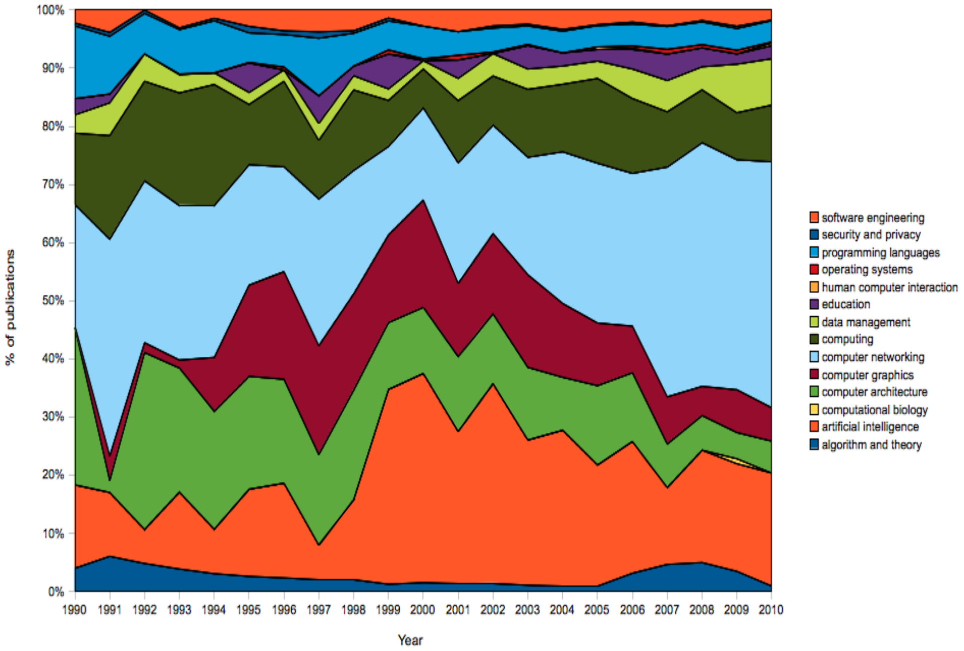


Figure 15: A landscape of Computer Science research fields from 1990 to 2010 based on the percentage of publications for each keyword each year for the IEEE dataset

algorithm as a node greatly reduced the degree of separation between other research topics and created a central node dominating other research topics.

Form 1990 to 2010, algorithm, database and neural network were the most frequent CS research topics. 311 other CS research topics have been mentioned along with algorithm at least once in the past 21 years. 78 of those are persistent (i.e., they co-appear with algorithm every year from 1990 to 2010). Out of 408 CS research topics, 286 have been mentioned with database but only 32 of them are persistent topics. 254 topics had appeared with neural network, but only 39 were persistent. The top five persistent topics for database are relational database, distributed database, database management, query language, and database design, while for neural network, they are pattern recognition, regression, supervised learning, reinforcement learning, and robotics. Besides the three most frequent topics, 11 others had persistent connections with multiple research topics every year $1990 - 2010$. Those are programming language, artificial intelligence, clustering, image processing, computer vision, network, distributed system, pattern recognition, robotics, software engineering, and integrated circuit. Also during $1990 - 2010$, 87 other research topics, such as image analysis, data transmission, and operating system are linked every year with up to three of the mentioned 14 topics.

In ACM networks using author-defined keywords, no persistent link appeared during $1990-2010$. This reinforces the earlier message that while a certain research topic may be important enough to be mentioned in the abstract, it may not represent the article's key research contributions. Another example of lack of link persistence is the neural network node in both IEEE and ACM networks. In IEEE networks, neural network is listed as a central node, a node with the highest total weight of its edges, almost every year. Yet in ACM networks, it never achieved this status. This is also the case with algorithm and database topics. In early 1990s, user interface, scheduling and multimedia were research topics that were connected to many CS research fields. In late 1990s, such interests shifted to world wide web, information retrieval, and computer supported cooperative work. Throughout the 2000s, the areas most connected to others were design, usability, and security. The mid 2000s saw strong interest in sensor network and later in wireless sensor network.

We performed clustering on the yearly network of keywords in the ACM dataset in which a keyword can appear in multiple clusters. Using the clusters, we measured the similarity between keywords k and a as

$$\frac{\text{Number of clusters with } a \text{ and } k}{\text{Number of clusters with } a}$$

In combination with network connectivity, we found a list of terms clustered together between 2006 and 2010, but have not been connected in at least 1% of documents [28]. We examined the top ten frequent words at various degree of separation. The results are shown in Tables 11, 12, and 13. From 2006 to 2010, simulation had been clustered with many keywords in database research such as data integration, data warehouse, and relational database. Yet these words were either not used, or rarely used, by the authors to describe their research in simulation. Instead, simulation was clustered with information retrieval, feature selection, and filtering. It was also clustered with various other topics related to data mining, machine learning, and artificial intelligence, but it was not used directly to describe the same research project often enough. Data mining has rarely been used to describe the research related to mobile networks and its related research topics.

5.5 Researchers in Computer Science

We used the cSpade sequence mining algorithm [28] to analyze sequences of publications in the same major research category by the same author. We required at most a one year gap in publication dates and appearance in at least 1% of documents. We recorded the maximum length of publication sequences in the same category. We measured the percentage of change in the number of publications of a given author after the first year in each category. From all the authors whose publications were in the same categories, we calculated the half-life time (the time it took for the number of authors who continued publishing papers in the category to reduce by half). For the first analysis, we used the ACM CCS to identify major research categories as reported in Table 14. Next, we performed the same analysis using the lists of conferences under six Computer Science categories listed in the first column of Table 16. Both Tables 14 and 16 show that most of the time the researchers published their article in an category and then quickly dropped this category. Yet, the rates of publication growth differed in each category.

From Table 14, the results indicate a relatively short half-life time as well as a high first year drop rate, especially for computer application, computing milieu, and data keywords, indicating that authors in these categories either became briefly involved in multiple research topics, or briefly collaborated with someone else from these categories. The researchers in computer systems organization, computing methodologies, and information systems tend to remain active in these categories for a longer time. Under ACM CCS major categories, data category included data structures, data storage representation, data encryption, coding and information theory, and files. Even if we increased the gap between publication to at most four years, there was still as high as 69% drop rate after the first publication, making data one of the rarest category for an author to continue to publish their work in. From Table 16, the data indicates that it is hard for researchers to be able to publish in Artificial Intelligence and Programming Language year after year, which is not the case in Human Computer Interaction. Even though the research took longer in Artificial Intelligence, the researchers working in this category remain active in it the longest, followed by researchers in human computer interaction category.

Note that while researchers can continue to publish in one area for a long time, the area itself evolves and may cover different topics in different time periods as demonstrated above. For example, HCI focused mainly on interaction design, visual design, and computer-supported cooperative work in 1990s, while it covered augmented reality, computer vision, human factor, and ubiquitous computing in early 2000s, to finally shift to social media, learning, computer-mediated communications, and tangible user interface in late 2000s. Also, an author may publish a paper in a different conference not listed in Wikipedia but the same pattern is observed in data in Tables 16, 17, 18, and 19. Although such data may be incomplete, they do show similar trend as those in Tables 14 and 14, where we used the pre-defined classification system, where each paper collected from ACM Digital library must be listed under.

To investigate further, we selected four prominent CS researchers, analyzed their publications using our approach and discussed the results with them. Prof. Jack Dongarra of the University of Tennessee, Knoxville, is renowned for developing high performance linear algebra software packages for various systems, yet his interests have evolved over time. In 1980s, he worked on parallel algorithms for linear equation routines and linear algebra subprograms. In early 1990s, he focused on parallel solutions for eigenvalue problems and numerical software libraries for high performance systems. From late 1990s to the 2000s, he worked on high performance linear algebra packages for multi-core systems. More recently, he has also focused on performance of grid computing. Overall, his research interests continuously evolve in response to challenges created by new computer technologies. Another researcher in this area, Prof. Francis Berman of Rensselaer Polytechnic Institute, Troy, NY, characterized her work in 1980s as "top-down mathematical modeling" of mapping and scheduling problems. In early 1990s, her papers used such keywords as data-driven, performance, and algorithms. From late 1990s to mid-2000s, she focused on grid computing from a "bottom up" perspective: application-level scheduling/rescheduling, job distribution, and performance. She described this evolution as a broadening and branching approach. Over the last decade she has made a major shift to large scale cyber-infrastructure and data preservation [1].

In the early 1990s, Prof. George Cybenko of Darthmouth College, Hanover, NH, studied the HPC systems and classification by neural networks. In the late 1990s, his focus shifted to mobile agents, mobile networks, and simulations. In early 2000s, he worked on target tracking, analyzing data, extracting information from web, and wireless networks. Over the past 10 years, he has investigated privacy and security issues, including cyber-security. Prof. Cybenko commented that he investigates each subject "in 5 year (more or less) phases" and then he "discovers *open field* often related to previous work." One exception was a major shift in 1992 related to moving from one university to another. As a final example, Prof. James A. Hendler of Rensselaer Polytechnic Institute, Troy, NY, has worked in Artificial Intelligence since the late 1980s. His major shift was from planning and web intelligence to semantic web. From late 1980s to early 1990s, his work focused on planning in AI, and later on agents, real-time systems, and web technology. In the 2000s, he mainly focused on semantic web and most recently also on large data and social networks.

Overall, faculty research interests typically evolve every five to 10 years by broadening the scope and branching into new applications, as well as responsing to technological innovations. Less frequently, usually once in a career, there is a major shift to a new area.

[1] However, "cyber-infrastructure" and "data preservation" did not show up as her keywords because the relevant publications are too new to be in our databases.

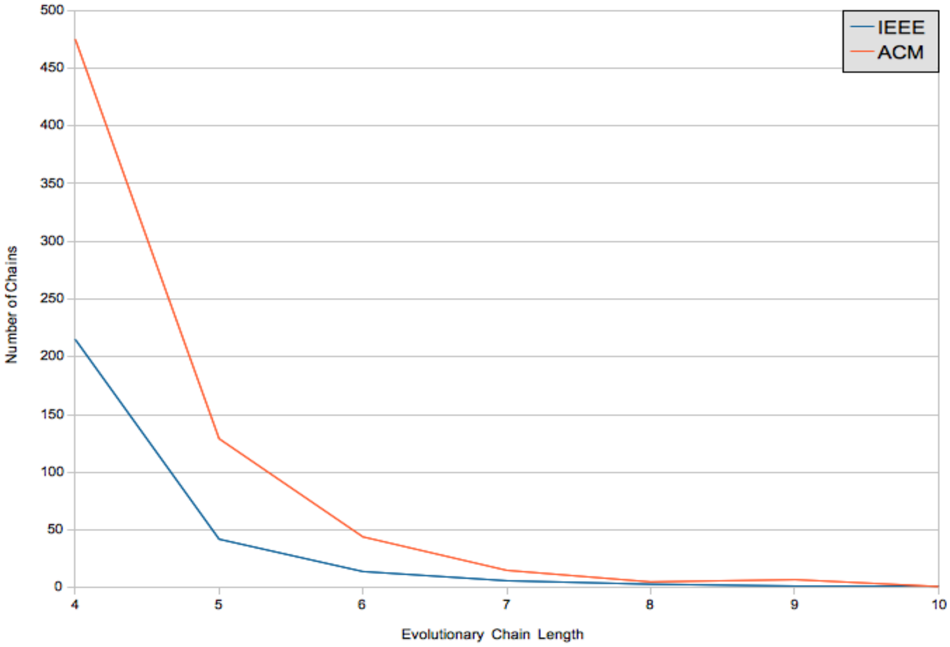


Figure 16: Distribution of the length of evolutionary chains showing number of years a slowly evolving research community remains continuously active based on the ACM and IEEE datasets.

5.6 Communities of Researchers in Computer Science

Using the framework for analyzing the evolution of social communities developed by [10], we tracked the evolution of CS researcher communities by searching for overlapping communities over consecutive time-periods. We used the networks of authors represented as a bipartite graph in which each node representing a paper has edges to all nodes representing this paper's authors. Specifically, if an author wrote a paper, then there is an edge between the author and the paper. The results are shown in Table 20 and Figure 16. The figure plots the number of communities that survived from one year to another in the ACM and IEEE datasets. The table shows the average evolutionary chain length, the average cluster size, the average size of intersections of two to four consecutive clusters, and the average relative density. It is measured as the combined weight of all edges with both endpoints in the cluster divided by the combined weight of all edges with at least one endpoint in the cluster. The recovered clusters had high average density of 0.8 for both datasets. The average length of the evolutionary chain is 4.5 years, while there are about two core researchers in each cluster. This is consistent with the typical university team consisting of one or two stable faculty and three to five graduate students and postdocs that join and leave continuously. Every four years or so, only a few stable researchers are left from the original research group.

6 Concluding Remarks

Computer Science is a large and ever changing research discipline. A majority of the publications mention the keyword algorithms, which is not surprising. However, interestingly, most abstracts mention one or more topics related to database, neural networks, and Internet. The data also showed that the world wide web has become a very attractive source of data and application testbeds. Since its creation, it has attracted various researchers working on data mining, information retrieval, cloud computing, and networks. Most of the research related to Internet has been done since 2000, even though its concept was introduced shortly after the standardization of TCP/IP protocol suite in the early 1980s. Web pages evolved from simple text written in mark-up languages such as HTML and XML to semantic web, where ontologies have been one of the key components for information retrieval by both humans and machines.

While the overall trends give us a clear picture of which direction each topic is taking, the fraction of publications on each topic oscillates from year to year to the point that the direction of change in this fraction in one year is reversed in the subsequent year. The same is true for the number of grants awarded for each topic in each year. Since novelty is highly prized in publications and grant applications, authors tend to stress novel aspects of their work in abstracts and keywords, contributing to the observed pattern. We also found a strong indication of money preceding research, because if a research topic burst in terms of NSF grants first, it is likely to burst in publications within a few years. The opposite pattern is at least twice less frequent. The data also indicates that while funding is not the key in the initial growth in a CS research topic, it is essential for maintaining the research momentum.

Looking from the researcher side, we can see that most authors only manage to get publication in each field at most once a year. Moreover, the authors tend to publish their work in the same major research category for at most a few years. Only a small fraction of researchers continues to publish in the same field year after year for a long time. This agrees well with the model of an academic research team in which permanent faculty represent only a small fraction of the entire team of faculty, students, and postdocs, with the latter changing topics after leaving a team. Moreover, a faculty member is often active in more than one area. Finally, since novelty is highly valued in publications, authors tend to pursue new directions in their research, which is reflected in a paper's abstract and keywords, further contributing to the observed pattern.

Acknowledgment

The authors thank Fran Berman and Jim Hendler of RPI, George Cybenko of Dartmouth, and Jack Dongarra of UTK for discussions on the evolution of their research interests. This research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army

Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

References

[1] ACM Digital Library, `http://dl.acm.org`

[2] Rakesh Agrawal, Anastasia Ailamaki, Philip A. Bernstein, Eric A. Brewer, Michael J. Carey, Surajit Chaudhuri, AnHai Doan, Daniela Florescu, Michael J. Franklin, Hector Garcia-Molina, Johannes Gehrke, Le Gruenwald, Laura M. Haas, Alon Y. Halevy, Joseph M. Hellerstein, Yannis E. Ioannidis, Hank F. Korth, Donald Kossmann, Samuel Madden, Roger Magoulas, Beng Chin Ooi, Tim O'Reilly, Raghu Ramakrishnan, Sunita Sarawagi, Michael Stonebraker, Alexander S. Szalay, and Gerhard Weikum, *The Claremont Report on Database Research*, Communications of the ACM, vol. 52(6):56–65, Jun. 2009.

[3] Hsinchun Chen, *AI and Global Science and Technology Assessment*, IEEE Intelligent Systems, vol 24(4): 68 – 88, Jul.-Aug. 2009.

[4] J. M. Cohoon, S. Nigai, and J. Kaye, *Gender and Computing Conference Papers*, Communications of the ACM, vol. 54(8):72–80, Aug. 2011.

[5] DBLP XML records, `http://dblp.uni-trier.de/xml/`

[6] D. Edler and M. Rosvall (2010), *The Map Generator software package*, online at http://www.mapequation.org.

[7] Eugene F. Fama, *The Behavior of Stock-Market Prices*, Journal of Business, vol. 38(1):34–105, Jan. 1965.

[8] IEEE Xplore, `http://ieeexplore.ieee.org/Xplore/`

[9] M. Goldberg, S. Kelly, M. Magdon-Ismail, K. Mertsalov, and W. A. Wallace, *Overlapping Communities in Social Networks*,

[10] M. Goldberg, M. Magdon-Ismail, S. Nambirajan, and J. Thompson, *Tracking and Predicting Evolution of Social Communities*, 3rd IEEE International Conference on Social Computing, Boston, MA, October 2011.

[11] Mary Hall, David Padua, and Keshav Pingali, *Compiler Research: The Next 50 Years*, Communications of the ACM, vol. 52(2): 60 – 67, Feb. 2009.

[12] Jim Hendler, and Time Berners-Lee, *From the Semantic Web to social machines: A research challenge for AI on the World Wide Web*, Artificial Intelligence (2009), doi:10.1016/j.artint.2009.11.010.

[13] A. Hoonlor, B. K. Szymanski, M. J. Zaki, and V. Chaoji, *Document clustering with bursty information*, Computing and Informatics, 31(6): 1533–1555, 2012.

[14] A. Lancichinetti, S. Fortunato, and J. Kertész, *Detecting the overlapping and hierarchical community structure in complex networks*, New Journal of Physics, vol. 11 (2009) 033015, [Online] Available: `http://www.njp.org/doi:10.1088/1367-2630/11/3/033015`. Date Last Accessed Jul. 28, 2011.

[15] T. Lappas, B. Arai, M. Platakis, D. Kotsakos, and D. Gunopulos, *On Burstiness-Aware Search for Document Sequences*, In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, Paris, France, 2009, pp. 477–486.

[16] Michael Ley, *DBLP - Some Lessons Learned*, VLDB 2009, August 24-28, 2009, Lyon, France.

[17] James Moody, *The Structure of a Social Science Collaboration Network: Disciplinary Cohesion from 1963 to 1999*, American Sociological Review, vol. 69: 213–238, Apr. 2004.

[18] Ronald P. Neilson, *High-Resolution Climatic Analysis and Southwest Biogeography*, Science, vol. 232(4746): 27–34, Apr. 1986.

[19] Alan L. Porter, and Ismael Rafols, *Is science becoming more interdisciplinary? Measuring and mapping six research fields over time*, Scientometrics, vol. 81(3): 719–745, 2009.

[20] Mark H Reacher, Anita Shah, David M Livermore, Martin C J Wale, Catriona Graham, Alan P Johnson, Hilary Heine, Marjorie A Monnickendam, Keith F Barker, Dorothy James, and Robert C George, *Bacteraemia and antibiotic resistance of its pathogens reported in England and Wales between 1990 and 1998: trend analysis*, BMJ, vol. 320(7229): 213–216, Jan. 2000.

[21] Thomson Reuters, *Web of Science*, [Online] available: `http://thomsonreuters.com/products\_services/science/science\_products/a-z/web\_of\_science/`, Date Last Accessed 07/27/2011.

[22] M. Rosvall and C. Bergstrom, *Maps of Information Flow Reveal Community Structure in Complex Networks*, PNAS 105, 1118 (2008).

[23] M. Rosvall and C. Bergstrom, *Mapping Change in Large Networks*, PLoS One, vol. 5(1): e8694, Jan. 2010.

[24] W. L. Ruzzo and M. Tompa, *A linear time algorithm for finding all maximal scoring subsequences*, In Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology, 1999.

[25] Mazeiar Salehie, and Ladan Tahvildari, *Self-Adaptive Software: Landscape and Research Challenges*, ACM Transactions on Autonomous and Adaptive Systems, vol. 4(2), article 14, 2009.

[26] Y. Sun, J. Tang, J. Han, M. Gupta, and B. Zhao, *Community evolution detection in dynamic heterogeneous information networks*, In Proceedings 2010 KDD Workshop on Mining and Learning with Graphs, 2010.

[27] X. Wang and A. McCallum, *Topics over Time: A Non-Markov Continuous-Time Model of Topical Trends*, Conference on Knowledge Discovery and Data Mining (KDD) 2006.

[28] Mohammed J. Zaki, *Sequences Mining in Categorical Domains: Incorporating Constraints*, In 9th ACM International Conference on Information and Knowledge Management. Nov 2000.

[29] *List of Computer Science Conference*, [Online] available: `http://en.wikipedia.org/wiki/List\_of\_computer\_science\_conferences`

Supplementary Materials

A Research Topic Tracked In IEEE Dataset

A.1 A

abstract state machine, adaptive system, algorithm, ambient intelligence, analytical database, anti virus software, applied statistics, artificial immune, artificial intelligence, artificial life, assembly language, association rule, automata theory, automated deduction, automated theorem proving, autonomous system, axiomatic semantics.

A.2 B

bayesian network, behavior based robotic, behavioral experiment, binary decision diagram, bioinformatics, bionics, boolean algebra, brain imaging

A.3 C

categorical sequence, chemical computer, cholesky decomposition, classification algorithm, cloud computing, cluster analysis, cluster computing, clustering, clustering algorithm, code generation, coding theory, cognitive linguistics, cognitive robotic, cognitive science, combinational logic, communication network, comparative genomics, competitive learning, compiler construction, compiler design, compiler technology, compiler, computability theory, computation complexity theory, computation theory, computational biology, computational chemistry, computational evolutionary biology, computational fluid dynamics, computational geometry, computational linguistics, computational mathematics, computational modeling, computational neuroscience, computational physics, computational science, computational theory, computer algorithm, computer architecture, computer arithmetic, computer cluster, computer engineering, computer graphics, computer insecurity, computer multitasking, computer network, computer programming, computer security, computer vision, computer visualization, concurrency, concurrent computing, constraint database, constraint logic programming, constraint satisfaction problem, content based image retrieval, context aware pervasive systems, context switch, cooperative multitasking, cryptanalysis, cryptographic primitive, cryptography, cryptosystem, cultural algorithm, cup design

A.4 D

data analysis, data compression, data hierarchy, data intervention, data mining, data mining agent, data prefetching, data security, data stream management system, data structures, data transmission, data warehouse, database, database centric architecture, database design, database management system, database model, database query language, dataflow architecture, datapath, decision tree, declarative language, declarative programming, denotational semantics, deterministic automata, differential evolution, digital communication, digital image processing, digital logic, digital organism, digital signal processing, directory service, distributed artificial intelligence, distributed computing, distributed data management, distributed database, distributed file system, distributed memory system, distributed networking, distributed processing, distributed system, document management system, document oriented database, drug discovery, dynamic semantics

A.5 E

eigenvalue decomposition, end-user database, ensemble learning, entity relationship, error correction, error management method, evolutionary computation, explanation based learning, external database, extrapolation

A.6 F

facial animation, factor analysis, finite difference, finite element method, finite state machine, finite volume method, firewall, flat model, flow networks, formal method, formal semantics, formal verification, functional analysis, functional programming, fuzzy logic

A.7 G

garbage collection, gaussian elimination, gene expression, gene finding, genetic algorithm, genetic programming, genome annotation, genome assembly, geometric modeling, gram schmidt process, graph algorithm, graph database, graph drawing, graph search algorithm, graph theory, grid computing

A.8 H

hardware architecture, hardware description language, hardware verification, harmony search, harvard architecture, heap management, heterogeneous database system, hierarchical model, HTML, human centered computing, human computer interaction, hypermedia database

A.9 I

image analysis, image processing, imperative programming, inductive logic programming, information extraction, information retrieval, information science, information system, information theory, instruction level parallelism, integrated circuit, interface agent, internet, internet network, interpolation, intrusion detection

A.10 K

karnaugh maps, knowledge discovery, knowledge representation, knowledge spaces

A.11 L

lagrange multiplier, lambda calculus, learnable evolution model, learning classifier system, linear bounded automata, linear programming, local area network, logic families, logic gate, logic minimization, logic programming, logic program construction, logic simulation, logical effort, longest path problem

A.12 M

machine learning, machine vision, markup languages, matrix decomposition, memory hierarchies, memory management, metaprogramming, microarchitecture, microcontroller, microelectronics, microkernel, microprocessor, mimd multiprocessing, minimum spanning tree, misd multiprocessing, mobile computing, monte carlo, motion planning, motor control, multi-core computing, multi agent, multicore computing, multiprocessing, multithreaded programming

A.13 N

named entity recognition, natotechnology, natural language processing, network, network architecture, network model, network theory, neural computation, neural network, neurobiological, nondeterministic automata, numerical analysis, numerical integration, numerical method, numerical ordinary differential equation, numerical partial differential equation, numerical recipes

A.14 O

object database model, object model, object oriented programming, object recognition, object relational model, ontology language, operating system, operational database, operational semantics, optical character recognition, optical flow estimation

A.15 P

parallel computing, parallel processing, parallel systems, pattern recognition, peer to peer network, planning scheduling, pose estimation, predicting sequences, predictive analysis, preemptive multitasking, principal component analysis, procedural programming, process management, processor symmetry, profiling practices, program analysis, programming language, protein expression analysis, protein interaction, protein structure alignment, protein structure prediction, public key cryptography, public key encryption, pushdown automata

A.16 Q

quantum computer, quasi monte carlo, query language, query optimization

A.17 R

real time database, real valued sequence, reference database, regression, regular expression, regulation analysis, reinforcement learning, relational database, relational engine, relational model, robotics, root finding algorithm, routing algorithm, run book automation

A.18 S

scalar processor, secure coding, secure operating system, security architecture, self organization, sensing, sensor network, sentient computing, sequence alignment, sequence analysis, sequential logic, shortest path problem, signal transmission, simd multiprocessing, simplex method, singular value decomposition, sisd multiprocessing, social engineering, software agents, software engineering, software process management, software semantic, spatial data mining, spectral image compression, sql, sql engine, standard library, state space search, static semantic, storage engine, stream processing, strongly connected components, structured data analysis, supervised learning, support vector machine, symbolic numerica computation, symmetric key cryptography, system architecture

A.19 T

task computing, telecommunications, temporal data mining, text mining, texture mapping, theoretical linguistic, transaction engine, transparent latch, traveling salesman problem, truth table, turing machine, type safety, type system, type theory

A.20 U

ubiquitous computing, unsupervised learning

A.21 V

vector processor, very large database, vhdl, virtual file system, virtual machine, virtual memory, virtual reality, volumetric visualization

A.22 W

wearable computer, web mining, wide area network, wireless network

A.23 X

XML

B Research Topic Tracked In ACM Dataset

B.1 A-C

awareness, bioinformatics, children, classification, cloud computing, clustering, code generation, collaboration, collaborative filtering, communication, compiler, complexity, component, compression, computer science education, computer vision, computer-mediated communication, concurrency, congestion control, constraint, context, context-awareness, coordination, creativity, cryptography, cs1, cscw, curriculum

B.2 D-E

data mining, data stream, data structure, database, debugging, design, design pattern, digital libraries, distributed algorithm, distributed computing, distributed system, dynamic programming, e-commerce, e-government, e-learning, education, embedded system, emotion, energy efficiency, ethnography, evaluation, evolutionary algorithm, evolutionary computation, eye tracking

B.3 F-I

fault tolerance, feature selection, formal method, fpga, framework, game, game theory, genetic algorithm, genetic programming, gesture, gi, grid computing, groupware, haptic, hci, human factor, human-computer interaction, human-robot interaction, image processing, image retrieval, indexing, information extraction, information retrieval, information visualization, input device, interaction, interaction design, interaction technique, interface, internet, interoperability, intrusion detection

B.4 J-M

java, knowledge management, learning, load balancing, localization, low power, machine learning, management, manet, measurement, metadata, metric, middleware, mobile, mobile ad hoc network, mobile computing, mobile device, mobile phone, mobility, model, model checking, modeling, monitoring, multi-agent system, multicast, multimedia

B.5 N-P

natural language processing, navigation, network, neural network, ontologies, ontology, operating system, optimization, p2p, parallel programming, participatory design, pattern, pedagogy, peer-to-peer, perception, performance, performance analysis, performance evaluation, personalization, pervasive computing, placement, prediction, privacy, program analysis, programming, protocol, prototyping

B.6 Q-R

qos, quality of service, query processing, ranking, real-time, real-time system, recommender system, refactoring, reinforcement learning, relevance feedback, reliability, rfid, robotic, routing, scalability

B.7 S-T

scheduling, search, search engine, security, semantic, semantic web, sensor, sensor network, simulation, social network, software architecture, software engineering, software testing, speech recognition, static analysis, support vector machine, synchronization, tangible interface, tangible user interface, tcp, testing, text mining, training, trust

B.8 U-Z

ubiquitous computing, uml, usability, user experience, user interface, user interface design, user studies, user study, user-centered design, verification, video, virtual environment, virtual machine, virtual reality, virtualization, visualization, vlsi, web, web 2.0, web search, web service, wiki, wikipedia, wireless, wireless network, wireless sensor network, workflow, world wide web, www, XML

C NSF Dataset

We collected the NSF data from all the awards from the dicretorates of NSF, listed below.

1. Division of Computer and Communication Foundation (CCF)
2. Division of Computer and Network Systems (CNS)
3. Division of Information Systems (DIS)
4. Division of Electrical, Communications and Cyber Systems (ECCS)
5. Division of Information and Intelligent Systems (IIS)
6. National Center for Science and Engineering Statistics (NCSE)
7. Division of Experimental and Integrative Activities (EIA)
8. Directorate for Computer and Information Science and Engineering (CSE).

D ACM Computing Classification System

The listed of ACM Computing Classification System that we used to extract data from ACM. We ignored general literature category because it consists of non-research-related topics such as biography, introduction and reference.

1. hardware
2. computer systems organization

3. software
4. data
5. theory of computation
6. mathematics of computing
7. information systems
8. computing methodologies
9. computer applications
10. computing milieu

E The list of Computer Science conferences from [29]

The list of major computer research topics and their corresponding conferences are listed in the table below. Note that Computing included research in concurrent computing, distributed computing, and parallel computing.

Table 3: Top 25 Keywords in papers included in the ACM and IEEE datasets.

#	IEEE Dataset				ACM Dataset			
	Word	DF	Word	TFIDF	Word	DF	Word	TFIDF
1	algorithm	142540	algorithm	144941	genetic algorithm	2487	security	2403
2	neural network	40915	network	113436	simulation	2420	scheduling	2401
3	database	23934	database	57797	security	2324	data mining	2346
4	internet	22563	internet	51626	neural network	2255	optimization	2221
5	clustering	15685	sensing	36692	data mining	2188	simulation	2126
6	image processing	10826	clustering	36214	scheduling	2077	information retrieval	1873
7	monte carlo	10088	regression	27639	optimization	2023	clustering	1765
8	information system	9970	interpolation	25231	algorithm	1808	comp. complexity	1619
9	network	9725	microprocessor	18679	clustering	1549	stability	1625
10	sensing	9699	telecommunications	16832	information retrieval	1542	privacy	1549
11	regression	9090	XML	16770	wireless sensor network	1534	visualization	1511
12	fuzzy logic	8169	robotics	16290	stability	1419	XML	1490
13	sensor network	8073	microcontroller	11938	sensor network	1417	machine learning	1475
14	support vector machine	7963	vhdl	11524	distributed system	1333	evaluation	1474
15	interpolation	7837	cryptography	9429	web service	1324	routing	1468
16	data mining	7070	concurrency	9041	performance eval.	1294	performance eval.	1431
17	distributed system	5671	microelectronics	8666	visualization	1285	internet	1413
18	pattern recognition	5623	compiler	7196	comp. complexity	1285	classification	1368
19	genetic algorithm	5474	bioinformatics	5317	internet	1278	software eng.	1328
20	data transmission	5362	extrapolation	5027	XML	1270	performance	1295
21	digital signal processing	5216	HTML	4588	privacy	1244	fault tolerance	1288
22	XML	5161	datapath	4380	evaluation	1235	parallel processing	1268
23	software engineering	5085	sql	3953	approx. alg.	1231	genetic algorithm	1257
24	microprocessor	4963	firewall	2575	classification	1176	02.30.yy	1227
25	telecommunications	4849	microarchitecture	2487	performance	1173	multimedia	1207

Table 4: The top 10 most frequent words that became bursty in the NSF dataset before they did so in the ACM dataset.

Keywords	NSF	ACM
genetic algorithms	1996	2003
simulation	2000	2003
security	2001	2003
neural networks	1990	2002
data mining	1999	2002
scheduling	1992	2002
optimization	1997	2004
clustering	1992	2003
information retrieval	1999	2002
wireless sensor network	2004	2006

Table 5: The top 10 most frequent words that became bursty in the NSF dataset before they did so in the IEEE dataset.

Keywords	NSF	IEEE	NSF-L
algorithm	1990	2002	2001
neural network	1990	2006	2005
database	1997	2004	2004
clustering	1992	2004	2002
image processing	1994	2006	2006
monte carlo	1995	2003	2002
information system	1991	2006	2006
network	2002	2004	2004
sensing	2002	2004	2004
regression	1993	2005	2003

Table 6: The top 10 bursty correlated words, listed in the order of the bursty ranking, in the burstiest period of the 10 most frequent words for the ACM dataset.

Keywords	BP	Top 10 Bursty keywords
security	2000 - 2010	wireless sensor networks, routing, sensor networks, web services, usability, grid computing, wireless networks, peer-to-peer, static analysis, rfid
simulation	1996 - 2010	scheduling, optimization, visualization, wireless sensor networks, sensor networks, qos, wireless networks, ad hoc networks, analysis, validation
data mining	2000 - 2010	genetic algorithms, privacy, bioinformatics, feature selection, time series, web mining, clustering, security, pattern recognition, text mining
scheduling	1990 - 1991	real-time systems, parallel processing
optimization	1992 - 1999	neural networks
neural networks	1992 - 2001	learning, pattern recognition, optimization, fuzzy logic, stability
clustering	2002 - 2010	wireless sensor networks, visualization, data mining, classification, ad hoc networks, genetic algorithms, text mining, neural networks, IR
IR	1999 - 2010	XML, semantic web, ontology, peer-to-peer text mining, information extraction, web search, query expansion, evaluation, search engine
stability	1991 - 1998	robust control, adaptive control, nonlinear systems, robustness, bifurcation
genetic algorithms	1995 - 2009	scheduling, fuzzy logic, heuristics, clustering multi-objective optimization, simulated annealing, neural networks, optimization, data mining

Table 7: The top 10 bursty correlated tracked topics, listed in the order of the bursty ranking, in the burstiest period of the 10 most frequent tracked topics in the IEEE dataset.

Keywords	BP	Top 10 Bursty keywords
algorithm	1990 - 2004	logic minimization, logic simulation, distributed processing, facial animation, virtual memory, sequential logic, truth table, concurrency, digital logic, object oriented programming
neural network	1990 - 1999	parallel systems, computer architecture, data compression, constraint satisfaction problem, traveling salesman problem, finite difference, object recognition, distributed processing, optical character recognition, competitive learning
database	1990 - 1993	logic programming, integrated circuit, entity relationship, local area network, concurrency, parallel processing, operating system, object oriented programming, type system, programming language
internet	1998 - 2009	multi agent, computer security, hardware architecture, association rule, XML, security architecture, concurrency, knowledge discovery, algorithm, grid computing
clustering	2003 - 2010	differential evolution, protein interaction, sensor network, artificial immune, bioinformatics, spatial data mining, support vector machine, intrusion detection, genetic programming, gene expression
image processing	1992 - 1997	data compression, data structures, network parallel processing
monte carlo	2000 - 2010	support vector machine, sensor network, wireless network, computer vision, bayesian network, robotics, genetic algorithm, network, machine learning, sensing
information system	2007 - 2010	cloud computing, sensor network, cryptography, data transmission, process management, support vector machine, data security, bioinformatics, ubiquitous computing, network model
network	2006 - 2010	network theory, sensor network, data mining, principal component analysis, data analysis, clustering algorithm graph theory, data transmission, virtual machine, regression
sensing	2006 - 2010	wireless network, network model, sensor network, microcontroller , support vector machine, data transmission, principal component analysis, decision tree, monte carlo, data mining

Table 8: The top 5 co-reference words, listed in the order of the bursty ranking, in the burstiest period of the 10 most frequent words in the ACM dataset.

Keywords	BP	Top 10 Bursty keywords
security	2000 - 2010	privacy, authentication, cryptography, access control, trust
simulation	1996 - 2010	modeling, wireless networks, performance evaluation optimization, wireless sensor networks
data mining	2000 - 2010	clustering, association rules, classification, machine learning, knowledge discovery
scheduling	1990 - 1991	real-time systems, parallel processing, performance evaluation, load balancing, partitioning
optimization	1992 - 1999	genetic algorithms, neural networks, simulation, scheduling, algorithms
neural networks	1992 - 2001	fuzzy logic, genetic algorithms, learning, pattern recognition, machine learning
clustering	2002 - 2010	data mining, classification, visualization wireless sensor networks, genetic algorithms
IR	1999 - 2010	evaluation, natural language processing, machine learning, query expansion, text mining
stability	1991 - 1998	robustness, adaptive control, robust control convergence, nonlinear systems
genetic algorithms	1995 - 2009	optimization, neural networks, simulated annealing, heuristics, evolutionary computation

Table 9: The top top 5 co-reference tracked topics, listed in the order of the bursty ranking, in the burstiest period of the 10 most frequent tracked topics in the IEEE dataset.

Keywords	BP	Top 10 Bursty keywords
algorithm	1990 - 2004	neural network, clustering, database, image processing, genetic algorithm
neural network	1990 - 1999	algorithm, network model, pattern recognition, fuzzy logic, network architecture
database	1990 - 1993	algorithm, relational database, neural network, distributed database, concurrency
internet	1998 - 2009	algorithm, database, network, XML information system
clustering	2003 - 2010	algorithm, data mining, neural network, database, sensor network
image processing	1992 - 1997	algorithm, neural network, pattern recognition computer vision, digital image processing
monte carlo	2000 - 2010	algorithm, neural network, regression clustering, sensor network
information system	2007 - 2010	database, data mining, algorithm, internet, XML
network	2006 - 2010	neural network, algorithm, sensor network wireless network, network model
sensing	2006 - 2010	algorithm, information system, image processing, neural network, sensor network

Table 10: Trend Prediction.

Year	ACM	IEEE	NSF
2	12.06%	21.68%	36.79%
3	49.54%	55.94%	64.51%
4	65.86%	65.49%	73.61%
5	72.21%	69.79%	74.63%
6	76.54%	70.28%	77.61%

Table 11: Keywords which were clustered with the specified keywords every time for five years from 2006 to 2010

Keyword	Similar keywords
security	None
simulation	access control, annotation, aspect-oriented programming, awareness, cluster analysis, compression, computational geometry, computer vision, constrained optimization, content-based image retrieval, data compression, data integration, data stream, data warehouse, decomposition, design pattern, duality, eigenvalue, embedding, emotion, empirical study, entropy, error analysis, ethnography, eye tracking, feature extraction, feature selection, filtering, finite field, fixed point, functional programming, garbage collection, gesture, gp, graph algorithm, groupware, hypertext, image retrieval, image segmentation, indexing, information extraction, innovation, interaction technique, kalman filter, knowledge discovery, local search, low-power, memory, metadata, mimo, mobile, monte carlo simulation, music, natural language processing, open source, parallelism, particle swarm optimization, partitioning, pattern matching, pattern recognition, pda, personalization, planar graph, principal component analysis, program analysis, program transformation, query processing, random walk, randomized algorithm, ranking, rdf, regularization, relational database, search engine, self-organizing map, semidefinite programming, singular value decomposition, soa, software maintenance, software quality, stabilization, standard, static analysis, support vector machine, system identification, tangible user interface, text mining, tracking, triangulation, type system, user experience, user studies, wavelet transform, web application, web mining, web search, wiki
data Mining	abstract interpretation, accessibility, adaptation, adaptive control, admission control, analysis of algorithm, animation, aspect-oriented programming, assessment, atm, augmented reality, authentication, awareness, bluetooth, broadcast, broadcasting, c++, cache, cad, case study, children, cmo, code generation, compiler, component, computer architecture, computer graphics, computer science education, computer-mediated communication, concurrency, concurrency control, congestion control, connectivity, constrained optimization, control, convergence, coordination, creativity, cryptography, cs1, data structure, delay, diffusion, digital signature, dynamic, dynamic programming, eigenvalue, embedded system, embedding, emotion, encryption, energy, error analysis, ethnography, evolution, eye tracking, fairness, fault-tolerance, finite element, finite element method, finite field, fixed point, formal specification, formal verification, fpga, garbage collection, groupware, haptic, high-level synthesis, human factor, human-robot interaction, identification, image retrieval, implementation, innovation, interaction technique, interconnect, interconnection network, interface, interpolation, inverse problem, kalman filter, local search, localization, low power, low-power, lower bound, mac, manet, medium access control, memory, message passing, mimo, mobile, mobile ad hoc network, mobile agent, mobile communication, mobile device, mobile phone, modularity, mpi, multicast, multiprocessor, object-oriented, object-oriented programming, ofdm, operating system, optimal control, parallel programming, parameter estimation, participatory design, pda, pedagogy, planar graph, power, power management, preconditioning, pricing, process algebra, program analysis, program transformation, programming, programming language, prototyping, quality, rdf, real-time, refactoring, refinement, reflection, replication, requirement, requirements engineering, resource allocation, reuse, robotic, robust control, routing, routing protocol, semidefinite programming, service, shortest path, signal processing, soc, software testing, specification, stabilization, static analysis, synchronization, synthesis, system, tangible user interface, tcp, temporal logic, throughput, topology, tree, triangulation, type system, uml, user experience, user interface design, user studies, validation, virtual environment, virtual machine, virtualization, vlsi, voip, web application, wiki, wireless, wireless communication, wireless mesh network, wlan

Table 12: Keywords which were clustered with the specified keywords every time for five years from 2006 to 2010 (Cont.)

Keyword	Similar keywords
scheduling	duality, linear system, robust control, semidefinite programming, triangulation
optimization	boolean function, interconnection network, monte carlo simulation, wavelet transform
neural networks	None
clustering	abstract interpretation, abstraction, access control, adaptive control, animation, aspect-oriented programming, assessment, atm, augmented reality, benchmarking, bluetooth, boolean function, c++, cache, case study, children, cmo, composition, computer architecture, computer graphics, computer science education, computer-mediated communication, concurrency, concurrency control, congestion control, consistency, constrained optimization, control, cryptography, cs1, cscw, culture, curriculum, debugging, decision support system, design pattern, diffusion, digital signature, distance learning, education, emotion, encryption, energy, error analysis, estimation, ethnography, eye tracking, fairness, finite field, fixed point, formal method, formal verification, functional programming, game, gesture, gp, groupware, haptic, high-level synthesis, human factor, human-robot interaction, implementation, information security, innovation, intelligent agent, interaction technique, interface design, interoperability, inverse problem, java, kalman filter, knowledge representation, logic programming, low-power, message passing, methodology, mimo, mobile communication, mobile computing, model checking, monte carlo simulation, multiagent system, natural language processing, nonlinear programming, object-oriented programming, ofdm, optimal control, parallelism, participatory design, pattern matching, pedagogy, planar graph, planning, preconditioning, pricing, probability, process algebra, program analysis, program transformation, programming, programming language, protocol, prototyping, refinement, reflection, reinforcement learning, replication, requirements engineering, resource allocation, resource management, reuse, rfid, robotic, robust control, routing protocol, semidefinite programming, sensor, service, service-oriented architecture, shortest path, signal processing, soa, soc, software, software architecture, software metric, software quality, software testing, specification, speech recognition, stability, stabilization, standard, static analysis, supply chain management, synthesis, system, tangible user interface, tcp, technology, telecommunication, temporal logic, testing, type system, usability, user experience, user interface design, user studies, user-centered design, verification, video, voip, wireless, wireless communication, wlan, workflow
IR	abstract interpretation, abstraction, access control, ad hoc network, adaptive control, admission control, anomaly detection, anonymity, approximation, artificial neural network, aspect-oriented programming, association rule, authentication, awareness, benchmarking, bluetooth, boolean function, broadcast, broadcasting, c++, cache, caching, cad, cmo, code generation, collaborative learning, combinatorial optimization, compiler, complexity, component, composition, computer architecture, computer graphics, computer science education, concurrency, congestion control, connectivity, consistency, constrained optimization, convergence, correlation, creativity, cryptography, cs1, curriculum, debugging, decomposition, delay, diffusion, digital signature, distributed algorithm, duality, dynamic, dynamic programming, dynamical system, e-government, eigenvalue, embedded system, embedding, empirical study, encryption, energy, energy efficiency, error analysis, ethnography, evolutionary algorithm, fairness, fault-tolerance, finite element, finite element method, finite field, fixed point, forecasting,

Table 13: Keywords which were clustered with the specified keywords every time for five years from 2006 to 2010 (Cont.)

Keyword	Similar keywords
IR (Cont.)	formal method, formal specification, formal verification, fpga, framework, functional programming, game, game theory, global optimization, graph theory, groupware, haptic, heuristic, high-level synthesis, human-robot interaction, identification, image segmentation, implementation, information security, innovation, integer programming, interaction technique, interconnect, interconnection network, interpolation, intrusion detection, inverse problem, kalman filter, linear programming, linear system, localization, low power, low-power, mac, management, manet, medium access control, message passing, methodology, middleware, mimo, mobile ad hoc network, mobile communication, mobile phone, model checking, modeling, modularity, monte carlo simulation, mpi, multicast, network security, nonlinear programming, object-oriented programming, ofdm, online algorithm, optimal control, parallel, parallel programming, parallelism, participatory design, particle swarm optimization, partitioning, pattern recognition, pda, pedagogy, performance analysis, petri net, placement, planning, power, preconditioning, pricing, process algebra, program transformation, programming, programming language, project management, protocol, prototyping, randomized algorithm, real-time, refactoring, refinement, reflection, regularization, replication, requirements engineering, resource allocation, resource management, reverse engineering, robotic, robust control, routing protocol, sampling, scenario, semidefinite programming, sensitivity analysis, sensor, sensor network, service-oriented architecture, shortest path, signal processing, soc, software, software architecture, software development, software evolution, software quality, software testing, specification, stability, stabilization, supply chain management, survey, synchronization, synthesis, system identification, tcp, technology, telecommunication, temporal logic, testing, throughput, time series, tool, topology, tree, triangulation, type system, validation, virtual environment, virtual machine, virtualization, vlsi, voip, wavelet, web application, wiki, wireless communication, wireless mesh network, wlan, workflow
stability	None
genetic algorithms	None

Table 14: Statistic of publications on ACM Digital Library in each major categories listed in the ACM Computing Classification System.

CCS	1-year gap			2-year gap		
	1st DR	$T_{\frac{1}{2}}$	Max. CL	1st DR	$T_{\frac{1}{2}}$	Max. CL
hardware	66%	0.94	5	59%	1.28	7
comp. sys. organization	54%	1.22	8	46%	1.49	9
software	52%	1.15	7	43%	1.47	9
data	81%	0.48	3	75%	0.59	3
theory of computation	60%	0.90	6	50%	1.27	8
mathematics of computing	51%	1.06	7	41%	1.58	10
information systems	48%	1.32	8	40%	1.70	11
computing methodologies	41%	1.26	8	32%	1.66	11
computer applications	72%	0.61	4	63%	0.83	5
computing milieu	68%	0.78	5	59%	0.99	6

Table 15: Statistic of publications on ACM Digital Library in each major categories listed in the ACM Computing Classification System.

CCS	3-year gap			4-year gap		
	1st DR	$T_{\frac{1}{2}}$	Max. CL	1st DR	$T_{\frac{1}{2}}$	Max. CL
hardware	56%	1.41	7	54%	1.53	8
comp. sys. organization	41%	1.64	10	39%	1.72	10
software	39%	1.64	10	36%	1.74	11
data	72%	0.74	4	69%	0.78	4
theory of computation	45%	1.51	10	42%	1.60	10
mathematics of computing	37%	1.80	11	34%	1.89	12
information systems	36%	1.83	11	34%	1.89	12
computing methodologies	28%	1.82	12	25%	1.89	12
computer applications	57%	0.94	6	54%	1.00	6
computing milieu	55%	1.09	6	52%	1.14	7

Table 16: Statistic of publications on ACM Digital Library in Computer Science major research categories. HCI is an abbreviation for human computer interaction.

Category	1-year gap			2-year gap		
	1st DR	$T_{\frac{1}{2}}$	Max. CL	1st DR	$T_{\frac{1}{2}}$	Max. CL
alg. and theory	61%	1.34	5	56%	1.54	7
programming language	59%	0.99	5	51%	1.42	6
computing	70%	0.66	3	64%	0.9	4
soft. eng.	67%	0.75	3	55%	1.11	5
operating systems	79%	0.44	2	72%	0.69	3
comp. arch	35%	1.61	8	30%	1.81	9
computer networking	52%	1.37	7	45%	1.67	7
security and privacy	75%	0.5	2	70%	0.57	2
data management	42%	1.41	7	35%	1.65	8
artificial intelligence	50%	1.54	5	45%	1.77	6
computer graphics	48%	1.28	6	42%	1.81	8
HCI	31%	1.65	9	25%	2.40	12

Table 17: Statistic of publications on ACM Digital Library in Computer Science major research categories. HCI is an abbreviation for human computer interaction.

Category	3-year gap			4-year gap		
	1st DR	$T_{\frac{1}{2}}$	Max. CL	1st DR	$T_{\frac{1}{2}}$	Max. CL
alg. and theory	54%	1.64	7	53%	1.66	7
programming language	47%	1.56	6	44%	1.47	7
computing	60%	1.01	4	58%	1.05	4
soft. eng.	51%	1.19	5	48%	1.28	5
operating systems	69%	0.74	3	67%	0.76	3
comp. arch	27%	1.92	9	27%	2.09	10
computer networking	42%	1.71	7	41%	1.73	7
security and privacy	66%	0.65	2	65%	0.67	2
data management	33%	1.72	8	31%	1.73	8
artificial intelligence	43%	1.79	6	42%	1.79	6
computer graphics	39%	2.04	9	38%	2.07	9
HCI	23%	2.46	13	22%	2.53	13

Table 18: Statistic of publications on IEEE Xplore in Computer Science major research categories. HCI is an abbreviation for human computer interaction.

Category	1-year gap			2-year gap		
	1st DR	$T_{\frac{1}{2}}$	Max. CL	1st DR	$T_{\frac{1}{2}}$	Max. CL
alg. and theory	70%	0.58	2	63%	0.83	3
programming language	83%	0.39	2	76%	0.55	3
computing	65%	0.86	5	57%	1.20	7
soft. eng.	82%	0.41	2	75%	0.50	2
operating systems	100%	N/A	1	100%	N/A	1
comp. arch	63%	0.95	6	54%	1.37	8
computer networking	48%	1.11	7	39%	1.47	9
security and privacy	N/A	N/A	N/A	N/A	N/A	N/A
data management	72%	0.65	3	65%	0.92	4
artificial intelligence	58%	0.88	5	47%	1.32	8
computer graphics	63%	0.89	5	57%	1.20	7
HCI	N/A	N/A	N/A	N/A	N/A	N/A

Table 19: Statistic of publications on IEEE Xplore in Computer Science major research categories. HCI is an abbreviation for human computer interaction.

Category	3-year gap			4-year gap		
	1st DR	$T_{\frac{1}{2}}$	Max. CL	1st DR	$T_{\frac{1}{2}}$	Max. CL
alg. and theory	59%	0.98	3	58%	1.27	4
programming language	73%	0.72	4	71%	0.78	4
computing	52%	1.39	8	50%	1.44	8
soft. eng.	72%	0.69	3	71%	0.74	3
operating systems	100%	N/A	1	100%	N/A	1
comp. arch	50%	1.54	9	47%	1.63	9
computer networking	35%	1.65	10	32%	1.72	10
security and privacy	N/A	N/A	N/A	N/A	N/A	N/A
data management	62%	1.07	5	60%	1.16	5
artificial intelligence	42%	1.51	9	39%	1.60	9
computer graphics	51%	1.47	8	49%	1.54	9
HCI	N/A	N/A	N/A	N/A	N/A	N/A

Table 20: Evolution of research communities in terms of average size of a research group and number of years it was active based on the ACM and IEEE datasets.

Dataset	Average Value of	
ACM	Chain Length	4.48
	Cluster Size	6.1
	Intersection of 2 Consecutive Clusters	3.45
	Intersection of 3 Consecutive Clusters	2.51
	Intersection of 4 Consecutive Clusters	2.0
	Density	0.84
IEEE	Chain Length	4.39
	Cluster Size	5.53
	Intersection of 2 Consecutive Clusters	3.17
	Intersection of 3 Consecutive Clusters	2.36
	Intersection of 4 Consecutive Clusters	1.90
	Density	0.80

Table 21: The list of Computer Science conferences from [29]

Research Categories	Conference abbriviations
Alg. and Theory	STOC, FOCS, SODA, SoCG, ICALP, STACS, ESA, LICS, ISAAC, APPROX, RANDOM, CCC, SPAA, PODC, MFCS, FSTTCS, COCOON, WoLLIC, SODA, WADS, SWAT, WAOA, SoCG, ACM GIS, GD, IMR, WAFR, CCCG, EuroCG, ISSAC, LICS, IPCO, DLT, CIAA, DCFS, FWCG
Prog. Lang.	POPL, PLDI, ECOOP, OOPSLA, ICLP, JICSLP, ICFP, CGO HOPL, ESOP, FOSSACS, CP, CC, PADL, LOPSTR, FLOPS,
Computing	PODC, ICDCS, SPAA, PPoPP, HiPC, DISC, CLUSTER, WDAG, SRDS, PACT, IPDPS, IPPS, SPDP, CCGrid, DSN, ICPP, Euro-Par, SIROCCO, OPODIS, ICPADS, Grid, Coordination, SC, SUPER, ICS, HPDC, PPSC, IWCC,
Soft. Eng.	ICSE, FSE, TACAS, PEPM, RTA, ICSM, ASE, SAT, FM, SAS, MoDELS, UML, RE, ICSR, ICECCS, CAV, FME, FORTE, WSA
Operating Systems	SOSP, OSDI, USENIX, FAST, EuroSys, HotOS, NOSSDAV, Middleware, MSST
Comp. Arch.	ASPLOS, ISCA, MICRO, HPCA, SPD, ASP-DAC, ISLPED, FCCM, FPGA, ISSS, CODES+ISSS, ISPD, ARVLSI, ISCAS, RTSS, RTAS, LCTES, CASES, CHES, EMSOFT, ECRTS, SCOPES, DAC, ICCAD, DATE,
Comp. Networking	SIGCOMM, NSDI, SIGMETRICS, IMC, INFOCOM, ICC, CONEXT, HotNets, IPTPS, ICNP, PAM, IWQoS, SenSys, MASCOTS, IM, P2P, ICCCN, Networking, LCN, HotMobile, GlobeCom, MobiCom, MobiHoc, MobiSys, WMCSA, IPSN, Ubicomp, PerCom, EWSN, ISWC, MSWiM, MobiQuitous, WoWMoM, SECON, WiOpt, DCOSS, MASS, IEEE RFID
Security & privacy	Oakland, USENIX, CCS, NDSS, ESORICS, RAID, ANTS, CRYPTO, EUROCRYPT, ACNS, TCC, CSF, CSFW, PKC, ASIACRYPT, FSE, RSA, CHES, SECRYPT, INDOCRYPT
Data Management	SIGMOD, VLDB, PODS, SIGIR, WWW, KDD, ICDE, CIDR, ICDM, ICDT, EDBT, SDM, CIKM, ICIS, SSTD, SSD, WebDB, SSDBM, CAiSE, ECIS
AI	AAAI, IJCAI, AISB, NLDB, AAMAS, ATAL, ICMAS, ICAPS, AIPS, ECP, ICML, NIPS, COLT, EuroCOLT, ECML PKDD, ECML, KR, PKDD, EWSL, ECAI, RuleML, FOGA, IJCAR, CADE, COLING, TABLEAUX, LPAR, WoLLIC, ICCV, CVPR, ECCV, BMVC, CICLing, ACCV, ICPR, CAIP, SCIA, PSIVT, SSIAI, ACL, NAACL, EACL, UAI
Comp. graphics	SIGGRAPH, I3D, SI3D, I3DG, MM, ACMMM, DCC, ICME, ICMCS, Vis, Eurographics, ACM SIGGRAPH, InfoVis, SCA, ICIP, GI
HCI	CHI, CSCW, UIST, IUI, DIS, INTERACT, MobileHCI, SIGDOC, VL/HCC, ASSETS