Identifying Temporal Patterns and Key Players in Document Collections

Benyah Shaparenko Rich Caruana Johannes Gehrke Thorsten Joachims
Department of Computer Science
Cornell University
Ithaca, NY 14853
{benyah, caruana, johannes, tj}@cs.cornell.edu

Abstract

This paper considers the problem of analyzing the development of a document collection over time without requiring meaningful citation data. Given a collection of time-stamped documents, we formulate and explore the following two questions. First, what are the main topics and how do these topics develop over time? Second, to gain insight into the dynamics driving this development, what are the documents and who are the authors that are most influential in this process? Unlike prior work in citation analysis, we propose methods addressing these questions without requiring the availability of citation data. The methods use only the text of the documents as input. Consequentially, they are applicable to a much wider range of document collections (email, blogs, etc.), most of which lack meaningful citation data. We evaluate our methods on the proceedings of the Neural Information Processing Systems (NIPS) conference. Even with the preliminary methods that we implemented, the results show that the methods are effective and that addressing the questions based on the text alone is feasible. In fact, the text-based methods sometimes even identify influential papers that are missed by citation analysis.

1. Introduction

Many document collections have grown through an interactive and time-dependent process. Earlier documents shaped documents that followed later, with some documents introducing new ideas that lay the foundation for following documents. Examples of such collections are email repositories, the body of scientific literature, and the web. To access and analyze such collections, it is important to understand how they developed. For example, consider a historian trying to get an understanding of the ideas and forces leading to the Iraq war from news articles. Or, consider the head of a hiring committee trying to understand which scientists had the greatest influence on the development of a discipline.

In this paper, we pose and consider the problem of analyzing the temporal development of a document collection. This problem requires simultaneously understanding what topics are popular and which documents and authors drive the changes in popularity of the topics. In particular, we address the following questions:

- What are the key topics in a collection of documents and how did their popularity change over time?
- Which documents introduced new ideas that had large impact?
- Who were the authors that significantly drove the evolution of ideas?

To answer these questions for general document collections, we impose that our algorithms must work without meta-data augmenting the document representation. In particular, since most collections lack meaningful citation and hyperlink structure, the analysis must be done entirely based on the text in the document.

Most existing work related to these questions has focused on exploiting meta-data like hyperlinks and citation information. Graph-based algorithms like HITS [9], PageRank [15], and its descendants (see e.g. [3]) exploit information in the hyperlink structure to find outstanding documents. These algorithms are based on citation-analysis methods from bibliometrics (see e.g. [12]) that are used to detect related work and define impact [4, 5]. In contrast to using citation data, we propose complementary methods that use solely the text of the documents to make them applicable beyond scientific literature and the web. To the best of our knowledge, there is no existing method that uses only the text of the documents to determine the most influential documents or authors.

For the problem of discovering topics and trends in a collection of documents, however, there is quite a body of work already. The TDT evaluations (see e.g. [1, 2]) emphasized online new topic detection for news articles. Other work

has focused on burst detection, correlating real-world events such as the rise and fall of a topic's popularity with single words from the documents [20, 10]. Evolutionary theme patterns demonstrate the entire "life cycle" of a topic from a probabilistic background [13]. Other recent work presents efficient algorithms specially designed for thread detection [6]. We build upon this work for visualizing the development of topics over time.

The main contribution of our work is the definition of an interesting research problem, namely, how to identify a document collection's most influential documents and authors using only the text of the documents. We present one such method and show that this type of problem is in fact feasible and that even simple methods lead to interesting results. In an empirical evaluation on a collection of scientific articles, our method was able to identify influential documents and authors successfully. In particular, we compare the results with citation counts and find that the new identification methods find papers with new and influential ideas even in some cases where citation analysis fails. By using the information of which author wrote which document, we can also determine who are the most influential authors of this document collection.

The paper is structured around the three questions from above, which we address in turn. After describing the related work in more detail in Section 2, we introduce in Section 3 the data that is used as the testbed. In Section 4 we present the clusters/topics visualization, and Sections 5 and 6 present our method for identifying influential documents and authors, respectively.

2. Related Work

Our work identifies influential documents and authors and provides a way of visualizing the topical development of a document collection. There is some work on identifying influential documents and authors – however, previous work uses citations, not simply the text. There is much work on identifying trends in document collection. We review the related work in the following.

2.1. Influential Documents and Authors

Since our main goal is identification of key documents and authors, the most related work exists in the fields of bibliometrics and citation analysis. Work in bibliometrics (e.g. [12]) uses citation analysis for a set of research papers to determine the most influential authors and papers. It finds that the number of citations is the best predictor for a paper's influence. Other bibliometric work has also considered the issues of how to find leading documents and authors [14, 22]. Leading documents and authors can be found by analyzing the citation graph.

McGovern et al. use the hubs and authorities algorithm [9] to identify authoritative documents, and then define authoritative authors as the authors who write several papers among the most authoritative ones. Like this previous work, we seek to identify the authoritative authors and documents. However, our work is more general since we use only the text of the documents instead of bibliographic information. Consequentially, we can also handle domains such as news articles where there are no formal citations and successfully find the leading documents and authors.

For finding leading documents in a hyperlinked environment, the classic algorithm is PageRank [15]. Used by Google, PageRank finds the most influential documents by considering the reputations of the documents in the collection, and which documents link to which other documents. A document's reputation is raised (or lowered) based on the number and reputation of the documents citing it. More documents linking to a document mean that document enjoys greater popularity. Reputable documents linking to a document mean that document should also be reputable.

Besides looking at the impact of individual documents or authors, citation analysis has also tackled the problem of finding the journals with the most impact [4, 5]. Though not without controversy, the impact factor uses citations to measure how important the articles within a journal are on average. In general, more citations means greater popularity. However, because of variables such as journal size and shifts in journal popularity over time, when compared with raw citation counts, the impact factor calculates a more accurate measure of the influence of a particular journal's papers. This vein of work is similar to ours because the problem formulation presented in our work generalizes to groups of documents and authors, not just individual documents and authors. For example, one could think of ranking universities by their influence in a research community. Instead of using citation analysis, we could use the text produced by the research groups in these universities to rank the universities.

2.2. Temporal Topic/Trend Detection

Besides finding leading documents and authors, we additionally present a visualization of the topics in a document collection. Related work in this area starts with early work on new topic detection. The TDT studies [1, 2] investigated online new topic detection for news articles. In some sense, the online version of the problem is harder than the one we consider. We assume that we already have all the documents in the collection with time stamps and that they can be processed offline. Although both our work and the TDT work both have as a goal topic detection, another difference is that TDT focused on detecting the arrival of new topics, while our work focuses on providing an overview for how

the topical foci of a document collection changes over time.

Independent component analysis (ICA) is another method that can be used for similar purposes as solutions to the TDT task. For example, ICA has been used to distinguish topics in the CNN news chat room logs [11]. Like our work, this usage of ICA is unsupervised and relies only on the text. After performing principal component analysis, the ICA algorithm distinguishes the main topics. This work graphs graphs the existence of these topics over time, but it is hard to guage relative strength of the topics. Our work shows how topics rise and decrease in strength over time.

Our work bears more similarity to burst detection [10] and timeline creation [20]. Both burst detection and timeline creation seek to correlate real-world events with the text used in the document collections. There is an implicit assumption that as real-world events change, the text used in the documents will change as well. Words that nobody used at one time may become widely popular, e.g. words describing a new technology or new idea. In the context of research papers, when the burst detection code is run on a set of research papers, the bursts seem to correspond to the rise and fall in popularity of research topics. By using a state machine approach, bursts can be detected in anything from email, to Presidential State of the Union addresses, to research papers. The wide-ranging applications are possible because, similar to our work, burst detection assumes only time-stamped text documents.

As in burst detection, recent work in thread detection has proposed efficient, formal models [6]. These models do not depend on a flat clustering or time-stamped documents, but instead focus on using time in the algorithm. For example, instead of calculating all pairwise document similarities, this thread detection work only considers two documents similar if they both contain the same term and occur within a set time window of each other. This work therefore does not suffer from one problem of flat clustering – that of emphasizing cluster coherence at the loss of identifying developing and changing strands of topics. Even though using an algorithm specifically designed for temporal clustering may provide better clusters, our emphasis is on visualizing the topics, not on the actual clustering method, so we just use a simple flat clustering.

Another interesting direction of previous work that works with developing strands of research is in detecting evolutionary theme patterns. It is different from burst detection and our work in that the evolutionary theme patterns emphasize displaying the entire "life cycle" of a theme [13], while burst detection and our work simply considers a flat version of clustering. Detecting evolutionary theme patterns not only detects when a topic develops and fades, but also what future or other topics may have been influenced by this topic. In some sense, the theme evolution graphs present ways of depicting flows of ideas in the document collection throughout time.

3. Data and Testbed

Before presenting the methods addressing the three questions from above, we first discuss the type of data we are considering. We assume that the collection consists of documents where:

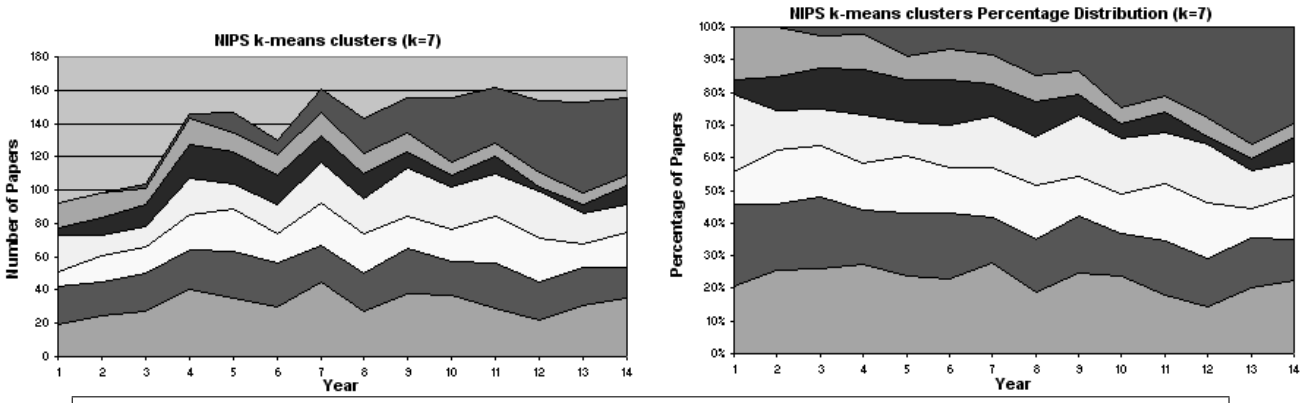
- the text of the documents is accessible,
- the documents are time-stamped and assumed to arrive in (or can be grouped into) batches,
- and there are dependencies between earlier and later documents.

Examples of such collections are email, proceedings of scientific conferences, scientific journals, news, and blogs.

As a testbed, we chose a collection of scientific articles, in particular the articles published in the proceedings of the Neural Information Processing Systems (NIPS) conference [8] between 1987 and 2000. The reason for choosing this data set is threefold. First, we believe that scientific document collections fulfill the assumptions stated above. Second, for scientific articles, citation data is available and we can compare our methods against citation counts. And third, we are familiar with the development of this scientific community, which allows us to evaluate the performance of the algorithms as an informed insider.

Since we consider fourteen years of research papers, we expect to see several strong trends as topics develop and change over time. This set of full text documents was obtained by OCR. There are a total of 1955 documents, with approximately 100 documents in the first two years, and then 150-160 documents each year in the last twelve years. We use only the text (not the citation or bibliographic information) from these documents. As meta-data, we use only the time-stamps (year) of the documents and the extracted author names of each article.

As our text-based representation, we chose a standard vector-space approach [18]. In particular, we convert the text documents to a standard TFIDF ("ltc") representation[19]. In this representation, the features are words from the available text. We ignore stopwords and words that only occur once, but consider all other words as features. No stemming is used. To build a TFIDF vector for each document, we count the number of times term t appeared in the document. Then we multiply by the IDF weighting factor of $\frac{n}{log(n_t)}$, where n is the number of documents in the corpus and n_t is the number of documents that contain the term t_i. To then determine the similarity of two documents, we use the standard cosine similarity between the TFIDF vectors.



Cluster Descriptions:	
6: bayesian, mixture, gaussian, posterior, likelihood	3: policy, reinforcement, state, controller, action
5: chip, circuit, analog, voltage, vlsi	2: image, images, object, objects, recognition
4: speech, word, hmm, recognition, speaker	1: spike, cells, neurons, cell, firing
	0: training, error, generalization, margin, hidden

Figure 1. Clusters proceed from cluster 0 on the bottom of the graph to cluster 6 on the top. (left) The distribution of $k = 7$ clusters. The histograms of each cluster are stacked on top of each other to show the effects of cluster popularity over time. (right) The percentage distribution of $k = 7$ clusters. In this case, we normalize the histograms by the number of documents per year.

4. How do key topics change over time?

The first problem we consider is that of visualizing the key topics of a document collection, and how the popularity of these topics develops over time. The goal is to provide a concise summary of the high-level development of topics even for large-scale document collections that are too expensive to analyze manually. Following the flavor of ideas from ThemeRiver [7], we will summarize the development of topics using "Temporal Cluster Histograms."

4.1. Method

Our method proceeds in three steps. In the first step, we determine the key topics in the document collection via clustering. Each cluster represents a key topic. In the second step, a concise description of the key topic for each cluster is formed. And in the final step, we visualize the temporal behavior of topics as a flow through time indicating increasing or decreasing popularity.

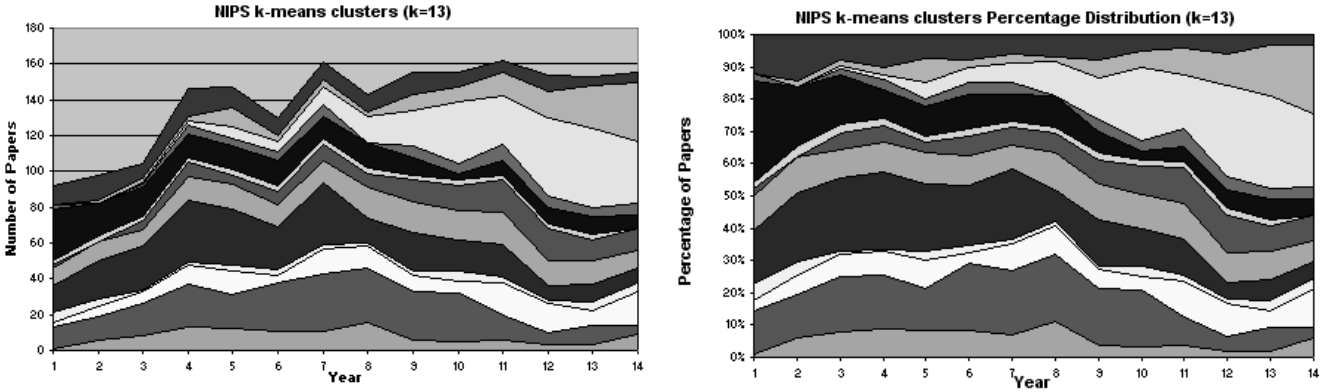
As the clustering algorithm in the first step we use k-means, in particular Weka's [17] implementation. We modified Weka for this application so that cosine distance could be used for k-means clustering. Since k-means may get stuck in local maxima, for each value of k, we chose 10 random seeds and selected the clustering that had the least squared error.

To describe each cluster's topic, we extract the five words with the highest weights in the cluster's centroid. These five words are the most important terms in defining the cluster centroid. The number five is somewhat arbitrary, but was chosen because we found that five words are sufficient to convey a good sense of the cluster's content without presenting an overwhelming amount of information. Using the top five words allows us to reliably identify important terms describing the topic of a cluster.

Finally, we plot how topic popularity varies over time. For each year, we compute the number of documents that fall into each cluster and plot each cluster's yearly breakdown as a stacked histogram. Using stacked histograms clearly presents the changes in cluster size over time as a flow. Note that while the k-means clustering does not take time into account when clustering the documents, this last step relates clusters to time.

4.2. Results

Figure 2 shows the results of the method as applied to the NIPS data for $k = 13$. Most clusters directly represent topics and reflect our knowledge of the NIPS community very well. In particular, clusters 10 and 11 clearly show the two emerging research areas in NIPS, namely "Bayesian Methods" and "Kernel Methods" like Support Vector Machines (SVMs). The graph correctly indicates that the topic of Bayesian analysis started before the kernel methods cluster, with both topics starting to dominate the NIPS conference in 2000. Also, it correctly indicates that the Kernel Methods topic strongly gained in popularity at that time. On the



Cluster Descriptions:

12: chip, circuit, analog, voltage, vlsi
11: kernel, margin, svm, vc, xi
10: bayesian, mixture, posterior, likelihood, em
9: spike, spikes, firing, neuron, neurons
8: neurons, neuron, synaptic, memory, firing
7: david, michael, john, richard, chair
6: policy, reinforcement, action, state, agent
5: visual, eye, cells, motion, orientation
4: units, node, training, nodes, tree
3: code, codes, decoding, message, hints
2: image, images, object, face, video
1: recurrent, hidden, training, units, error
0: speech, word, hmm, recognition, mlp

Figure 2. Clusters proceed from cluster 0 on the bottom of the graph to cluster 12 on the top. (left) The distribution of $k = 13$ clusters. The histograms of each cluster are stacked on top of each other to show the effects of cluster popularity over time. (right) The percentage distribution of $k = 13$ clusters. In this case, we normalize the histograms by the number of documents per year.

other hand, cluster 4 on supervised neural network training (e.g. feedforward neural networks), cluster 1 on recurrent neural networks, and cluster 8 on biologically-inspired neural memories were very strong in the early years of NIPS, but by 2000 almost disappeared from the conference. This phenomenon also agrees with our prior perception of the NIPS conference.

The only cluster that does not represent a topic is cluster 7. This cluster groups together the outliers in the collection, which are not scientific papers, but other types of documents. In particular, cluster 7 contains author indexes, subject indexes, the NIPS introductory page, and the start of the proceedings. In this case, the clustering helps clean the data and identify outliers that do not fit any "content" topic classes.

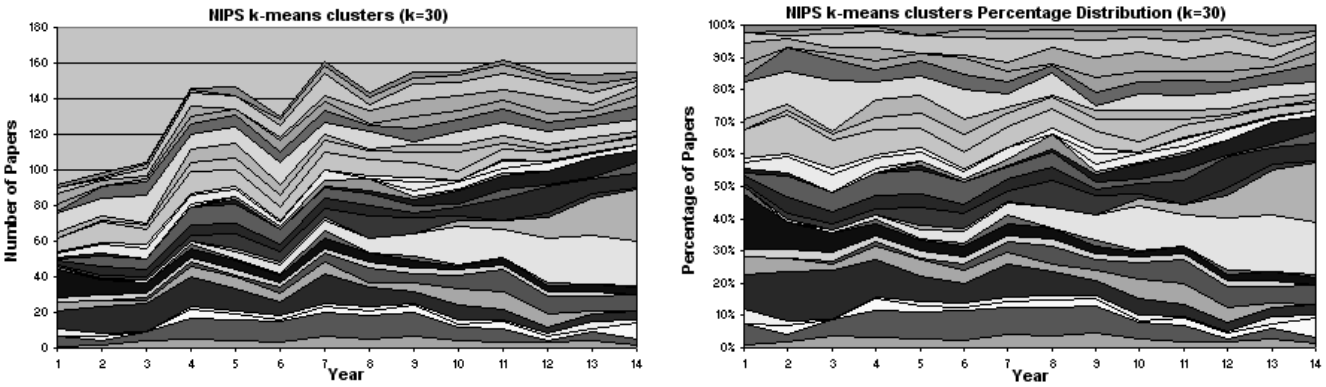
Our method of extracting keywords from the cluster centroids works reasonably well – many of the words are highly informative for the cluster content. The top five words shown give a reasonable description of the main topics in the NIPS conference.

Figures 1, 2, and 3 show the results for all values of k that we used, namely k = 7, 13, and 30, respectively. For the clusterings with more or fewer numbers of clusters, topics get merged and split in a reasonable fashion. Interestingly, the emerging clusters on Bayesian Methods and Kernel Methods are rather homogeneous, and do not get split even for large numbers of clusters. In Figure 3, with 30 clusters, these two areas are still well-defined and seem to show similar behavior as the depiction with 13 clusters. These two clusters are very strong – even when there are only 7 clusters as in Figure 1, these two topics still stand out among all the rest (even though the Kernel Methods and Neural Nets clusters have been combined by the clustering algorithm).

Overall, we believe that the cluster analysis and its visualization reflect correctly the development of the NIPS conference.

5. Which are the most influential documents?

Now that we have a way of visualizing clusters and determining how the topics developed over time, we would like to identify the proponents driving these changes. At the first level, these proponents are documents and the ideas they convey. Determining which documents are most influential on later work gives insights into the ideas driving the changes in the document collection over time. While some of this influence is conveyed through citations in the area of scientific literature, the goal is a general solution that will work for any text document. Consequentially, we restrict our methods to using only the text of the document, but use citation data to evaluate the quality of our methods.



Clusters:

29: auditory, sound, cochlear, speech, frequency
28: clustering, cluster, clusters, som, codebook
27: theorem, vc, bounds, bound, ¡
26: student, teacher, dynamics, replica, spin
25: spike, firing, spikes, neuron, neurons
24: object, tree, node, nodes, objects
23: cells, cell, cortical, cortex, neurons
22: robot, controller, control, reinforcement, critic
21: obs, gradient, convergence, momentum, obd
20: chip, circuit, analog, voltage, vlsi
19: vor, head, vestibular, eye, velocity
18: trajectory, units, hidden, weights, training
17: option, policy, portfolio, call, traffic
16: tangent, td, distance, prototypes, simard
15: ica, blind, separation, sources, eeg
14: word, speech, speaker, recognition, words
13: image, images, face, texture, wavelet
12: motor, eye, movement, movements, visual
11: kernel, margin, svm, kernels, adaboost
10: bayesian, gaussian, posterior, mixture, likelihood
9: routing, rod, bipolar, router, game
8: memory, capacity, synaptic, associative, memories
7: david, michael, john, richard, chair
6: policy, reinforcement, agent, action, state
5: motion, visual, velocity, orientation, direction
4: units, hidden, classifier, training, unit
3: code, codes, decoding, hint, hints
2: video, tracking, audio, image, camera
1: recurrent, state, units, hidden, network
0: mlp, hmm, speech, ensemble, rbf

Figure 3. Clusters proceed from cluster 0 on the bottom of the graph to cluster 29 on the top. (left) The distribution of $k = 30$ clusters. The histograms of each cluster are stacked on top of each other to show the effects of cluster popularity over time. (right) The percentage distribution of $k = 30$ clusters. In this case, we normalize the histograms by the number of documents per year.

5.1. Method

We define the impact of a document as the amount of followup work it generates. As a measure of influence of a paper on later work, we propose a *lead/lag index*. It is based on the assumption that "imitation is the highest form of flattery," i.e. if one document spawns a great deal of followup work that uses similar vocabulary, then that document was very influential. In particular, the lead/lag index measures whether a document is more of a leader or more of a follower. We assume that leaders have many papers following them, and vice versa. The general idea is illustrated in Figure 4. More formally, the index is defined as follows.

For each document d, we find the k nearest neighbors $knn(d)$ in terms of the cosine distance between TFIDF vectors. We then count the number of neighbors that are published later than d

$$k_{later} = |\{d'|(d' \in knn(d)) \wedge (time(d') > time(d))\}|$$

and the number of papers that precede the paper

$$k_{earlier} = |\{d'|(d' \in knn(d)) \wedge (time(d') < time(d))\}|\,.$$

By comparing these two numbers, it is possible to determine the degree to which a paper builds upon influential ideas vs. proposing new ideas that have influence on later documents.

The raw lead/lag index of a document d is computed by subtracting the number k_{later} of papers following the current paper in time from the number $k_{earlier}$ of papers preceding the current paper in time.

$$I^d_{raw} = k_{later} - k_{earlier}$$

However, the index is strongly affected by edge effects. For example, $k_{earlier}$ is guaranteed to be zero for documents from the first time step. To avoid such biases, we scale each year's documents by normalizing it across all papers from the same time step. In particular, we subtract the average of

Rank	Year	Citations	Paper Title and Author(s)
1.167	1996	128	"improving the accuracy and speed of support vector machines" chris j.c. burges, b. scholkopf
1.128	1999	17 (466)	"using analytic qp and sparseness to speed training of support vector machines" john c. platt
0.986	1999	18	"regularizing adaboost" gunnar ratsch, takashi onoda, klaus-robert muller
0.953	1996	41 (3711)	"support vector method for function approximation, regression estimation, and signal processing" vladimir vapnik, steven e. golowich, alex smola
0.945	1998	27	"training methods for adaptive boosting of neural networks" holger schwenk, yoshua bengio
0.945	1997	3	"modeling complex cells in an awake macaque during natural image viewing" william e. vinje, jack l. gallant
0.934	1998	17	"em optimization of latent-variable density models" c. m. bishop, m. svensen, c. k. i. william
0.934	1995	584	"a new learning algorithm for blind signal separation" s. amari, a. cichocki, h. h. yang
0.934	1995	16	"fast learning by bounding likelihoods in sigmoid type belief networks" t. jaakkola, l. k. saul., i. jordan
0.914	1998	49	"dynamically adapting kernels in support vector machines" nello cristianini, cohn campbell, john shawe-taylor
0.914	1999	27	"approximate learning of dynamic models" xavier boyen, daphne koller

Figure 5. Based on the lead/lag index, above is a list of the most influential NIPS papers when considering the paper's $k = 14$ nearest neighbors. According to our algorithm, these influential papers inspire the most followup work. We also provide the year of publication and the number of citations the papers received according to Google Scholar. Numbers in parentheses signify that there is a related publication by the same author(s) with similar content that receives most of the citations.

the raw lead/lag indices for a year from each raw lead/lag index in that year.

$$I^d_{scaled} = \frac{1}{k}\left(I^d_{raw} - \frac{|\{d_i : time(d_i) = time(d)\}|}{\sum_{\{d_i:time(d_i)=time(d)\}} I^{d_i}_{raw}}\right)$$

The resulting *scaled* lead/lag index corrects for such edge effects. The higher the scaled lead/lag index, the more influential the paper. Note that the scaled lead/lag index is also normalized with respect to k. This scaling process makes values from different choices of k comparable. The scaled lead/lag index scores typically fall in the interval from -1 to +1, with extremely strong papers receiving scores slightly above +1 and extremely lagging papers receiving scores slightly below -1.

5.2. Results

We computed the scaled lead/lag index for the NIPS data set. The value of k is the only parameter that needs to be selected. With a small k, only the closest documents to a particular document are considered. If a paper is very influential, then other documents influenced by that paper are missed in this analysis. On the other end of the spectrum, if k is too large, documents that are only marginally affected by a particular paper are included in the ranking. This can lead to noisier results. We run the experiments for k = 7, 14, 24, and 49.

Figure 5 shows the results on the NIPS data for $k = 14$. Different values of k agreed more or less on which documents are most influential, with only small changes in the ordering among the top scoring papers. This indicates that the method is robust with respect to the choice of k, and that most reasonable values of k produce comparable results.

The list of the most leading NIPS papers computed by our algorithms closely reflects our insider perception of the NIPS conference. First, among the highest ranked papers are those presenting central new ideas on support vector machines, which, as also evident from the Temporal Cluster Histograms, have had an outstanding influence on the

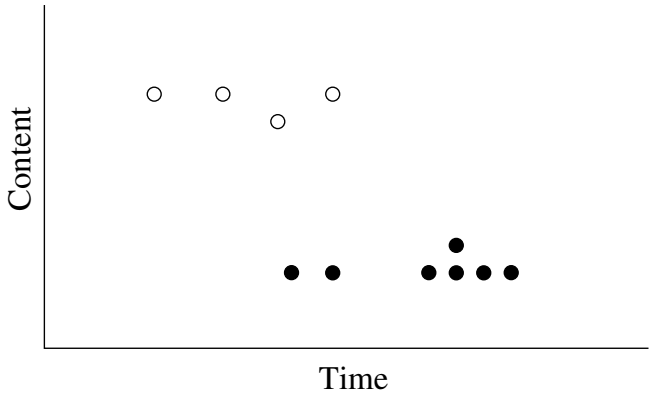


Figure 4. The main idea of the lead/lag index is to decide whether a paper is more of a leader or follower based on whether similar papers content-wise follow or precede the paper in question. In this figure, graphed items represent documents where the open circles are of one topic and the black dots are of another topic. The open circles (and black dots) towards the left of their respective clusters are leaders because similar documents follow these documents. On the other hand, the circles (and dots) towards the right are followers because documents with similar content precede these points.

development of the NIPS community. The first two SVM papers published in NIPS are ranked first and fourth in our algorithm's ranking, receiving recognition for being first in one of NIPS hottest topics.

Second, the results from our method are different from what citation analysis would produce. The number of citations of each paper according to Google Scholar is given in the third column of Figure 5. (These counts measure the impact throughout all venues, whereas our ranking measures the impact within the NIPS community.) Our method reflects the importance of ideas presented in the paper, not how often this paper was cited. For example, the publication by John Platt was the first refereed paper to propose the SMO algorithm for support vector machine training, which has become one of the standard methods for this problem. However, this paper has only 17 citations in Google Scholar, since most authors cite a book chapter with similar content and 466 citations. Due to this, citation analysis would not have recognized the importance and influence of the ideas in this paper. Another example is Vapnik's paper, which ranks fourth on this algorithm's ranking. Although the NIPS paper has just 41 citations, other work by Vapnik on support vector machines has many more citations (e.g. Vapnik's first book has 3711 citations).

Third, while many of the papers in Figure 5 are in the area of support vector machines, this dominant topic does not drown out influential ideas in other topic areas. An example is the paper by Amari et al., which is a fundamental paper for the topic of independent component analysis with 449 citations in Google Scholar. This paper is not only fundamental for independent component analysis, but as it turns out, it is also one of the most influential NIPS papers overall. In fact, this paper is the second most often cited of all NIPS papers.

In summary, the list of most influential papers agrees well with our opinion of the most influential ideas in the NIPS corpus. These results validate our assumption that measuring textual similarity provides an adequate method for determining which papers have influence on later papers in the document collection.

6. Who are the most influential authors?

Since papers do not write themselves, once we can determine the most leading documents, the next logical step is to ask who wrote them. Given a collection of documents, we would like to answer the questions of which authors produce the most original work, which authors are most influential in spreading their ideas, and which authors determine the pulse of the field and future directions of research.

6.1. Method

The document lead/lag index already provides a method for determining the influence of a document. To identify the most influential authors in the document collection, we can aggregate the document lead/lag information by author. Specifically, we address the following question: Which authors write documents that have a significantly high scaled lead/lag index?

To aggregate the lead/lag index scores by author, we compute the 95% confidence interval around the average lead/lag scores for each author. We then rank the authors by the lower 95% confidence bound. More specifically, consider an author with n papers receiving scaled lead/lag scores $L_{scaled}^{d_1}, \cdots, L_{scaled}^{d_n}$. For these scores, one can compute the confidence interval for the mean m from the sample variance v under assumption of normality as $m \pm 2 * \frac{\sqrt{v}}{n}$. However, this confidence interval is quite sensitive to anomalies for small samples. For example, one author may have two papers with medium rank and identical means. Then, the author will receive an excellent score because the variance estimate is zero. To smooth the variance estimate and reduce this problem, we add an extra document with weight -1 (a weight which is near the bottom of the

lead/lag rankings) to the list of docs. With the new mean m' and variance v', the lead/lag index of an author is

$$I^a = m' - 2 * \frac{\sqrt{v'}}{n+1}. \quad (1)$$

6.2. Results

We computed the author lead/lag index for all authors in the NIPS collection. Figure 6 has the results for $k = 14$. Results for different values of k are similar, so we only present $k = 14$. Overall, we find that this ranking for the most part identifies a document collection's key players.

From bibliometrics, we know that typically the best predictor of an author's importance is the number of citations that author receives [12]. Therefore, we compare our aggregated author lead/lag ranking to the number of citations an author has received on Google Scholar. (We searched by the author's name and added citation counts for the first 200 documents by that author or as many documents as had citations.) Note, however, that the citation counts measure the impact of an author in all venues, while our lead/lag index measures the impact in NIPS. Additionally, we present numbers for how prolific an author is, measured by the number of papers the author has published in NIPS. The author with the most papers is Terrence Sejnowski, who has 46. We find that highly-cited authors typically rank high in the author lead/lag index. For example, Michael Jordan has often published influential work, and the algorithm recognizes this by ranking him at the top of the list. Of the 1931 authors in NIPS, the 20 most influential authors according to the author lead/lag index (Figure 6) in general have a significant number of publications. Authors that are lower down in the ranking do not have nearly this number of publications. The authors that the aggregated lead/lag index identified for the most part represent well-known, leading names in the NIPS community. Therefore, we believe that aggregating the lead/lag index by author leads to a meaningful ranking of an author's influence on following work.

By and large, the algorithm works quite well in identifying key, influential authors. There are just 2 cases out of the top 20 where the authors do not have many citations. As it turns out, in both cases, the reason is an artifact of the data used, not our method of computing the author lead/lag index. Since the NIPS data set is obtained by OCR, we used an automated string match process to match the names (same first initial and last name within edit distance of 2). For both "D. D. Coon" and "Harrison Monfook Leong," the above name match heuristic combines many names. The name "D. D. Coon" here in fact represents many authors with short names. Similarly, the name "Leong" has many similar names in the NIPS data. With a perfect list of who authored which documents, this phenomenon would disappear. Therefore, we conclude that our method works as expected, producing a list of well-known, well-published authors.

7. Summary

We propose the problem of analyzing the temporal development of document collections for which there is no meaningful citation data available. As proof of concept, we propose simple methods that show that this problem is feasible and interesting. Unlike existing approaches from bibliometrics, the new methods are applicable even if no citation or hyperlink data is available. Using the proceedings of the NIPS conference as a testbed, Temporal Cluster Histograms were found to give an accurate and concise summary of the popularity of topics over time. To identify the papers with largest influence on topic development, we defined a document lead/lag index that is an effective indicator of the influence of a document. Finally, we extended the influence analysis to authors by aggregating document lead/lag indices. These lead/lag scores are the first measures able to identify key authors and documents in collections that lack citation information.

We believe that temporal analysis of document collections is an exciting area that deserves future research. The methods presented in this paper give evidence that such analyses are possible even without citation information. However, more principled approaches are likely to be even more accurate and could provide more meaningful insights. For example, currently there is no way to associate influential documents with clusters in the Temporal Cluster Histograms. It would be interesting to identify the set of papers that are responsible for spawning a new topic cluster. Similarly, it would be interesting to design specialized clustering algorithms that directly capture the splitting and merging of topics over time to get an overview of the "flow" of ideas in the collection. We are planning to explore these questions in future work.

References

[1] J. Allan, J. Carbonell, G. Doddington, J. Yamron, and Y. Yang. Topic Detection and Tracking Pilot Study: Final Report. In *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop-1998*, 1998.

[2] J. Allan, R. Papka, and V. Lavrenko. On-Line New Event Detection and Tracking. In *Research and Development in Information Retrieval*, pages 37–45, 1998.

[3] D. Cohn and H. Chang. Learning to Probabilistically Identify Authoritative Documents. In *Proceedings of the 17th ICML*, pages 167–174, Morgan Kaufmann, San Francisco, CA, 2000.

[4] E. Garfield. The Impact Factor. *http://www.isinet.com/essays/journalcitationreports/7.html/*.

Author	Rank	Papers	Citations	Author	Rank	Papers	Citations
jordan, michael i.	0.037	27	9284	bengio, yoshua	-0.131	18	1805
smola, alex	-0.004	13	3038	saad, david	-0.133	11	694
scholkopf, b.	-0.022	10	5338	bialek, william	-0.135	11	1547
atkeson, christopher g.	-0.06	10	3378	dayan, peter	-0.138	24	4014
williams, christophe k.i.	-0.067	16	1605	ghahramani, zoubin	-0.142	14	3171
sejnowski, terrence j.	-0.069	46	13955	shawe-taylor, john	-0.158	9	4014
hinton, geoffrey e.	-0.075	27	11643	tresp, volker	-0.162	16	672
jaakkola, tommi	-0.091	10	2918	sollich, peter	-0.173	9	739
miller, kenneth d.	-0.106	11	2447	barto, a.g.	-0.175	12	7100
coon, d. d.	-0.112	21	531	leong, harrison monfook	-0.196	15	0

Figure 6. The above list contains the NIPS authors with the highest ranking in the author lead/lag index. From considering each paper's $k = 14$ nearest neighbors for the document lead/lag index and then aggregating the document lead/lag index by author, our algorithm produces a ranking for how groundbreaking an author's work is. Since citation analysis has shown that the number of citations an author receives is typically the best estimate for an author's importance, we provide Google Scholar citation counts for these authors. Additionally, we provide counts for how many NIPS papers these authors have published.

[5] E. Garfield. The Meaning of the Impact Factor. *International Journal of Clinical and Health Psychology*, 3(2):363–369, 2003.

[6] R. Guha, D. Sivakumar, R. Kumar, and R. Sundaram. Unweaving a Web of Documents. In *Proceedings of KDD-2005*, Chicago, Illinois, 2005.

[7] S. Havre, B. Hetzler, and L. Nowell. ThemeRiver: In Search of Trends, Patterns, and Relationships. *IEEE Transactions on Visualization and Computer Graphics*, 2002.

[8] http://nips.djvuzone.org/txt.html. NIPS Online: The Text Repository.

[9] J. Kleinberg. Authoritative Sources in a Hyperlinked Environment. *Journal of the ACM*, 46(5):604–632, 1999.

[10] J. Kleinberg. Bursty and Hierarchical Structure in Streams. In *Proceedings of KDD-2002*, Edmonton, Alberta, Canada, 2002.

[11] T. Kolenda, L. K. Hansen, and J. Larsen. Signal Detection using ICA: Application to Chat Room Topic Spotting. In Lee, Jung, Makeig, and Sejnowski, editors, *Proc. of the Third International Conference on Independent Component Analysis and Signal Separation (ICA2001)*, pages 540–545, San Diego, CA, USA, 2001.

[12] A. McGovern, L. Friedland, M. Hay, B. Gallagher, A. Fast, J. Neville, and D. Jensen. Exploiting Relational Structure to Understand Publication Patterns in High-Energy Physics. In *Proceedings of KDD-2003*, Washington, DC, 2003.

[13] Q. Mei and C. Zhai. Discovering Evolutionary Theme Patterns from Text - An Exploration of Temporal Text Mining. In *Proceedings of KDD-2005*, Chicago, Illinois, 2005.

[14] F. Osareh. Bibliometrics, Citation Analysis and Co-citation Analysis: A Review of Literature I. *Libri*, 46:149–158, 1996.

[15] L. Page, S. Brin, R. Motwani, and T. Wingrad. The PageRank Citation Ranking: Bringing Order to the Web. Technical report, Stanford University, 1999.

[16] A. Popescul, G. W. Flake, S. Lawrence, L. H. Ungar, and C. L. Giles. Clustering and Identifying Temporal Trends in Document Databases. In *IEEE Advances in Digital Libraries ADL-2000*, pages 173–182, Washington, DC, 2000.

[17] P. Reutemann, B. Pfahringer, and E. Frank. Proper: A Toolbox for Learning from Relational Data with Propositional and Multi-Instance Learners. In *Proceedings of the 17th Australian Joint Conference on Artificial Intelligence*. Springer-Verlag, 2004.

[18] G. Salton. Developments in Automatic Text Retrieval. *Science*, 25(3):974–979, 1991.

[19] G. Salton and C. Buckley. Weighting Approaches in Automatic Text Retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[20] R. Swan and D. Jensen. TimeMines: Constructing Timelines with Statistical Models of Word Usage. In *Proceedings of KDD-2000*, pages 73–80, Boston, MA, 2000.

[21] F. B. Viegas, M. Wattenberg, and K. Dave. Studying Cooperation and Conflict between Authors with history flow Visualizations. In *Proceedings of CHI-2004*, Vienna, Austria, 2004.

[22] H. D. White. Citation Analysis and Discourse Analysis Revisited. *Applied Linguistics*, 25(1):89–116, 2004.