Knowl Inf Syst (2006) 11(1): 105–129
DOI 10.1007/s10115-006-0023-9

Knowledge and Information Systems

REGULAR PAPER

Wangzhong Lu · J. Janssen · E. Milios · N. Japkowicz · Yongzheng Zhang

Node similarity in the citation graph

Received: 16 February 2004 / Revised: 10 January 2005 / Accepted: 15 February 2005 /
Published online: 22 April 2006

Abstract Published scientific articles are linked together into a graph, the citation graph, through their citations. This paper explores the notion of similarity based on connectivity alone, and proposes several algorithms to quantify it. Our metrics take advantage of the local neighborhoods of the nodes in the citation graph. Two variants of link-based similarity estimation between two nodes are described, one based on the separate local neighborhoods of the nodes, and another based on the joint local neighborhood expanded from both nodes at the same time. The algorithms are implemented and evaluated on a subgraph of the citation graph of computer science in a retrieval context. The results are compared with text-based similarity, and demonstrate the complementarity of link-based and text-based retrieval.

Keywords Networked information spaces · Document similarity metric · Citation graph · Digital libraries

1 Introduction

The concept of information space has been proposed for collections of information that are organized so that the user can be aware of their structure and content,

W. Lu · E. Milios (✉) · Y. Zhang
Faculty of Computer Science, Dalhousie University, 6050 University Ave., Halifax, Nova Scotia, Canada, B3H 1W5
E-mail: eem@cs.dal.ca

J. Janssen
Department of Mathematics and Statistics, Dalhousie University, Halifax, Nova Scotia, Canada, B3H 3J5

N. Japkowicz
School of Information Technology and Engineering, University of Ottawa, Ottawa, Ontario, Canada, K1N 6N5

and use such awareness to navigate through them [6, 8, 11, 26]. Some information spaces are designed [1, 8], however, others are self-organizing and self-evolving by large numbers of people over a period of time. Several important information spaces, such as the World Wide Web, are networks consisting of information entities and links between them that imply a relation between the entities. We call the latter "Networked Information Spaces," to emphasize their connectivity aspect and the growing realization in the research community that connectivity is as important as content in organizing and retrieving information from such information spaces.

In order to navigate and mine the contents of a networked information space, it is of crucial importance to be able to judge similarity between information entities. Traditionally, similarity between information entities is computed based on their content. However, in a networked information space, a lot of information about similarity is encoded in the link structure of the graph. This link-based similarity can complement the classic content-based similarity measures [4] to produce a highly accurate similarity metric. Similarity is a key concept, not only in classical information retrieval (for which direct methods based on indexing are more efficient) but also for higher level tasks that involve the organization of large hyperlinked document corpora. Such tasks include clustering [16], automatic term extraction from clusters to build thesauri, and visualization of document corpora [6].

The body of scientific literature, where information entities are articles and links represent references to other articles, has existed as a networked information space in paper form for a long time, and is rapidly becoming available in electronic form through digital libraries and the World Wide Web [19]. In this paper, we explore various similarity metrics for the graph of the scientific literature, the citation graph, purely based on link structure. Our goal here is to investigate how much similarity information can be extracted just from the link structure. Our methods are easy to compute. Moreover, they are based only on a local neighborhood of the information entities in the networked information space. Therefore, they are applicable even when the networked information space is too large to fit on a desktop machine, provided a mechanism is available for local navigation from one information entity to its neighbors. For access to the computer science literature we use the electronic database *Citeseer (ResearchIndex)* [19].

Use of citation information to compute relatedness between scientific papers has been studied previously in contexts more limited than ours [14]. Since citations of other papers are hand-picked by the authors as being related to their research, the reference list of a paper contains information which can be exploited to judge relatedness. The simplest relation, a direct reference or citation, is likely to occur among related papers which are published apart in time. It does not occur very frequently among papers published in the same year or very close in time. Two different citation relations between papers have been specifically identified and used to calculate similarity, namely co-citation (two papers referenced by the same paper) and bibliographic coupling (two papers citing the same paper) [24]. Two papers are related by co-citation if they are cited together by the same paper. Small has studied the co-citation pattern among research papers and highlights its importance in similarity computation [24]. Co-citation links are often present in two related older papers. Two papers are bibliographically coupled, if they reference the

same paper. If two recent papers are published in the same or similar research area, a bibliographic coupling pattern is very likely to be found in their reference lists.

Bibliographic coupling and co-citation have been employed to compute similarity between research papers. But each of them is only suitable for computing similarity in specific cases. For instance, researchers have used co-citation frequency to compute relatedness between two papers, but the papers to be judged have to be well cited by other authors for the algorithm to work properly. Apparently co-citation is not efficient in judging similarity among recent papers which have not yet had the chance to be cited by many other authors. In terms of the direct link pattern, if the two papers are published almost at the same time, a direct citation link is not likely to be found between them, even if their content is related. Similarly, papers which appeared in the early stages of the development of a research specialty are not good candidates for bibliographic coupling analysis. In our metrics, we do not need to know which of these citation patterns our papers fall under. All patterns of citation relations are accounted for by using the citation graph.

Giles et al. [19] proposed a similarity measure based on common citations to judge the relatedness between papers. The metric, called "common citation $\times$ inverse document frequency"(CCIDF), is conceptually similar to the text-based similarity metric "term frequency $\times$ inverse document frequency"(TFIDF). The CCIDF metric assigns a weight to each paper, which is equal to the inverse of citation frequency in the entire database. To find documents related to a given paper, all the papers which have at least one reference in common with that specific paper are generated. The CCIDF metric is used by the automatic citation indexing system of *Citeseer*.

Our motivation for using the citation graph instead of comparing reference lists as in CCIDF is that the citation graph contains information which is much richer than that embedded in the reference lists, and which cannot be obtained just by comparing reference lists from different papers. Two papers may have no co-citation or bibliographic coupling relationship at all, but they could still have a strong relationship between them if their local citation graphs intersect substantially. For example, in Fig. 1 paper A references paper C, paper B references paper D, but A and B do not reference each other. Obviously, paper A and paper B are not related to each other in terms of CCIDF, co-citation or bibliographic coupling (i.e., through their direct references). But, if we expand the citation graph a little further, we may find out that papers C and D are strongly connected by bibliographic coupling links, and we could infer the relationship between papers A and B from papers C and D. Our method generalizes this notion by using both citations and references in the neighborhood of the two papers.

Dean and Henzinger [9] present algorithms for finding pages in the World Wide Web that are related to a given page. Their "companion algorithm" is similar to our algorithms in that it builds a neighborhood graph of the given page, it calculates hub and authority values of the nodes in this graph and returns the top ranked authority papers as the most similar papers to the given page. However, their algorithm does not, and cannot be trivially adapted to compute a similarity measure between two given nodes. The evaluation metric used is a precision metric (based on user studies) similar in nature to ours.

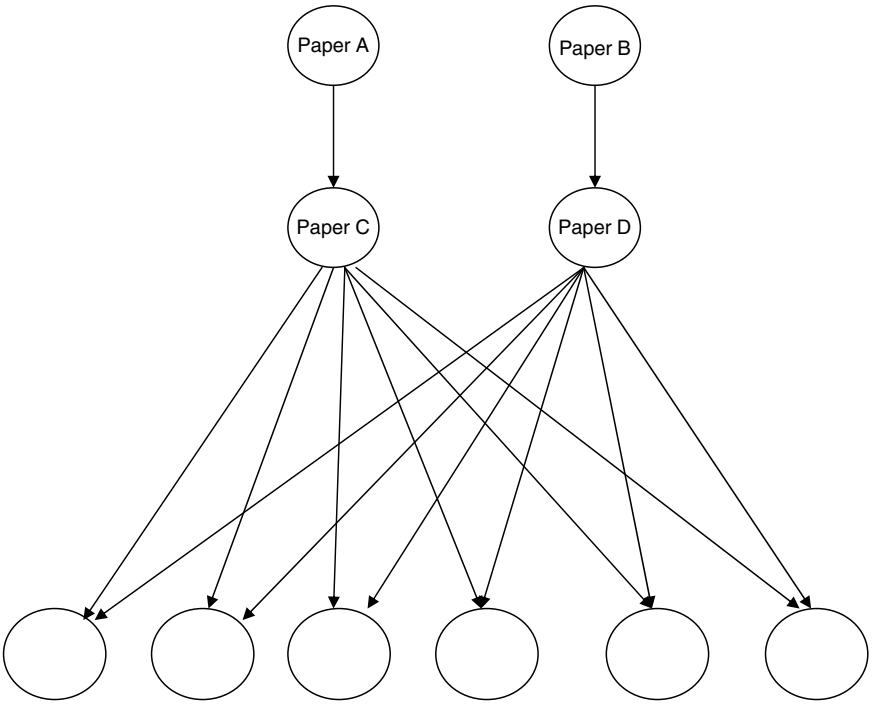

Fig. 1 Relation between papers inferred from citation graph

In our work, we propose two different graph-based metrics: the maximum-flow metric and the authority vector metric. In the maximum-flow metric, one joint local citation graph is generated from a pair of papers to be compared by following incoming and outgoing links from both papers. One paper is treated as a source node and the other as a sink node. Flow capacities are assigned to the edges. Then, the value of the maximum flow which could be pushed through from source node to sink node is computed, and used to represent the similarity between the two papers. In the authority-based metric, a local citation graph is grown separately for each paper to be compared, by following incoming and outgoing links separately for each paper. Then, authority weights [17] are computed for all nodes in each of the local citation graphs. Each paper is then represented by a vector, whose elements are the authority weights of the nodes in its local citation graph. Finally, similarity is computed as the vector distance between these vectors.

The motivation for using a citation graph for the evaluation of our graph-based similarity measures is twofold. Firstly, recent literature in bibliometrics [14, 15, 24, 25] suggests considerable interest in the comparison and classification of documents based on their citation environment. Secondly, the networked information space formed by scientific papers and their references can be expected to have a certain homogeneity. Therefore, such a space is more suited for the initial testing of new ideas than a less homogeneous space such as the World Wide Web. To emphasize the linked structure of our information space, we chose an online citation index for our studies, namely Citeseer, an online database of scientific

papers in Computer Science. Our access to this database was only via the Internet. We built a web robot to automate this access. Application of our measures on the World Wide Web, which is a lot less homogeneous than the citation graph, is a future research project. The similarities and differences between the citation graph and the Web are explored in [3].

The advantages of using the particular citation graph are:

- The papers included are fairly homogeneous in length and structure, and the references and citations have a close relation to the semantic content of the papers.
- The papers are in an area familiar to the authors, so the possibility exists to compare experimental results with our own judgement.
- Access was fairly straightforward, and full papers could be retrieved easily.

The disadvantages are:

- Our data consists of a body of scientific literature, so similarity of papers can only be judged by experts, and involves considerable time and effort.
- Citeseer contains a certain amount of "clutter" such as duplicate papers.
- Citeseer itself, as well as any subset that we used for our experiments, is not complete. In other words, the full text of the references of a paper in the database is not necessarily available in the database. This may well be a feature of any citation index, though. A cursory comparison with the science citation index, for example, showed that this well-established database showed about the same degree of "incompleteness" as our own collection.

In Sects. 2 and 3 we describe our metrics and the methods to compute them. In Sect. 4 we evaluate the metrics and the impact of their key parameter settings. We also describe how the local citation graphs, which are required for the similarity metrics, are being built. In Sect. 5 we compare the performance of the link-based metrics with text-based similarity metrics. Finally we discuss the results and propose future research directions.

2 Authority vector metrics

In this section, we describe the similarity metrics based on a vector representation of the neighborhoods of the two papers being compared. Given two research papers A and B, we construct two separate local citation graphs, graph A and graph B, for each of them. The idea is to compute the similarity of the given papers by comparing the similarity of their citation environments. It is not a trivial problem to compare graphs. Rather than comparing local citation graphs directly, we wish to use the most important or "authoritative" papers to represent a specific citation environment. The similarity between citation environments will then be based largely on these authority papers.

2.1 Authority papers

In order to implement this approach, we need to address first how to identify the authority papers in a given link environment. In other words, we need to find a

criterion to judge the importance of a given paper. In bibliometrics, citation analysis was used to measure the importance of scientific papers by Garfield [14]. Garfield also proposed a well-known metric to estimate the importance of journals by Impact Factor [15]. This metric, in graph-theoretic terms, amounts to a pure counting of the in-degree of nodes in the citation graph to compute how important a journal is. In ranking search results on the World Wide Web, people face the problem of how to determine the importance of a web page. Brin and Page proposed the PageRank algorithm [5]. Kleinberg [17] proposed a measure of the importance of web page by computing hub and authority weights. Kleinberg's hub and authority measure has been shown to be more stable [23]. Hubs and authorities as defined by Kleinberg's method can be seen as follows: a hub is a paper that points to many authorities; an authority is a page that is pointed to by many hubs. Hub and authority weights are computed by an iterative algorithm.

Intuitively, papers with high authority weight form the core of the most important papers in a specific research area. Hub papers might be review papers or tutorials; their content is broad and therefore papers that cite or are cited by hub papers are generally more loosely related than those that cite or are cited by authority papers. This intuitive notion has been confirmed by the hub/authority calculations done on the Web [17, 18]; web pages with high authority weight tend to be pivotal and important web pages on a certain subject, while web pages with high hub weight often are resource pages with many relevant links. One exception worth mentioning could be the "classic" review papers, that are heavily cited, therefore making them both hubs and authorities. It is possible to detect such papers and exclude them from the similarity metric, although we have not done so in our experiments. In order to get high similarity between two papers, their local citation graphs must have a large intersection and there must be many authority papers in the intersection area.

Finally, we note that our metric is different from CCIDF, the citation-based metric used by Citeseer to locate related papers. First of all, CCIDF uses in-degree as a measure of importance. Our metric uses authority weight, which is a more subtle measure of importance than in-degree. In many cases, authority weights are very close to the in-degree. In some cases, however, authority weights can be better than in-degree, for example when an authority paper is relatively recent, and it does not have a high in-degree, but it is cited by hub papers, or when a paper is near the fringe of the citation graph. This metric also is expected to reduce the negative effect of survey papers. These kinds of papers will be more likely to have long list of references, hence they are linked to many papers and can thus be wrongly treated as related to many other papers. In our authority vector metric, normalizing the vector before computing similarity will greatly lessen the negative influence of those papers. The local citation graph of survey papers will be larger than that of other research papers, and thus the vector of authority weights will have lots of components. If a vector has lots of components and is normalized, then in general individual entries will have lesser values, since their squares need to add up to 1. This will make the inner product smaller and hence lead to lower similarity with other papers. Also CCIDF is based on the citation frequency over the entire citation graph, while to compute our metric only a local citation graph is needed.

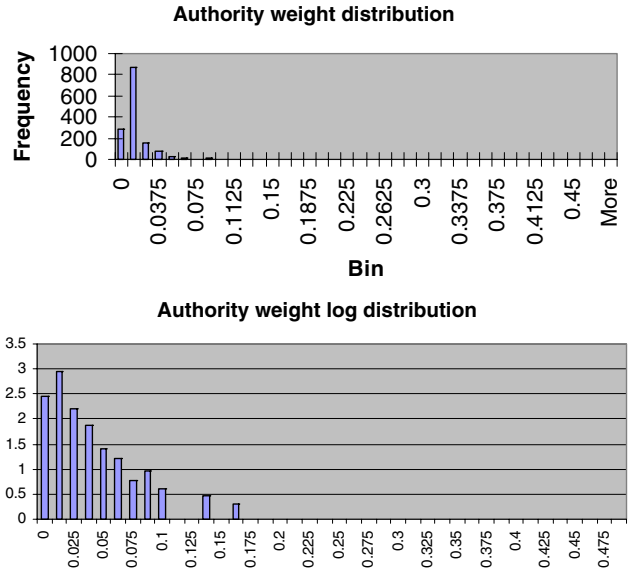

Fig. 2 The distribution histogram of authority weights. The *top graph* uses a linear scale and the *bottom graph* uses a logarithmic scale for the frequencies of authority weights

2.2 Algorithm

Given two papers, taking each of the papers as a seed, we build two separate local citation graphs, each consisting of k levels. Then we compute hub and authority weight for nodes for each of the graphs. We associate with each of the two papers a vector indexed by the nodes in the union of the two graphs. The component corresponding to a node has value 0 if the node is not in that graph. Otherwise, the value of the component is the authority weight of that node. Both vectors are normalized to length 1. Finally we obtain the similarity of the two papers by computing the cosine distance (i.e., the dot product, since vectors are normalized) of the two corresponding vectors.

For the computation of authority weights we use the iterative algorithm of [17]. Approximately 40 iterations were sufficient for convergence. The distribution of authority weights is shown in Fig. 2. We observe that a few nodes have authority weight that is much higher than average. These are the nodes we use for the vector representation of the local citation graph.

The outline of the algorithm is shown in Fig. 3.

3 Minimum cut/maximum flow metric

The key idea of this algorithm is to count the number of different paths in the citation graph between nodes representing two papers. The number of paths between two nodes is related to the *minimum cut*, the minimum number of edges needed to be cut to disconnect one node from the other. Therefore, our metric is based on a maximum flow/minimum cut computation in a local citation graph built from

ComputeVectorDistance(PaperID1, PaperID2)
Grow the citation graph $g1$ and $g2$ for paper 1 and paper 2.
Set authority weights for $g1$ and $g2$
Build $vector1$ for paper 1
Set the elements of $vector1$ to the union of vertices of $g1$ and $g2$
Initialize the value of each element in $vector1$ as 0
For all the vertices v in $g1$
Set value of $element[v]$ in $vector1 = AuthorityWeight[v]$ in $g1$
End For
Build $vector2$ for paper 2
Set the elements of $vector2$ to the union of vertices of $g1$ and $g2$
Initialize the value of each element in $vector1$ as 0
For all the vertices v in $g2$
Set value of $element[v]$ in $vector2 = AuthorityWeight[v]$ in $g2$
End For
Normalize $vector1$ and $vector2$
Return (inner product of $vector1$ and $vector2$)

Fig. 3 Outline of the Vector-based metric computation

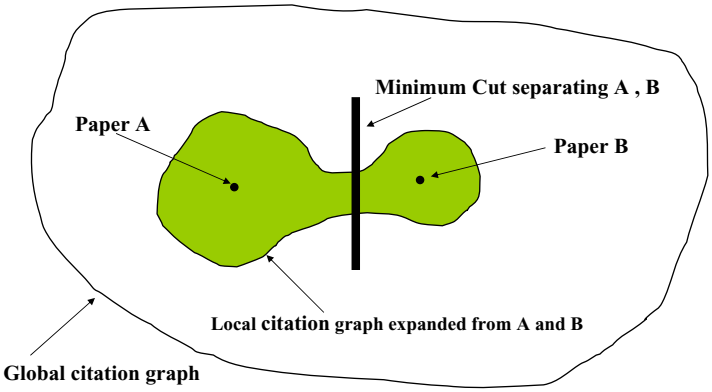

Fig. 4 Using minimum cut to compute similarity between papers

the two papers simultaneously, as shown in Fig. 4. The capacities of the edges are chosen judiciously, to represent the fact that longer paths are less indicative of similarity than shorter ones.

Finding all the possible paths between two nodes is not a simple task. We introduce a flow idea to compute similarity. All the paths between the two papers together constitute a flow. If we view the network as having one source and one sink, the amount of flow between source and sink is restricted by the number and capacity of the links. The maximum flow from source to sink can be efficiently computed and is equal to the minimum capacity of a cut of the graph between source and sink, where a cut is a set of links and nodes that disconnect the graph [10]. Intuitively the more links and the higher the capacity of links between source and sink, the more flow can be sent through, and the more links will need to be cut to disconnect the source from the sink.

We use the concept of maximum flow/minimum cut to judge the relation between two papers, as shown in Fig. 4. The more flow that can be pushed from source to sink, the higher the similarity between source and sink will be. We want

to deemphasize the effect of longer paths, which is done by adjusting the edge capacities. The capacity of an edge is the maximum flow that can be pushed through the edge. An edge which is far away from the source or sink paper gets a lower capacity weight. Therefore, edges that are in the middle of long paths will have small capacity, and therefore longer paths contribute less to the overall flow from source to sink. The detailed definition of capacity is given in Sect. 3.1.

In this metric, we do not care about the direction of each edge, since we try to find out how strongly the two papers are connected together in an undirected graph grown from the papers to be compared. This is done so that paths of all types are taken into consideration.

3.1 Capacity assignment

The minimum flow similarity metric uses a parameter d, which is used to adjust the capacities of the edges. Parameter d represents the relative importance of paths of various lengths. Large values of d will tend to emphasize the importance of short paths, while smaller values of d will tend to equalize the importance of shorter and longer paths.

Parameter d should be chosen so that d paths of length k are equivalent to one path of length $k-1$. Specifically, parameter d represents the number of paths of length two that are considered equivalent to a direct edge between two nodes. In other words, the similarity metric should give the same value for two adjacent nodes as for two non-adjacent nodes having d common neighbors. Note that in the first case, the nodes can be separated by cutting one edge, while in the second case, d edges must be cut. This suggests that the capacity of the edge in the first scenario should be d times as high as the capacity of the edges in the second scenario. In view of the above, a reasonable choice for d should depend on the maximum or average degree in the graph, or the maximum or average number of common neighbors of any pair of nodes.

The argument suggests that we should assign to each edge e a capacity of the form $c(e) \sim (1/d)^k$, where k should be related to the distance of the endpoints of e to $\{u, v\}$. Let the capacity of an edge for parameter d, $c_d : E(G^k[u, v]) \to R$, be defined as follows. For each edge $e = ww'$,

$$c_d(e) = \begin{cases} \left(\frac{1}{d}\right)^{2dist(w,\{u,v\})} & \text{if } dist(w, \{u, v\}) = dist(w', \{u, v\}) \\ \left(\frac{1}{d}\right)^{2dist(\{w,w'\},\{u,v\})+1} & \text{otherwise} \end{cases} \quad (1)$$

where $dist(w, \{u, v\})$ is the shortest distance from w to u or v, and $dist(\{w, w'\}, \{u, v\})$ is the shortest distance from w or w' to u or v.

After experimentation, we noticed that a direct link between the papers under consideration skewed the computation result. Hence we decided to set the capacity of an edge which is a direct link between source and sink papers as equal to that of a node at distance one. In other words, we treat a direct link as equally important as a path of length two which could be co-citation or reference coupling, by giving it a capacity $1/d$, and not 1, as indicated by the formula. We tested several options

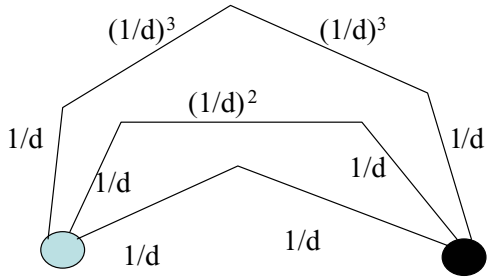

Fig. 5 Example of edge weights in flow-based metric

for the assignment of capacities in our experiments. The capacities assigned in our research are based on the length of the path or the distance between edge and source/sink papers. We mention here another option of setting edge capacities, which is based on how much you trust the edge in similarity computation. An example of edge weights is shown in Fig. 5.

For $d = 1$, paths of all lengths are given equal importance, and, by a well-known result from graph theory (see e.g. [10]), the minimum flow value equals the number of disjoint paths (of any length) between source and sink. Intuitively, the equal treatment of paths of all length does not lead to a good measure of similarity. Our results in Sect. 4.2 confirm this. Analysis of several large subgraphs of the citation graph of Computer Science literature (see [2]) shows that this graph is well-connected, and the minimum cut between two nodes will almost always fall in the neighborhood of one of the two nodes. This was also confirmed by our results for the case where $d = 1$. As shown in [20], the maximum flow in this case depends almost completely on the degree of the source or sink.

In our experiment we tested the maximum flow metric for values of d set to 50, 25, 12, 6, 3, 2, and 1 (Fig. 6). We found that the precision drops with decreasing values of d. Further, the results for $d = 25$ and $d = 50$ were almost identical. Therefore to get optimal precision d should be set to 25 or greater.

The results also indicate that the parameter d can be used to control the scope of the result set. We can let the end user choose different values of d to change the scope of the papers. For example, the user can start with a higher value of

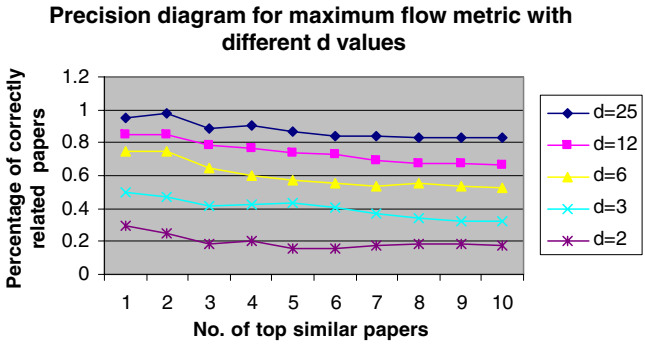

Fig. 6 Test results for the flow-based algorithm values of d equal to 25, 12, 6, 3, and 2

ComputeMaximumFlow(PaperID1, PaperID2, d)
Grow the citation graph g for paper 1 and paper 2.
Convert g into bi-directed graph
For each node n in graph g
Set $DistanceFromSource[n]$ = level no of n
End For
For all edges e in graph g
Calculate edge capacity from d and $DistanceFromSource[]$
End For
Normalize the *capacity weights* for each edge
Initialize $flow\ value$ of each edge to zero
Calculate the $maximum\ flow\ value$ from paper1 to paper2
Return ($maximum\ flow\ value$)

Fig. 7 Outline of the Flow-based metric computation

d to locate related papers in the immediate vicinity of the query paper, and then gradually reduce the value of d to locate related papers that are farther away, with the cost of getting more unrelated papers as well (since precision drops).

3.2 The algorithm

Given two papers, we consider both of them as seeds, and build a common local citation graph of k levels. We then make the graph undirected, by assuming that flow can go in either direction through an edge. We assign capacities to all edges of the local citation graph according to the formula given in Sect. 3.1. Then, using a standard maximum flow/minimum cut algorithm, we compute the maximum amount of flow that can go between the two papers. This value represents the measure of similarity between both papers. The outline of the algorithm is shown in Fig. 7.

4 Evaluation of the link-based similarity measures

Our metrics were tested on a total of 10 query papers from the field of neural networks. The choice of the neural network domain was motivated by the availability of "experts" to judge the results. For each of these papers, we found a result set of papers in the citation graph of neural network papers (constructed by crawling Citeseer), that are most related to it according to each of our similarity metrics, ranked by degree of similarity. The size of the citation graph was 109,519 nodes, out of which 23,371 nodes were fully parsed (i.e., their full text, and hence their references, was available in Citeseer). The degree of relatedness of each paper in the result set was judged by two of the authors, E.M. and N.J., who represented the domain experts. The experimental setup was blind, in that the experts received the query paper and a list of papers for evaluation returned by all metrics in random order and without repetitions. In general, our methods compared favorably to CCIDF, the graph-based method used by *Citeseer*. In this section, we describe the details of our experiments, and discuss the effect of the setting of parameters

and other implementation decisions on the results. In the next section, we will also compare our link-based similarity measures to more classical text-based ones.

All test results are shown as precision diagrams, that are obtained from the response of our domain experts. The domain experts rated each of the top 10 papers in each result set as "related," "somewhat related" or "not related," expressed by numerical values 1, 0.5, and 0, respectively. In the precision diagrams, the total score of the first k papers in a result set is plotted as a function of k. In total, over 500 papers were rated by the domain experts. The precision diagrams represent the average results over all papers considered.

We chose CCIDF as our comparative metric because it is a metric also solely based on citation information, and because it is a metric used by the *Citeseer* database. It should be pointed out that the retrieval task was chosen to demonstrate the validity of the proposed algorithms, and not to compete with more efficient information retrieval algorithms in terms of computational performance. Building an efficient information retrieval system that incorporates our algorithms is a topic for future research.

4.1 Building the local citation graph

We now describe in detail how the local citation graphs are obtained. The local citation graph will be built from one or two papers, and it will be given as input to our similarity metric computation. We use the term "layer" to represent different sets of nodes. Nodes in layer 0 only contain the starting points corresponding to the seed papers. Nodes in layer 1 are nodes citing or cited by layer 0 nodes. Nodes in layer 2 are nodes citing or cited by nodes in layer 1, but not by nodes in layer 0. In terms of edges, we say that edge e is in layer n if one endpoint of e is in layer n and the other endpoint is in layer $n - 1$. If both endpoints of an edge are in layer n, we say that the edge is in layer $n.5$. For instance, if an edge has its endpoints in layer 1 and layer 2, we call it a layer 2 edge; if both endpoints of an edge are in layer 2, we call it a layer 2.5 edge. In terms of the local citation graph, we use the maximum layer number to name the layer of a graph. For example, a 2.5 layer graph will include nodes in layer 0, 1, and 2, and edges in layer 0.5, 1, 1.5, 2, and 2.5. The process of building our local citation graph is shown in Fig. 8.

4.2 Experimental results

In the computation of the maximum flow metric, we used local citation graphs of 2.5 and 1.5 levels. For the computation of the authority metric, we used local graphs of 1.5 level. For CCIDF we used our global citation graph (a subgraph of the *Citeseer* citation graph) to compute the citation frequency of each paper. The results, based on the average precision results taken over all 10 test papers, are presented as precision diagrams in Fig. 9.

From the precision diagrams, we can see the potential of our methods to give highly accurate results. The high precision is quite remarkable, given the fact that only information about the citation graph, not about the content of the papers, is used in our methods. Moreover, the small vector and flow metrics perform substantially better than CCIDF, while the big vector metric performs similarly to

GraphBuilding(Paper1, Paper2, $MaxExpandLayer$)
Put Paper1 and Paper2 into $ProcessQueue$.
Initialize empty graph G
Set W = end element of the $ProcessQueue$
Set $ExpandLayer = 0$
While($ProcessQueue$ is not Empty)
Set P = pop one paper out of $ProcessQueue$
Create a new node for P in G // P cannot be in G
For each paper Q citing P or cited by P
if Q already in G then add edge between P and Q
else if Q is not in $ProcessQueue$ and $ExpandLayer < MaxExpandLayer$
put Q in $ProcessQueue$ // hence papers in $ProcessQueue$ are not in G
End for
If $P = W$ then
$ExpandLayer = ExpandLayer + 1$
Set W = end of $ProcessQueue$
End If
End While
Return (G)

Fig. 8 Growing the local citation graph for one or two seed papers

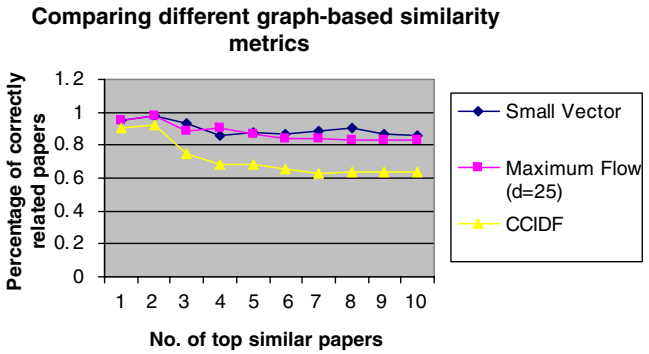

Fig. 9 Test results of different metrics

CCIDF. We demonstrate these statements by a two-factor analysis of variance (ANOVA) with replications on the family of the four metrics together, and then on all pairs of metrics. We show the F-statistic and P-value for each pairwise test in Table 1. The result of the overall ANOVA is that there is statistically significant difference among the four metrics, $F_{3,360} = 24.5$, P-value < 0.0001.

In the remainder of this section, we discuss various observations on the experimental results that may be of interest.

Table 1 Pairwise ANOVA results for the four metrics

	Small vector	CCIDF	Flow ($d = 25$)
Big vector	$F = 36.6$, $P < 0.0001$	$F = 1.07$, $P = 0.30$	$F = 28.5$, $P < 0.0001$
Small vector		$F = 51.8$, $P < 0.0001$	$F = 1.29$, $P = 0.26$
CCIDF			$F = 36.7$, $P < 0.0001$

$F = F_{1,180}$ and $P = P$-value.

4.2.1 The effect of a hub paper

When analyzing our results, we found one paper containing more than 1000 references, which was not judged to be similar to the given query paper by our metrics but which was nonetheless given a high similarity rank by CCIDF. This points to a reason why CCIDF may not give satisfactory results. CCIDF is based only on the common references between two papers. If a paper has a long list of references, it is very likely to have more papers in common with a given test paper, and hence it will receive high CCIDF value. As noted earlier, however, hub papers that are highly cited can still cause problems to our metrics, too.

4.2.2 Overlap in the result sets from the different metrics

In addition to comparing precision curves, it is interesting to examine the amount of overlap in the result sets obtained from the various metrics. For three different query papers, we found that there was an overlap between 10 and 40% in the result sets of the flow-based and vector-based metrics. Furthermore, the result sets using CCIDF have similarly small overlap with the flow-based and vector-based metric result sets. This implies that the result sets from the two metrics differ substantially, and points in the direction of possibly integrating the results from multiple metrics.

4.2.3 The effect of graph quality

A factor that was found to influence our results is the quality of the local citation graph. Some nodes in our local citation graphs correspond to papers for which the full paper, and hence also its reference list, is not available in the database. Such nodes are said to be not fully parsed. The quality of a local citation graph, in terms of its proportion of fully parsed nodes, will affect the precision of our results. To investigate the effect, we used the flow-based metric with $d = 25$ on papers with local citation graphs of different quality. We found that the percentage of fully parsed nodes in the first layer has the most influence on precision. Based on our test, it appears that at least 30% of the nodes in the first layer should be fully parsed in order to obtain dependable results [20].

4.2.4 The effect of directly linked papers

In the result set, if one paper is citing or is cited by a query paper, we say it is a directly linked paper to the query paper. In this section, we compare the percentage of directly linked papers in the result sets returned by the different metrics.

The importance of direct link has been noticed and studied in previous research in library science literature. Henry Small found that a very effective predictor of strong co-citation linkages between papers is provided by the direct citation patterns [24]. In our research we also analyze the influence of direct link on our similarity metric. It is easy to see how the existence of a direct link will affect both the values of the maximum flow metric and the authority metric. For the maximum flow metric, if there is a direct link between source and sink paper, it provides a shortcut with high capacity between source and sink that can accommodate more

flow. For the authority metric, graphs grown from two adjacent nodes will have a larger intersection than those grown from nodes which are far apart. So it is intuitive that we will obtain a certain percentage of papers with a direct link to the query paper in the result set.

We examined the percentage of direct link papers in the result sets of the various metrics. The results verify that all metrics retrieve a certain percentage of direct link papers in the range from 30% for CCIDF to 60% for the flow-based and vector-based metrics [20].

We also experimented with small adjustments to our algorithms in terms of the treatment of the direct link. For the maximum flow metric, we performed adjustments to the capacity of the direct link, while leaving the capacities of all other edges the same. Our findings were as expected, namely that we obtain more direct link papers with a higher capacity of the direct link. For the authority metric, we modified the way in which we grow the local citation graph, so as not to use direct link information in the process of growing the local citation graph. We found that this affects the percentage of direct link papers very little. This result indicates that when there is a direct link between papers, their local citation graphs have a large overlap even if the direct link itself is discarded.

4.2.5 Distance of returned papers from source paper

It is useful to know how far the result papers from each one of our metrics are from the query papers. Intuitively, the farther away from the query papers the results are, the more unexpected are the uncovered connections. We examine the average number of hops from the query papers to the result papers in Fig. 10.

We observe that that the Big Vector and CCIDF metrics have similar average distances to the Flow metric with $d = 25$. The Small Vector metric tends to retrieve papers closer to the query papers. We also observe that the average distance grows with the decrease of value of parameter d. Figure 11 shows the maximum distance in the result sets (top 10) for each metric.

The important finding is that from the flow-based metric we can get papers which are 3 hops away from the query papers when d drops to 3. Such papers

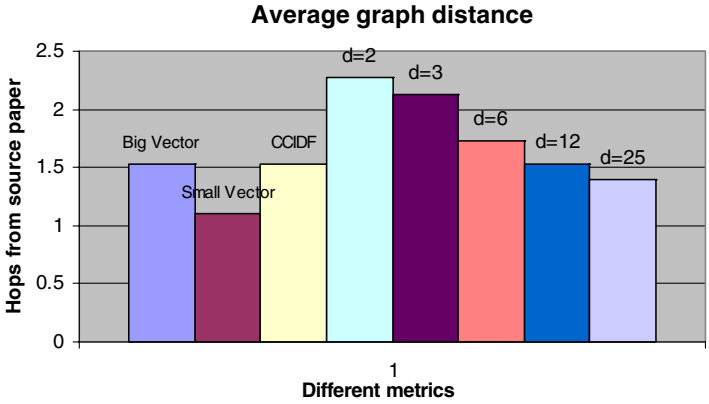

Fig. 10 Average graph distance (number of hops) of result papers from query papers

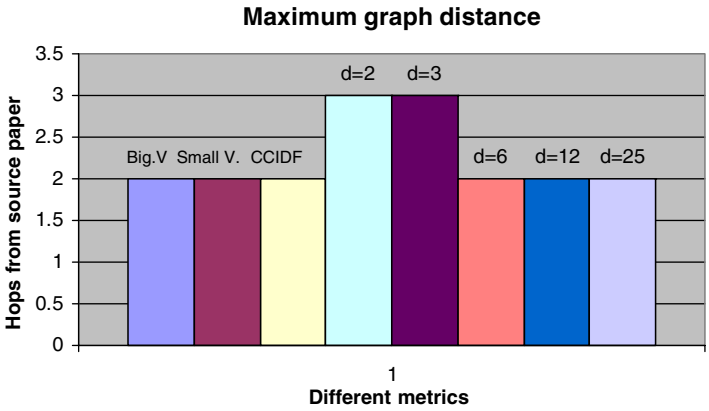

Fig. 11 Maximum graph distance of result papers from source papers

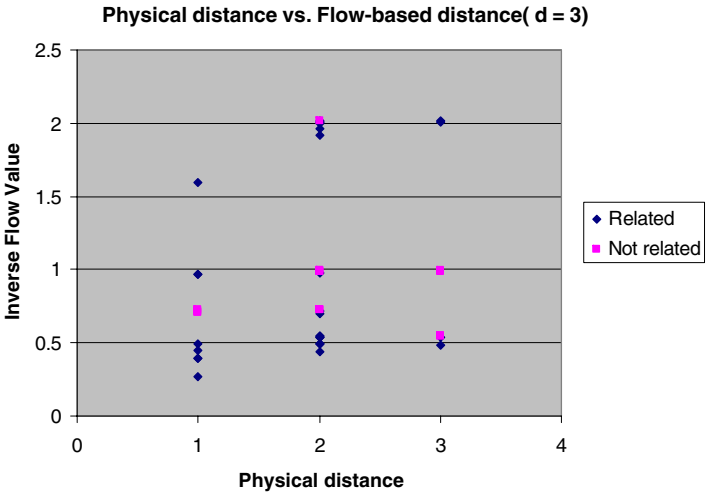

Fig. 12 Visualization of test results from flow-based metric with d set to 3

cannot be found by using CCIDF, bibliographic coupling, co-citation or direct link metric. The test result of flow with d set to 3 (Fig. 12) shows related papers found three hops away from the query papers.

4.2.6 Metrics in the formal sense

The various metrics proposed in this paper are not guaranteed to be metrics in the formal sense (in that they satisfy the triangle inequality). There are algorithms for efficient indexing using distance measures that are not metrics in the formal sense. For example, FastMap [12] is a method that maps data entities to points in a Euclidean space, given a distance measure, while preserving the distance structure of the data space.

4.2.7 Computation time

Computation time for a single query paper on a Pentium III, 700 MHz with 128 MB of RAM is about 5 h for the flow-based methods and about 2 h for the vector-based methods and CCIDF. In the vector-based methods, the vector representation of each paper in the database is computed once and stored, however no special file structures are used to quickly identify whether two papers have any overlap in their local neighborhoods (which could reduce the computation time by orders of magnitude). In the flow-based methods, the joint local citation graph is different and must be grown for each pair of papers. Design of appropriate indexing structures for quickly identifying the papers, for which the graph is connected, and therefore similarity is non-zero, is a topic of future study.

5 Comparison of link- with text-based similarity measures

In this section, we present results of the application of text-based similarity measures to the same retrieval task using the same document corpus and the same query papers that were used in the evaluation of link-based similarity measures presented in the previous section.[1] The key lesson from this comparison is the complementary nature of term/word-based and link-based methods.

Two variants of text-based similarity are presented, based on words and on multi-word terms extracted using the *C*-value/*NC*-value method [13]. The first variant uses single-word nouns as features. The second variant uses noun phrases, obtained by the application of a linguistic filter on the sequence of part-of-speech tags of the text, and treated as candidate terms. The candidate terms are further ranked on the basis of statistical metrics that account for the frequency of the noun phrases in the corpus, and nesting relations between noun phrases (*C*-value) and in addition the presence of "context" words that appear in the vicinity of candidate terms (*NC*-value).

Words or terms are used to define a *document vector space* for defining a document similarity measure. The vector space model is widely used for the measurement of similarity between documents [21] because of its conceptual and computational simplicity. Documents and queries are represented as vectors in a vector space, where the dimensions correspond to "features" (words or terms). We applied the following equation [21] to define the term weight.

$$\text{weight}(i, j) = \begin{cases} (1 + \log tf_{i,j}) \cdot \log \dfrac{N}{df_i}, & \text{if } tf_{i,j} \geq 1 \\ 0, & \text{if } tf_{i,j} = 0 \end{cases} \tag{2}$$

where $tf_{i,j}$ is the frequency of term i in document j, df_i is the number of documents in which term i occurs, and N is the total number of documents in the corpus.

Documents are ranked in the vector space model by measuring their similarities with the query vector using the standard cosine similarity metric.

[1] This research was first presented in [22].

5.1 Features used for text-based similarity

Two different methods were used to evaluate document similarity based on content, *term-based* and *word-based*.

In the term-based method, we generated the corpus terms from full papers sorted by *NC-value*. There were 189,043 candidate terms extracted from the whole corpus with a specific linguistic filter, but not all the terms are suitable for information retrieval [28]. For example, those terms appearing in most documents in the corpus are not useful, because they do not help discriminate among documents. So we re-rank the list of terms, according to their Document Frequency in order to set proper upper and lower cut-offs for selecting terms appearing with intermediate frequency [28]. Document frequency is the number of documents in which the term occurs [21]. Cut-offs were determined empirically, as is common practice in information retrieval. We specified the cut-off interval (4, 250) to exclude the most frequent and the least frequent terms, leading to a subset of 6100 terms with document frequency between 4 and 250.

In the word-based method, we extracted all the nouns from the corpus as features. The number of nouns is 11060, after setting the cut-off interval based on document frequency at (12, 8700), almost twice the number of terms.

The top 10 similar papers for each query paper were judged by two domain experts, and were assigned a score 1 (related), 0.5 (somewhat related) or 0 (not related).

5.2 Comparison of term-, word- and link-based methods

The term-based method gave on the average somewhat better precision than the word-based and link-based methods, as shown in Fig. 13.

To determine the statistical significance of the differences in precision curves, we performed a two-factor Analysis of Variance with replications on the raw scores from the above experiments. We show the F-statistic and P-value for each

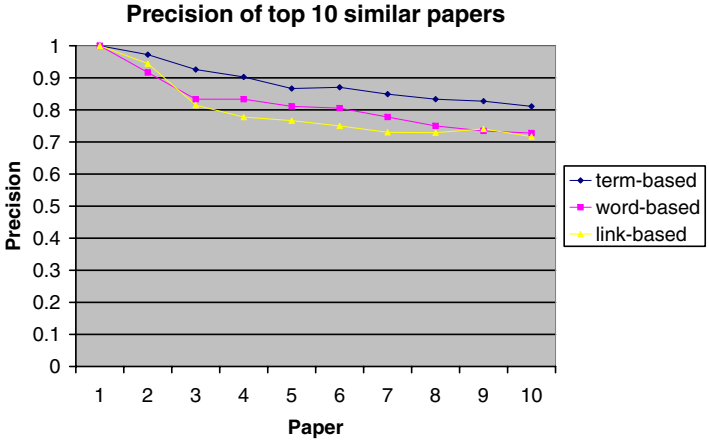

Fig. 13 Precision comparison of three methods: term-based, word-based, and link-based

Table 2 Pairwise ANOVA results for the three experiments

	Term-based	Word-based
Word-based	$F_{1,\,162} = 2.28$, P-value $= 0.13$	
Link-based	$F_{1,\,162} = 3.10$, P-value $= 0.08$	$F_{1,\,162} = 0.04$, P-value $= 0.85$

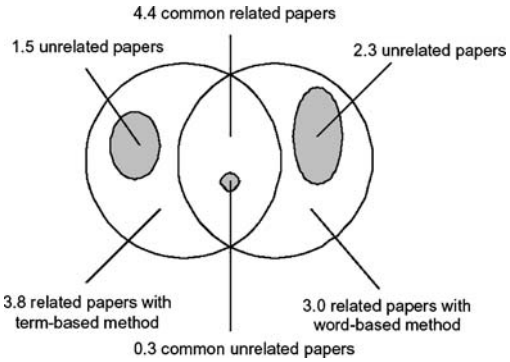

Fig. 14 Venn diagram for the complementarity of the results from the term-based and word-based methods

pairwise test in Table 2. As we can see, the significance level of the difference between the term-based and word-based methods is 87% (i.e., there is probability 13% that the observed difference came about by chance). The significance level of the difference between the term-based and link-based methods is 92% (i.e., there is probability 8% that the observed difference came about by chance). There is no significant difference between the word-based and the link-based methods.

5.3 Complementarity of methods

5.3.1 Term-based vs. word-based

The word-based and term-based methods complement each other by producing different sets of related papers as shown in Fig. 14. Averaged over the query papers, they had 4.4 relevant papers in common against the top 10 similar papers and for the remaining non-common papers, 3.8 papers were judged as relevant with term-based method and three papers were judged as relevant with word-based method.

5.3.2 Term-based vs. link-based

The link-based and term-based methods complement each other by producing different sets of related papers, as shown in Fig. 15. Averaged over the query papers, they had 2.4 relevant papers in common against the top 10 similar papers and for the remaining non-common papers, 5.3 papers were judged as relevant with term-based method and 4.8 papers were judged as relevant with link-based method.

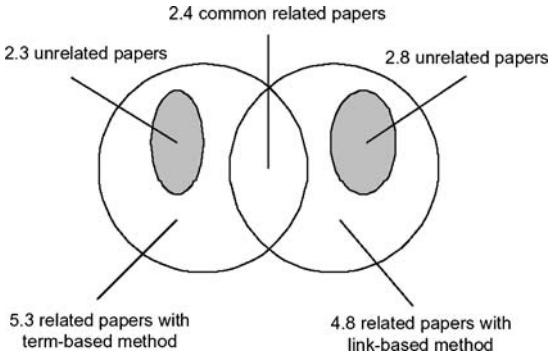

Fig. 15 Venn diagram for the complementarity of the results from the term-based and link-based methods

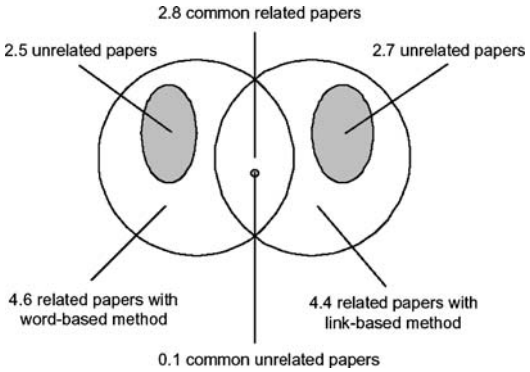

Fig. 16 Venn diagram for the complementarity of the results from the word-based and link-based methods

The term-based method can get higher precision but needs time to preprocess the texts and build an inverted index. Term- and link-based methods can be used together to gain higher precision and attract more similar papers to the top similar paper list.

5.3.3 Word-based vs. link-based

Figure 16 demonstrates that the link-based and word-based methods complement each other too by producing different sets of related papers. Averaged over the query papers, they had 2.8 relevant papers in the 2.9 common papers against the top 10 similar papers and for the remaining non-common papers, 4.6 papers were judged as relevant with word-based method and 4.4 papers were judged as relevant with link-based method.

6 Future work

In the experiments presented here, we limited the evaluation of results to the top 10 similar papers to a given query paper. It would be interesting to see whether

the conclusions presented here still hold if we considered instead a larger number of top similar papers, such as 20 or 50, especially the conclusion about similar papers found by one metric but not the others. The general problem is that reliably evaluating the similarity of hundreds of results requires substantial additional amounts of the experts' time. Making relevance judgments on research papers requires deep domain expertise beyond that of graduate students, while faculty and researcher time who possess such expertise in in short supply. A way to address the evaluation problem is to follow the text retrieval conference (TREC) model, where an organization such as the National Institute of Standards and Technology (NIST) pools resources together to create standard corpora with associated relevance judgments, on which researchers from around the world could evaluate their algorithms.

A related issue is the size of the optimal neighborhood used in the metrics. We noted that a larger neighborhood does not seem to improve retrieval performance but it may uncover unexpected relations. Defining the sense of optimal and determining the optimal neighborhood for the various metrics is an important future research topic.

In the future, we may want to take more detailed citation information into account. One direction for future work is to assign weights to the citations made in an article. Namely, citations made in the same paper have a different importance to the author or research. We might assign different weights to references in terms of where they appear (introduction, body) and how often they appear in the research paper as opposed to treating all the references equally.

Another direction for future research is to treat co-citation and reference coupling differently in computing the similarity, by assigning weights to different edges using directed graph. In this paper we do not consider the direction and treat co-citation and bibliographic coupling equally. Depending how much you trust each of these two relationship in judging relatedness, one could set up an adjustable parameter to leverage the similarity judgment.

Combining text-based metric and link-based similarity metrics is worth pursuing. One simple way is just use a weighted sum of each individual similarity measure to compute similarity. Another option is to use text-based methods to simplify the citation graph by highlighting important citations, that are identified by analyzing the text.

Finally, there are other contexts besides the citation graph in which the link-based similarity algorithms we propose here may be applicable. One example is the problem addressed in [7] of determining fraudulent telephone accounts by analyzing the calling patterns. The idea of that paper is the following: someone who has previously used a fraudulent account may set up a new one under a different name and address, but the calling patterns are largely the same. That paper extracts the local neighborhood of the calling pattern graph for each, and compares them using an ad-hoc way of computing the similarity. The authority vector metric we present here has the potential to improve upon those results, especially with respect to the problem of popular nodes (toll-free numbers, information numbers) that are common to a lot of calling patterns, but do not give much information about similarity, so they can skew the results. Similar issues appear in the domain of tracking financial transactions for detection of financial crime [27].

7 Conclusions

Our research aims to find efficient ways to judge relatedness among research papers using only citation information represented in the citation graph. Our key hypothesis is that, if two research papers are related, it should be possible to infer this from their local neighborhoods in the citation graph. Our work can be viewed as an effort to generalize and improve upon bibliographic coupling and co-citation analysis, that have been shown to reflect similarity and relatedness between papers. The maximum flow algorithm can be used to find out how strongly each pair of papers is connected in the citation graph; the authority metric measures the similarity of the local neighborhood of two papers in the citation community. From our experiments, we conclude that both metrics are promising and effective in finding related papers for scientific research. Our algorithms can be viewed as complementary to the global analysis of citation graph carried out in [2].

Acknowledgements We would like to thank Dr. Steve Lawrence from NEC Research Laboratories for providing computer science citation information and for allowing access to the *Citeseer* Database for our robot. The experiments reported in the article were carried out off-line using a citation graph built by Yuan An. We thank Joe MacInnes and Dr. Nauzer Kalyaniwalla for helpful discussions on the analysis of variance. Bin He and Li Dong helped with the implementation of automatic term extraction and the text-based similarity measures.

References

1. ACM Special Interest Group on Hypertext, H. and the Web (2002) Hypertext 2002 Conference. ACM, http://www.cs.umd.edu/ht02/ (last accessed Sept. 7, 2002)
2. An Y (2001) Characterizing and mining the citation graph of the computer science literature. Technical Report CS-2001-02, Faculty of Computer Science, Dalhousie University, Halifax, Nova Scotia, Canada
3. An Y, Janssen J, Milios E (2004) Characterizing and mining the citation graph of computer science. Knowl Inf Syst 6(6):664–678
4. Baeza-Yates R, Ribeiro-Neto B (1999) Modern information retrieval. Addision Wesley/ ACM Press, New York
5. Brin S, Page L (1998) The anatomy of a large-scale hypertextual web search engine. In: Proceedings of the 7th international world wide web conference April 1998, Brisbane, Australia, pp 107–117
6. Chen C (1999) Visualising semantic spaces and author co-citation networks in digital libraries. Inf Process Manage 35(3):401–420
7. Cortes C, Pregibon D, Volinsky C (2001) Communities of interest. In: Proceedings of the 4th international conference on advances in intelligent data analysis (IDA-2001), pp 105–114
8. Davis R, Neviett W, Foltz M (2002) Information architecture. Technical Report http:// www.infoarch.ai.mit.edu/ (last accessed Sept. 7, 2002), MIT AI Lab
9. Dean J, Henzinger MR (1999) Finding related web pages in the world wide web. In: Proceedings of the 8th international world wide web conference (WWW8), pp 389–401
10. Diestel R (2000) Graph theory, 2nd edn. Springer, Berlin Heidelberg, New York
11. Dodge M (2002), An atlas of cyberspaces: information space maps. Technical Report http://www.cybergeography.org/atlas/info_maps.html (last accessed Sept. 7, 2002)
12. Faloutsos C, Lin K-I (1995) Fastmap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets. In: Proceedings of the 1995 ACM SIGMOD international conference on management of data, San Jose, California, United States
13. Frantzi K, Ananiadou S, Mima H (2000) Automatic recognition of multiword terms. Int J Digit Libr 3(2):117–132
14. Garfield E (1955) Citation indexes for science. Science 122(3159):108–111

15. Garfield E (1972) Citation analysis as a tool in journal evaluation. Science 178(4060):471–479
16. Kaufman L, Rousseeuw P (1990) Finding groups in data: an introduction to cluster analysis. Wiley, New York
17. Kleinberg JM (1999) Authoritative sources in a hyperlinked environment. J ACM 46(5):577–603
18. Kumar S, Raghavan P, Rajagopalan S, Tomkins A (1999) Extracting large scale knowledge bases from the web. In: IEEE international conference on very large databases (VLDB), Edinburgh, Scotland
19. Lawrence S, Giles CL, Bollacker K (1999) Digital libraries and autonomous citation indexing. IEEE Comput 32(6):67–71
20. Lu W, Janssen J, Milios E, Japkowicz N (2001) Node similarity in networked information spaces. Technical Report CS-2001-03, Faculty of Computer Science, Dalhousie University, Halifax, Nova Scotia, Canada
21. Manning C, Schuetze H (1999) Foundations of statistical natural language processing. MIT Press, Cambridge, Massachusetts
22. Milios E, Zhang Y, He B, Dong L (2003), Automatic term extraction and document similarity in special text corpora. In: Proceedings of the 6th conference of the pacific association for computational linguistics (PACLing'03), Halifax, Nova Scotia, Canada, pp 275–284
23. Ng A, Zheng A, Jordan M (2001) Stable algorithms for link analysis. In: Proceedings of the 24th annual internation ACM SIGIR conference on research and devlopment in information retrieval (SIGIR)
24. Small H (1973) Co-citation in the scientific literature: A new measure of the relationship between two documents. J Am Soc Inf Sci 24:265–269
25. Small H (1986) The synthesis of specialty narratives from co-citation clusters. J Am Soc Inf Sci 37:97–110
26. Tjaden G (2002) The knowledge enterprise in information space. Technical Report http://www.ces.btc.gatech.edu/report4.html (last accessed Sept. 7, 2002), The Centre for Enterprise Systems, Georgia Institute of Technology
27. Treasury U (n.d.) Financial Crimes Enforcement Network (FinCEN). Technical Report http://www.ustreas.gov/fincen/sitemap.html (accessed Oct. 25, 2001), US Government
28. van Rijsbergen C (1999) Information retrieval. http://www.dcs.gla.ac.uk/∼iain/keith/index.htm, 2nd ed., last accessed on Apr. 17, 2002

Wangzhong Lu holds a Bachelor's degree from Hefei University of Technology (1993), and a Master's degree from Dalhousie University (2001), both in computer science. From 1993 to 1999 he worked as a developer with China National Computer Software and Technical Service Corp. in Beijing. From 2001 to 2005 he held industrial positions as a senior software architect in Atlantic Canada. He is currently with DST Systems, Charlotte, NC, as a senior data architect.

Jeannette Janssen's research area is applied graph theory. She has worked on the problem of frequency assignment in cellular and digital broadcasting networks. Her current interest is in graph theory applied to the World Wide Web and other networked information spaces. Dr. Janssen did her Master's studies at Eindhoven University of Technology in the Netherlands, and her doctorate at Lehigh University, USA. She is currently an associate professor at Dalhousie University, Canada.

Evangelos Milios received a diploma in electrical engineering from the National Technical University of Athens, and Master's and Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology. He held faculty positions at the University of Toronto and York University. He is currently a professor of computer science at Dalhousie University, Canada, where he was Director of the Graduate Program. He has served on the committees of the ACM Dissertation Award, and the AAAI/SIGART Doctoral Consortium. He has worked on the interpretation of visual and range signals for landmark-based positioning, navigation and map construction in single- and multi-agent robotics. His current research activity is centered on Networked Information Spaces, Web information retrieval, and aquatic robotics. He is a senior member of the IEEE.

Nathalie Japkowicz is an associate professor at the School of Information Technology and Engineering of the University of Ottawa. She obtained her Ph.D. from Rutgers University, her M.Sc. from the University of Toronto, and her B.Sc. from McGill University. Prior to joining the University of Ottawa, she taught at Ohio State University and Dalhousie University. Her area of specialization is Machine Learning and her most recent research interests focused on the class imbalance problem. She made over 50 contributions in the form of journal articles, conference articles, workshop articles, magazine articles, technical reports or edited volumes.

Yongzheng Zhang obtained a B.E. in computer applications from Southeast University, China, in 1997 and a M.S. in computer science from Dalhousie University in 2002. From 1997 to 1999 he was an instructor and undergraduate advisor at Southeast University. He also worked as a software engineer in Ricom Information and Telecommunications Co. Ltd., China. He is currently a Ph.D. candidate at Dalhousie University. His research interests are in the areas of Information Retrieval, Machine Learning, Natural Language Processing, and Web Mining, particularly centered on Web Document Summarization. A paper based on his Master's thesis received the best paper award at the 2003 Canadian Artificial Intelligence conference.