

# Searching and Classifying the Web Using Hyperlinks: A Logical Approach

Justin Picard and Jacques Savoy  
Institut Interfacultaire d'Informatique  
Pierre-a-Mazel 7  
2000 Neuchâtel  
Switzerland

## Abstract

As the amount of information stored on the Web increases at an amazing pace, it gets harder for search engines to retrieve the information relevant to a user. Recently, attention in the Web community has focused on the potentiality offered by hyperlinks to help index and organize information. Several methods have been proposed to take account of the knowledge induced by hyperlinks, with different objectives in mind. In this paper, we focus on three of them: improving document ranking by a propagation of document scores through the hyperlink structure, estimating the popularity of a Web page, and extracting the most important hubs and authorities related to a given topic. Using probabilistic argumentation systems, a technique for dealing with uncertain knowledge which integrate propositional logic and probability theory, we show how all these techniques can be modeled in a unified logical framework. This allows comparisons of the different methods for using hyperlinks, and illustrates some of their weaknesses. Some experiments illustrate the feasibility of the approach.

## 1 Introduction

Traditional techniques for indexing documents may be inappropriate in the context of the Internet. (1) Web pages may vary from a few words to megabytes, such that the use of word frequencies for document indexing and retrieval may produce unpredictable effects. (2) Moreover, the assumption that the words contained in a document are strong indicators of the underlying topics is not as much verified as in classical text collections (e.g. news stories, scientific papers): people who design Web pages may have specific objectives in mind, which are reflected in the words they choose. Also, tactics to influence the ranking of search engines, called "spamming" or "the search engine persuasion problem" [11], are widespread. Finally, (3) the well-known problems of polysemy and synonymy are aggravated on the Web: in a click, one can find commercial information, scientific papers, data repositories, propaganda or the Web pages of his friends. This seriously limits the use of thesaurus or domain knowledge, and it seems that relying only on the words and keywords is not a good indexing strategy for the Web.

To compensate the difficulties with indexing and retrieving information, many search engines are also offering hierarchical classifications. These classifications were created and maintained up to now by humans. However the human cost of purely manual organization of information does not make it a viable option at long term, because Web pages change continuously and new categories or "cyber communities" appear everyday on the Web. And as reported in [11], "repositories are now themselves resorting to search engines to keep their database up-to-date".

### 1.1 Hypertext links, information and knowledge

The difficulties with searching and organizing the Web have led researchers to investigate other sources of knowledge. Recently, attention has focused on one of them: the few billion hypertext links which "glue" the Internet together. The Web would after all not exist without these links, which are the paths which lead to information. Indeed, browsing is for many users the usual way to find "nearby" information, and as quoted in [11]:

”The power of the Web resides in its capability of redirecting the information flow via hyperlinks, so it should appear natural that in order to evaluate the information content of a Web object, the Web structure has to be carefully analyzed.”

Recent experiments seem to confirm that hyperlinks can be very valuable in locating or organizing information [11, 9, 2, 3, 1]. They have been used: (1) to improve an initial ranking of documents [13], (2) to compute an estimate of a Web page’s popularity [2], or (3) to find the most important hubs and authorities for a given topic [9, 1, 3]. It seems that each of these techniques is based on some underlying, sometimes partly implicit idea, on the type of knowledge which can be induced by the presence of a hyperlink between two pages. For example, when they are used to estimate the popularity of a Web page, hyperlinks can be interpreted in the following way: ”if a document is cited by a popular document, then it is possibly popular itself”. This type of knowledge can easily be captured by propositional logic, if some measure of uncertainty is associated.

## 1.2 Outline of this paper

Probabilistic argumentation systems is a technique for dealing with uncertain knowledge, by integrating propositional logic with probability theory. In this paper, we will show how the techniques mentioned in Section 1.2 to handle hypertext links can be described in this unified framework. This allows comparisons of the methods, highlights some previously unseen weaknesses and leads to new methods for using hyperlinks.

Section 2 will introduce the reader to probabilistic argumentation systems, which have already been applied to information retrieval in hypertext [12]. Sections 3,4 and 5 will describe techniques to handle hyperlinks in order to improve document ranking, estimate the popularity of a Web page, and extract the most important hubs and authorities related to a given topic. We will see how probabilistic argumentation systems can be used to deal with the knowledge induced by the hypertext structure for the same purposes. Section 6 will present some experiments with the model developed for improving an initial ranking of documents. Section 7 will conclude this paper.

## 2 Introduction on probabilistic argumentation systems

In this section, we make a short introduction to probabilistic argumentation systems (PAS). There is much more about PAS than what is shown here. For a detailed overview of PAS, the reader is referred to [6]. However, this tutorial should be sufficient to understand their application in this paper.

### 2.1 PAS

Propositional logic is one of the simplest and most convenient ways of encoding knowledge. An apparent drawback is that pure propositional logic seems to be unsuitable for taking account of uncertainty. However, uncertainty can be handled rather easily by adjoining particular propositions called **assumptions**. Assumptions are propositions which state the unknown conditions or circumstances upon which the facts and rules depend. If an assumption is known to be true, then the fact or rule which depends on it holds. Otherwise, nothing can be deduced from this fact or rule.

For example, let proposition  $D_1$  denote ”document  $d_1$  is relevant to the information need”. Proposition  $D_1$  can be either true or false. There might be some uncertainty associated to this fact, for example it may depend on the reliability of the search engine which has retrieved it for a given query. By using an assumption  $a_1$  which denotes the uncertain conditions under which the fact  $D_1$  holds, the uncertainty can be captured by:  $a_1 \rightarrow D_1$ <sup>1</sup>. Similarly, uncertainty in rules can also be captured by assumptions. For example, suppose that documents  $d_1$  and  $d_2$  concern similar topics such that in some cases, they are both relevant to the same information need. These conditions which do not always apply can be captured by the following:  $l_{12} \rightarrow (D_1 \rightarrow D_2)$ , where  $D_2$  means ”document  $d_2$  is relevant”, and  $l_{12}$  denotes the uncertain circumstances under which the rule  $D_1 \rightarrow D_2$  applies. We will in general prefer the following equivalent notation for uncertain rules:  $D_1 \wedge l_{12} \rightarrow D_2$ .

---

<sup>1</sup>For reading commodity, assumptions will be denoted by minor letters, and other propositions by capital letters

Most applications also require a numerical assessment of uncertainty. The numerical aspect of uncertainty is obtained by assigning probabilities to assumptions. For example, if for the uncertain rule  $D_1 \wedge l_{12} \rightarrow D_2$ , the condition  $l_{12}$  is known to hold with probability 0.3, then we may set:  $p(l_{12}) = 0.3$ . Note that this is conceptually different from assigning a probability to the whole logical sentence ( $p(D_1 \rightarrow D_2) = 0.3$ ), as is done in other frameworks for integrating uncertainty with logic.

Given a knowledge base composed of uncertain facts, rules or conditions modeled with logical formulas containing assumptions, we are interested in finding which symbolic **arguments** support or discard a given hypothesis  $h$ . A symbolic argument  $h$  is a conjunction of literals of assumptions which, if added to the knowledge base, makes the hypothesis true (a literal is a proposition or its negation). We will then compute the **symbolic support** of  $h$  given by the knowledge base  $\xi$ , denoted  $sp(h, \xi)$ , which contains the disjunction of all symbolic arguments which allow to derive  $h$  if added to the knowledge base. We may also want to evaluate the reliability of the support given by arguments, using probabilities assigned to the assumptions. We will then compute the **degree of support**  $dsp(h, \xi) = p(sp(h, \xi))$ .

## 2.2 An example

For example, take the knowledge base:

$$\xi = (a_1 \rightarrow D_1) \wedge (a_2 \rightarrow D_2) \wedge (D_1 \wedge l_{12} \rightarrow D_2) \quad (1)$$

Remark that a knowledge base can always be represented as a conjunction of rules, facts, and more generally of clauses. We are interested in finding the arguments for hypothesis  $D_2$  given by  $\xi$ . It is easily seen that  $a_2$  is an argument for  $D_2$ , because  $(a_2 \rightarrow D_2) \wedge a_2 \models D_2$ . The same way,  $(a_1 \wedge l_{12})$  is another argument for  $D_2$ . Thus, the symbolic support given by the knowledge base for  $D_2$ , denoted  $sp(D_2, \xi)$ , is:

$$sp(D_2, \xi) = a_2 \vee (a_1 \wedge l_{12}) \quad (2)$$

The following probabilities are assigned to the assumptions:  $p(a_1) = 0.4, p(a_2) = 0.25, p(l_{12}) = 0.3$ . What is the probability that the support holds? In order to make an exact computation, independence assumptions must be made, e.g.  $p(a_1 \wedge a_2) = p(a_1) \cdot p(a_2), p(a_1 \wedge \neg a_2) = p(a_1) \cdot (1 - p(a_2))$ , etc. The degree of support of  $D_2$  given by the knowledge base,  $dsp(D_2, \xi)$ , is:

$$\begin{aligned} dsp(D_2, \xi) &= p(sp(D_2, \xi)) \\ &= p(a_2 \vee (a_1 \wedge l_{12})) \\ &= p(a_2 \vee (a_1 \wedge l_{12} \wedge \neg a_2)) \\ &= p(a_2) + p(a_1) \cdot p(l_{12}) \cdot (1 - p(a_2)) \\ &= 0.34 \end{aligned}$$

The passage from line 2 to line 3 in the previous equation comes from the equality:  $A \vee B = A \vee (B \wedge \neg A)$ . Next sections will show how the notions on PAS presented here can be applied to deal with the knowledge induced by the hypertext structure.

## 3 Using hyperlinks to modify document score and rank

### 3.1 Spreading activation in hypertext

The implicit reasoning made in the spreading activation technique is as follows: a link from a document  $d_1$  to a document  $d_2$  is evidence that their content is similar or related, such that if  $d_1$  is relevant to a certain information need,  $d_2$  may also be relevant. Having an initial ranking of documents given by the retrieval system for a given query, the links can be used to improve the ranks of the documents which are linked to the best ranked documents. For example, if  $d_2$  is linked to  $d_1$  which is ranked first, then  $d_2$  should be placed at a better rank.

The general scheme is to take an initial ranking and to re-rank it as follows. The retrieval engine provides an initial retrieval status value (RSV) to each document based on its similarity with the query. It is assumed that the links have not been used to produce this RSV. The RSV of document  $d$  is updated by adding the weighted RSV of its  $m$  neighbors through a certain number of cycles. The neighbors can be linked by incoming but also outgoing documents. Suppose that document  $d$  has neighbors  $d_1$  to  $d_m$ . The RSV of  $d$  at cycle  $i + 1$  is computed by the following:

$$RSV(d^{i+1}) = RSV(d^i) + \sum_{j=1}^m \lambda_j RSV(d_j^i) \quad (3)$$

The parameter  $\lambda_j$  can be seen as the degree of certainty regarding the evidence provided by the link from  $d_j$  to  $d$ . It can be a fixed value according to the link type <sup>2</sup>, or may vary according to a measure of similarity between the documents and the query [5, 14].

The underlying assumption of a repeated propagation through a certain number of cycles is that documents may have direct but also indirect influences on each other through the links. Indeed, if the RSV of a document depends on the RSV of each of its neighbors, their RSV depend in turn on the RSV of their neighbors, and so on. In a way, this repeated propagation can be seen as an inference process, though with no guarantee that the inferences are always appropriate. However, the number of cycles is often limited to one: more than one cycle is usually harmful to retrieval effectiveness [14].

Several problems can be found with this heuristic technique: there is no theoretical background guiding the choice of the number of cycles  $c$  or the value of the parameter  $\lambda_j$ . Moreover, evidence may propagate indefinitely if there are cycles in the network.

### 3.2 Improving document ranking with PAS

With the PAS model developed here, we will also attempt to improve document ranking using the hypertext structure. But instead of propagating document scores, we will seek all symbolic arguments supporting the relevance of a document. In a second phase, probabilities are assigned to the assumptions and the degree of support given by the arguments is computed. Finally documents are returned to the user by decreasing degree of support.

For each document  $d_i$ , let us denote proposition  $D_i$  as: "document  $d_i$  is relevant". If a document is retrieved by the retrieval system, it is evidence in favor of that document's relevance. Let assumption  $a_i$  denote, "the retrieval system has correctly retrieved document  $d_i$ ". Then for each document in the collection, we have:

$$a_i \rightarrow D_i \quad (4)$$

For a given query, we may adjust the probability of the assumption  $p(a_i)$  to the rank at which  $d_i$  is retrieved ( $p(a_i|rank)$ ), or set  $p(a_i) = 0$  (i.e.  $\neg a_i$ ) if  $d_i$  is not retrieved. In practice, we fit a logistic regression on the rank for a set of training queries, as we will see in Section 6.

We want to use the hypertext structure to improve the initial ranking. For each link from  $d_i$  to  $d_j$ , we induce the knowledge that under some condition  $l_{ij}$ , the relevance of  $d_i$  implies the relevance of  $d_j$ . The assumption  $l_{ij}$  may denote the conditions under which the link implies relevance in the present context. We have then:

$$D_i \wedge l_{ij} \rightarrow D_j \quad (5)$$

We may then find the arguments supporting the relevance of each document. As an example, take a collection of documents  $d_1, d_2, d_3$ . There are links from  $d_2$  to  $d_1$  and from  $d_3$  to  $d_1$ . The following knowledge base is generated:

$$\xi = (a_1 \rightarrow D_1) \wedge (a_2 \rightarrow D_2) \wedge (a_3 \rightarrow D_3) \wedge (D_2 \rightarrow D_1) \wedge (D_3 \wedge D_1) \quad (6)$$

We find for the support of  $D_1$ :

$$sp(D_1, \xi) = a_1 \vee (a_2 \wedge l_{21}) \vee (a_3 \wedge l_{31}) \quad (7)$$

<sup>2</sup>There can be links of various types: hypertext links, citation, nearest neighbor, etc. Also, links can be distinguished by their orientation (incoming, outgoing), because this orientation can affect the amount of information about relevance contained in the link.

Here,  $d_1$  has three symbolic arguments. For a real query, one may want a numerical evaluation. The degree of support of  $D_1$  is:

$$\begin{aligned} dsp(D_2, \xi) &= p(sp(D_1, \xi)) = p(a_1) + p(a_2 \wedge l_{21} \wedge \neg a_1) + p(a_3 \wedge l_{31} \wedge \neg a_1 \wedge \neg(a_2 \wedge l_{21})) \\ &= p(a_1) + p(a_2) \cdot p(l_{21}) \cdot (1 - p(a_1)) + p(a_3) \cdot p(l_{31}) \cdot (1 - p(a_1)) \cdot (1 - p(a_2) \cdot p(l_{21})) \end{aligned}$$

This computing formula for the degree of support is the same for all queries. For a given query, one needs to give values to the  $p(a_i)$ 's according to the rank of  $d_i$ , and probabilities for the links  $p(l_{ij})$ . However, the former can be fixed.

It is interesting to notice that the hypertext structure, interpreted logically, has been integrated in the computing formulas for the degree of support of each document. This way, the computations can be done very fast. Section 6 will show some experiments comparing the spreading activation technique to the PAS model developed in this section, and demonstrate how probabilities can be computed in practice.

## 4 Estimating the popularity of a Web page

### 4.1 PageRank

The PageRank algorithm, used in the Google search engine ([www.google.com](http://www.google.com)) considers that users have an absolute preference among Web pages: it assumes that the more a Web page is visited, the more it is appreciated by the users. To measure the popularity of the pages, it is not possible to have access to the logs of the servers, but a reasonable assumption is that the preference of users is reflected in the hypertext structure: a link toward a Web page is often an indication that this page is acknowledged by someone as a good source of information. A simple way to implement this idea would be to count the number of times a Web page is cited. Microsoft's home page, surely one of the most visited page on the Web, is cited more than 23 million times in Altavista's index (probably much more in reality). However, each link should not be treated equally, since its impact also depends on the popularity of the parent node: a page cited only a few times but which is in Yahoo!'s index would certainly be quite visited. Thus the popularity of a page also depends on the popularity of the pages that cite it.

The idea behind PageRank is that a user who crawls the Web by selecting the hyperlinks at random is more likely to visit certain Web pages than others, simply because there are more possible ways by which the user can reach these pages. It is possible to model the behavior of a "random" surfer as a Markov process, where the states of the system are each of the Web pages. The measure of popularity of a Web Page, its PageRank, is given by the stationary probabilities of this Markov process - the limit probability that the user will be on a certain page. The PageRanks are computed very simply by an iterative algorithm, which converges after a few steps.

The popularity measures are used in the Google search engine to boost the scores of the documents, independently of the query. This algorithm is criticized because it biases the access to information [10]. The "perverse" effect of PageRank is that it will push popular pages to get even more popular, and new or unknown (unlinked) Web pages to stay unknown. As said in [11], "visibility is likely to be a synonym of popularity, which is completely different than quality, and thus using it to gain higher score is a rather poor choice". To our advice, the frequency at which a page is visited by all the users of the Web is not necessarily an indicator of its relevance to a user who has its own preferences, hobbies, cultural background, etc. The PAS modeling offers a clean way to take account of these a priori preferences of the user.

### 4.2 Using PAS to compute a personalized "popularity" measure

Suppose that for each user, it is possible to compute some personalized "popularity" measure. For example, each user may produce a set of keywords or Web pages reflecting its interests, such that each page on the Web can be assigned a probability  $p(a_i)$  that it will be among the favorites given its similarity with the keywords or the given Web pages. We would like to refine this knowledge on the user's interest by taking account of the hypertext structure. Let us define  $P_i$

as "document  $d_i$  corresponds to the user's interest". Then for each document, there is some condition  $a_i$  under which  $d_i$  corresponds to the user's interest. Similarly as before, we have:

$$a_i \rightarrow P_i \tag{8}$$

The probability  $p(a_i)$  can be initially computed if the user gives some information about his interests, and then updated by keeping track of the pages he visits. We want to use the hypertext structure to improve our knowledge on the user's interests. For each link from  $d_i$  to  $d_j$ , we induce that, under some condition  $l_{ij}$ , the relevance of  $d_i$  implies the relevance of  $d_j$  to the user's interest. We have then:

$$P_i \wedge l_{ij} \rightarrow P_j \tag{9}$$

Note that the symbolic support that a document is among the favorite to the user is the same as the support that it will be relevant. Indeed each user will have the same symbolic arguments supporting the popularity of each document. Then if an equal probability  $p(a_i)$  is assigned to each document, we will be in the case where it is assumed that the user has no preference. This is the case of PageRank. However if the user gives some hints allowing to compute personal a priori probabilities  $p(a_i)$  or eventually link probabilities, it will be possible to have a personal ranking for this user. It should be noted that in PageRank, it is not possible to assign different initial probabilities on the Web pages.

## 5 Finding hubs and authorities

### 5.1 Kleinberg's algorithm

In many cases, the user does not know what exactly he is looking for, and is rather interested in having good starting points for browsing in order to learn general information on the domain. Given a general topic sufficiently represented on the Web (e.g. "human rights", "ice hockey"), it is possible to distinguish two types of potentially relevant pages: **authorities** and **hubs**. Authorities are pages which contain high quality and exhaustive information on the topic, and hubs are pages which contain links to the authorities, thus giving access to the relevant information. The Web is very rich in central pages, fan sites and other classifications of resources, and those can be very helpful for automatic classification of information.

How can we find hubs and authorities? The assumption made by Kleinberg [9] is that a good authority is a page which has links from many good hubs, and a good hub is a page which has links towards many good authorities. The algorithm has some similarity with PageRank in that the quality of a page depends recursively on the quality of the neighbors, although here the links are followed in both directions. The idea of Kleinberg's HITS algorithm [9] is to consider a root set of usually 200 documents, composed of the most likely relevant documents to a given topic. These 200 documents are found with a traditional search engine. This root set is expanded with all documents which point to or are pointed by these pages, to form the base set in which authorities and hubs will be found (the expansion can be done twice to have a larger base set). Then the connectivity of this base set is used as follows to find the best hubs and authorities. For each document  $d_p$  in the base set, a hub score  $h_p$  and a authority score  $a_p$  are computed. The initial scores are set to 1, but the final result is not sensitive to any non degenerate values of initial scores. Then the hub and authority scores are updated iteratively, by respectively the sum of authority scores of the pages that cite  $d_p$ , and the sum of hub scores of the pages that  $d_p$  cites. The updating equations are:

$$h_p = \sum_{d_p \rightarrow d_i} a_i$$

$$a_p = \sum_{d_i \rightarrow d_p} h_i$$

where  $d_p \rightarrow d_i$  means that there is a link from  $d_p$  to  $d_i$ . It can be shown that the scores will converge if they are normalized after each step. The exact scores are not so important, since the user is presented with a ranked list of hubs and authorities:

It is argued that the algorithm has an "objective" justification because it finds some intrinsic properties of a set of linked pages [9, 3]. However, we believe there are some weaknesses in this algorithm. One is that a base set has to be chosen for a given topic, among which the hubs and authorities will be selected. Although it is argued in [8] that the method is robust (i.e. gives similar results) for different base sets, the choice of a particular base set is nonetheless purely heuristic. Another is that the initial ranking of documents is not used as prior evidence, while clearly an initially better ranked document has more chances to be relevant, and certainly more chances to be a good hub or authority.

## 5.2 A model for computing hub and authority scores

Given a general search topic and a document  $d_i$ , proposition  $H_i$  denotes "document  $d_i$  is a good hub" and  $A_i$  denotes "document  $d_i$  is a good authority". Unlike Kleinberg's algorithm, we consider that there is initial evidence  $h_i$  that  $D_i$  is a good hub, and  $a_i$  that it is a good authority.

$$h_i \rightarrow H_i, a_i \rightarrow A_i \quad (10)$$

As in Kleinberg's algorithm, we make the assumption that if a document  $d_i$  is cited by a good hub  $d_j$ , then this is evidence that  $d_i$  is a good authority. We have then:

$$H_j \wedge f_{ji} \rightarrow A_i \quad (11)$$

Similarly, if a good authority  $d_i$  is cited by a document  $d_j$ , then this is evidence that that  $d_j$  is a good hub:

$$A_i \wedge g_{ij} \rightarrow H_j \quad (12)$$

For each hyperlink from  $d_j$  to  $d_i$ , there will be two rules generated:  $(H_j \wedge f_{ji} \rightarrow A_i), (A_i \wedge g_{ij} \rightarrow H_j)$ . From this knowledge base  $\xi$ , one can compute for each document  $d_i$ , the symbolic support from  $\xi$  that it is a good hub and a good authority,  $sp(H_i, \xi)$  and  $sp(A_i, \xi)$ . Then, for a given topic, different probabilities are assigned to the assumptions  $p(a_i)$  and  $p(h_i)$ , and eventually to the assumptions  $f_{ij}$  and  $g_{ij}$ , which can be fixed or depend on some similarity value with the topic. The numerical degrees of support  $dsp(H_i, \xi)$  and  $dsp(A_i, \xi)$  will be the hub and authority scores of document  $d_i$  for this topic. Note that compared with Kleinberg's algorithm, there is no need to determine a base set, which is here the same for all topics. For two different topics, only the assigned probabilities will change.

## 6 Experiments with the model for spreading activation

In this subsection, we present our experiments done on the CACM and Trec'8 Web test collections. If the paper is accepted, we should be able to present experimental results on another test collection of bigger size. The CACM test collection (3.2 Megabytes) contains 3204 documents, and has 50 requests with associated relevance judgments. The Web Track collection (2.3 Gigabytes) contains around 250'000 pages extracted from the Web, and has 100 requests with associated relevance judgments. The retrieval effectiveness is evaluated by the mean average precision at 11-point recall, the most used evaluation measure in information retrieval. Section 6.1 describes briefly the summary of necessary operations, and Section 6.2 presents some experiments made on two test collections.

### 6.1 Learning

The set of symbolic operations is as follows:

- Convert each link from  $d_i$  to  $d_j$  to  $D_i \wedge l_{ij} \rightarrow D_j$ .
- For each document  $D_i$ , compute the support  $sp(D_i, \xi)$ .
- Put the support of each document in disjoint form using Heidtmann's algorithm [7]. Convert this disjoint form to a directly usable formula for computing the degree of support  $dsp(D_i, \xi)$  (see e.g. Section 4).

Type of link	SA	PAS	PAS vs baseline	PAS vs SA
Citing	0.266 (0.3)	0.267	+5.57%	+0.01%
Cited	0.261 (0.4)	0.273	+7.83%	+4.11%

Table 1: Experimental results on the CACM collection

Model	Baseline	Best incoming	Best outgoing	Combined
okapi-npn	0.267	0.267 (0.00%)	0.267 (0.00%)	0.267 (0.00%)
lnu-ltc	0.234	0.239 (+2.18%)	0.240 (+2.65%)	0.239 (+2.10%)
atn-ntc	0.257	0.260 (+1.44%)	0.261 (+1.52%)	0.260 (+1.32%)
ntc-ntc	0.138	0.139 (+0.14%)	0.139 (+0.14%)	0.138 (0.00%)
ltc-ltc	0.136	0.138 (+0.88%)	0.138 (+0.88%)	0.138 (+0.88%)
lnc-ltc	0.107	0.108 (+0.65%)	0.109 (+1.49%)	0.107 (+1.31%)
lnc-lnc	0.072	0.074 (+2.35%)	0.073 (+1.38%)	0.073 (+1.52%)
anc-ltc	0.082	0.084 (+2.31%)	0.087 (+5.59%)	0.084 (+2.79%)
nnn-nnn	0.071	0.072 (+0.56%)	0.072 (+0.42%)	0.070 (-0.98%)
bnn-bnn	0.096	0.100 (+4.18%)	0.101 (+5.02%)	0.099 (+3.56%)

Table 2: Experimental results on the WT collection

At this point, all the logical operations are done. A learning process is also necessary to assign probabilities to the assumptions  $p(a_i)$  and  $p(l_{ij})$ . For the first ones, a set of training queries is used to fit a logistic regression on the probability of relevance given the rank. The probabilities of link assumptions can be static or may depend on some similarity value between the link and the search topic. For these experiments, we will only consider static probabilities. They can also be estimated on a set of training queries. The reader is reported to [12, 15, 16] for more details.

When a query is processed, the only operation remaining is to compute the degree of support for each document  $d_i$ ,  $dsp(D_i, \xi)$ .

## 6.2 Experiments

For the CACM collection, links were considered separately in the forward direction (citing) and in the backward direction (cited). For each document, arguments of order three and less were computed, and the symbolic support was put in disjoint form with Heidtmann’s algorithm.

The basic retrieval process was done using a classical retrieval system based on the cosine similarity measure. The baseline retrieval effectiveness is 0.253, computed with the TRECEval software. Comparisons were made between PAS and baseline, and PAS and spreading activation (SA). The results shown for spreading activation are the best ones obtained for a range of values of the parameter  $\lambda$ , which was fixed for all links of a certain type. The best  $\lambda$  value is shown between parenthesis. Score was spread for only one cycle, since more than one was harmful to retrieval. The results are shown in Table 1. For the PAS technique, arguments of length three or less (at most three links assumptions) were computed.

For the WT collection, ten different weighting schemes were used (okapi-npn, ..., bnn-bnn, see Table 2), to produce ten different rankings of documents. Table 2 shows the retrieval effectiveness using only the best incoming links, the best outgoing links, and both. There are slight but generally not significant improvements over the baseline.

These results show that PAS can compete with an established technique such as spreading activation. Also, indirect neighbors can be considered without depreciating performance, at least for one test collection. For citing links, there is no average difference in retrieval effectiveness, while for cited links, the difference is nearly significant. However, for spreading activation there is no understanding of the parameters involved, while in the PAS framework, parameters are the probabilities of the links which have a clearer meaning both from a statistical and a logical viewpoint. This leaves much improvement for the PAS technique.

## 7 Discussion

In this paper, we have shown how various approaches to the use of hyperlinks in order to search and organize the Web can be modeled using a single theoretical framework. Experiments have validated the approach when links are used to improve an initial ranking, but experiments for computing hub and authorities, and a personalized popularity measure have not been conducted yet because there is no existing test collection for these purposes. The experiments conducted so far in these cases do not respect the standards necessary to draw reliable conclusions, and building a good quality test collection is a very costly and demanding task. But the information retrieval community is tackling the task of building a good quality Web test collection, and experiments should be possible in the next future.

With the several methods that have been proposed to use hyperlinks for different purposes, there is a need for a formal framework to allow analysis and comparisons between them. We have shown one possible framework here, which uses the commodity of propositional logic for translating intuitions into knowledge. With the use of PAS, we have gained some insight on those intuitions: for example, the similarity between the computation of popularity measures and improving an initial ranking have been shown, and we have seen how some heuristic aspects of Kleinberg's algorithm could be described

## References

- [1] K. Bharat and M. Henzinger. Improved algorithms for topic distillation in hyperlinked environments. In *Proc. of the Int. ACM-SIGIR Conf.*, pages 104–111, 1998.
- [2] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the World Wide Web Conference*, pages 107–117, 1998.
- [3] S. Chakrabati, M. Van der Berg, and B. Dom. Focused crawling: A new approach to topic specific resource discovery. In *Proceedings of the World Wide Web Conference*, pages 545–567, 1999.
- [4] S. Chakrabati, B. Dom, P. Raghavan, S. Rajagopalan, D. Gibson, and J. Kleinberg. Automatic resource compilation by analyzing hyperlink structure and associated text. In *Proceedings of the World Wide Web Conference*, 1998.
- [5] H.P. Frei and D. Stieger. The use of semantic links in hypertext information retrieval. *Information Processing & Management*, 31(1):1–13, 1995.
- [6] R. Haenni, J. Kohlas, and N. Lehmann. Probabilistic argumentation systems. Technical Report 99-09, Institute of Informatics, University of Fribourg, 1999.
- [7] K.D. Heidtmann. Smaller sums of disjoint products by subproduct inversion. *IEEE Transactions on Reliability*, 38(3):305–311, 1989.
- [8] J. Kleinberg. Authoritative sources in a hyperlinked environment. Technical Report RJ 10076(91892), IBM, May 1997.
- [9] J. Kleinberg. Authoritative sources in a hyperlinked environment. In *Proceedings of the 9th Symposium on Discrete Algorithms*, pages 668–677, 1998.
- [10] S. Lawrence and C.L. Giles. Accessibility of information on the web. *Nature*, 400(8):107–109, 1999.
- [11] M. Marchiori. The quest for correct information on the Web: Hyper search engines. In *Proceedings of the World Wide Web Conference*, 1997.
- [12] J. Picard. Modeling and combining evidence provided by document relationships using probabilistic argumentation systems. In *Proceedings of the International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 182–189, Melbourne, Australia, 1998.

- [13] J. Savoy. A learning scheme for information retrieval in hypertext. *Information Processing & Management*, 30(4):513–533, 1994.
- [14] J. Savoy. Ranking schemes in hybrid Boolean systems: A new approach. *Journal of the American Society for Information Science*, 48(3):235–253, 1997.
- [15] J. Savoy and J. Picard. Report on the TREC-8 experiment: Searching on the web and in distributed collections. In D. Harman, editor, *TREC'8*, Gaithersburg (DC), 1999.
- [16] J. Savoy and Y. Rasolofo. Report on the TREC-9 experiment: Link-based retrieval and distributed collections. In D. Harman, editor, *TREC'9*, Gaithersburg (DC), 2001.