A Comparative Evaluation of Different Link Types on Enhancing Document Clustering

Xiaodan Zhang, Xiaohua Hu and Xiaohua Zhou

College of Information Science and Technology, Drexel University

3141 Chestnut Street, Philadelphia, PA 19104, USA

{xzhang,thu}@ischool.drexel.edu, xiaohua.zhou@drexel.edu

ABSTRACT

With a growing number of works utilizing link information in enhancing document clustering, it becomes necessary to make a comparative evaluation of the impacts of different link types on document clustering. Various types of links between text documents, including explicit links such as citation links and hyperlinks, implicit links such as co-authorship links, and pseudo links such as content similarity links, convey topic similarity or topic transferring patterns, which is very useful for document clustering. In this study, we adopt a Relaxation Labeling (RL)-based clustering algorithm, which employs both content and linkage information, to evaluate the effectiveness of the aforementioned types of links for document clustering on eight datasets. The experimental results show that linkage is quite effective in improving content-based document clustering. Furthermore, a series of interesting findings regarding the impacts of different link types on document clustering are discovered through our experiments.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Clustering

General Terms

Algorithms, Experimentation, Performance

Keywords

Link-based clustering, Markov Random Field

1. INTRODUCTION

The links between documents are considered as very useful information for text processing because they usually encode external human knowledge beyond document contents. PageRank [18] and HITS [8] are two very successful models using such linkage information for document importance ranking.

Exploiting link information to enhance text classification has been studied extensively in the research community [1] [3] [4] [7] [17]. Most of these studies fall into two frameworks. One is referred to as relaxation labeling (RL) in which the label of a document is determined by both local content and its neighbors' labels [3]. The other improves classification accuracy by

incorporating neighbors' content information text into the local content. However, Ghani [7] et al. discovered that neighbors' text content information could be useful only when the neighbor link structure exhibits encyclopedia regularity.

Moreover, a growing number of works [10, 15, 16, 20, 21 and 22] used hyperlink information in the clustering of web search results. Whereas these approaches provide valuable insights on employing link information, they all rely on heuristic similarity measures, which linearly combine text similarity information with link similarity or co-citation similarity information. However, how to set up the parameter for a linear combination is really data dependent and requires a great deal of tuning. Furthermore, the findings from only clustering web search results may not work for other datasets.

We adopt a relaxation labeling (RL) based clustering algorithm to evaluate the effects of various link types in document clustering. Relaxation labeling is initially designed to handle link-based text classification. It incorporates both text and link information into a unified probabilistic framework, and has been proved very effective in text classification [1, 3 and 17]. Relaxation labeling requires some seed documents, i.e. documents with labels. In the setting of text classification, training documents can serve as seeds. For document clustering purpose, we use a content-based clustering tool to initialize labels of all documents. We argue that relaxation labeling would iteratively utilize the linkage information to improve the initial clustering. Angelova and Siersdorfer [2] applied this method to enhance traditional text clustering and achieved very positive results. However, they only studied the undirected linkage, and they did not consider many other factors such as different neighborhoods settings, pure-link effects, etc. Moreover, the existing findings should be evaluated on more types of links. All these reasons encourage us to deeply study the behavior of linkages in text clustering problems within the RL framework.

The types of link studied include explicit links, implicit links, and pseudo links. Explicit links such as hypertext and citations usually encode topic transition patterns. Implicit links often indicate the similarity of the corresponding documents. For instance, two documents by the same author should have an implicit link denoting the topic similarity of these two documents. A pseudo link is constructed as long as the content similarity between two documents is over a threshold. Similarity links as pseudo links have proven to be useful for text summarization [6], but their effectiveness in the setting of clustering is still unclear.

We conduct clustering experiments on eight data sets with three different types of link information including one set from DBLP[1], one in WebKB(http://www.cs.cmu.edu/~webkb/), two from CORA[14], and four others using pseudo link information

(TDT2_10, LATimes10, 20NG, and Reuters10). We make the following comparative studies: (1) link-based clustering using explicit, implicit or pseudo links vs. content-based clustering; (2) pure link-based clustering vs. pure content-based clustering; (3) uniform priors vs. empirical priors; (4) the effects of different neighborhood; (5) the effects of thresholding and scaling. Our main findings are: (1) link-based clustering performance is significantly better than content-based clustering except for pseudo links; (2) different link types affect clustering differently; (3) uniform priors is better than empirical priors for clustering; (4) out-neighbors of a document have more impacts on clustering than in-neighbors; (5) thresholding and scaling have negative or neutral effects on clustering.

The rest of the paper is organized as follows: Section 2 describes related work. Section 3 shows the proposed method. In section 4, we present and discuss experimental results. Section 5 concludes the paper.

2. RELATED WORKS

In recent years, link-enhanced text classification and clustering has received more and more attentions from the text mining community. We make a brief review of the methods that integrate linkage between documents.

A group of researchers have examined the hyperlink effects in text classification [1, 3 and 17] and text clustering [2] using MRF-RL-based methods. Chakrabati et al [3] found that the method is very effective in improving text classification without using neighbors' text. Ghani et al. [7] discovered that neighbors' text is helpful when a document is similar to most of documents it connects to, namely, an "Encyclopedia" scenario. Later on, MRF-RL-based methods have been applied in [17, 1] and [2] for link-based document classification and clustering, respectively. All of them used some heuristics such as thresholding to refine link graphs, making the link pattern closer to the "Encyclopedia" scenario. However, improperly omitting links between documents may cause serious information loss and thus distort the clustering results.

There are also some works [4, 13 and19] that studied the use of hyperlinks from different angles with regard to text classification. Slattery and Mitchell [19] employed FOIL (First Order Inductive Learner), a relational learner to exploit the relational structure of the web, and a Hubs & Authorities style algorithms [9] to exploit the hyperlink topology. Lu and Getoor [13] used aspect models for link-based classification. Cohan and Hoffman [4] applied a factorized model to combine the link model and the content model. However, these generative linear models require optimizing the parameters that determine how much links should affect clustering, which is very challenging.

Moreover, a group of studies developed some heuristic similarity metrics that linearly combined link information with text information for clustering web search results [10, 15, 16, 20, 21, and 22]. Modha and Spangler [16] proposed an algorithm called TORIC k-means that clusters hypertext documents using words, in-links and out-links. Similarly, [25] represents each document using the combination of three vectors: in-link vector, out-link vector and text vector. Each cluster is annotated using six information nuggets: summary, breakthrough, review, keywords, citation, and reference. In [10], the similarity measure includes three types of information: hyperlink structure, textual information and co-citation pattern. Works [22] and [15] combined shortest path and content similarity information to enhance text classification. Moreover, Halkidi et al. [9] claimed that a page's classification is enriched by the detection of its incoming links' semantics. All these approaches rely on heuristic similarity metrics using both text and link information. As discussed earlier, these approaches are limited to web search results, more data dependent, and requiring a lot of tuning.

Therefore, it becomes necessary to make a comprehensive study on how different link types affect text clustering within an objective framework. That is why we conduct this study.

3. THE CLUSTERING METHOD
3.1 Basic Model

Pelkovwitz [12] developed an algorithm for labeling a Markov Random Field defined on an arbitrary finite graph. Later on, it was applied to text classification by Chakrabarti et al [3]. We use the generative model from [3] to describe the link-based document clustering process. Let $D = \{d_i, i = 1, 2, ..., n\}$ be a document set and $e_{i \to j}$ be the directed links from d_i to d_j; let $T = \{\tau_i\}$ representing the entire collection of text corresponding to D. Each τ_i is a sequence of $\{w_i \mid i = 1, 2, ..., k\}$ tokens; let $C = \{c_i\}$ be the set of class assignments for the entire collection D. Assuming that there is a probability distribution for collection D, we choose a class assignment C such that $\Pr(C \mid G, T)$ is the maximum. As $\Pr(G, T)$ is not a function of C, it is sufficient to choose a C to maximize $\Pr(G, T \mid C)\Pr(C)$.

$$\Pr(C \mid G, T) = \frac{\Pr(G, T \mid C)\Pr(C)}{\Pr(G, T)} \approx \Pr(G, T \mid C)\Pr(C) \qquad (1)$$

We believe that the class labels of all documents neighboring document d_i can form an adequate representation of d_i's neighborhood. A usual Bayes classifier is then employed to update a document' class label based on its immediate neighbors' class labels such that a c_i is chosen to maximize $\Pr(c_i \mid N_i)$, where N_i represents all the known class labels of the neighbor documents of d_i. Similarly, since $\Pr(N_i)$ is not a function of c_i, maximizing $\Pr(c_i \mid N_i)$ is equal to maximizing

$$\Pr(N_i \mid c_i)\Pr(c_i) \qquad (2)$$

Thus, when we assume there is no direct coupling between the texts of a document and the classes of its neighbors, equation (1) can be rewritten as:

$$\Pr(G, T \mid C)\Pr(C) = \Pr(N \mid C)\Pr(T \mid C)\Pr(C) \qquad (3)$$

In equation (2), N_i can be further decomposed into in-neighbors I_i and out-neighbors O_i. A class prior $\Pr(c_i)$ is the frequency of class c_i from content-based clustering results. Notice that $\Pr(c_i)$ is not a true prior as there is no true class label for text clustering. So using empirical $\Pr(c_i)$ from the content-based clustering process may distort the clustering results. This problem will be further discussed in the experimental section. Given the class label of the current document, based on Markov network's assumption, all the neighbor class labels (including in-neighbors and out neighbors) are independent of each other (see equation (4)).

$$\Pr(N_i \mid c_i)\Pr(c_i)$$
$$= \Pr(c_i)\prod_{d_j \in I_i} \Pr(c_j \mid c_i, e_{j \to i})\prod_{d_k \in O_i} \Pr(c_k \mid c_i, e_{i \to k}) \qquad (4)$$

3.2 Iterative Relaxation Labeling

From the discussions above, a class label that maximizes equation (5) is chosen for a document:

$$c_i = \arg\max_{c_i} \log\{\Pr(N_i \mid c_i)\Pr(\tau_i \mid c_i)\Pr(c_i)\} \qquad (5)$$

The initial class label assignment is simply the output of a content-based clustering. However, a one step re-estimation that uses equation (5) may not achieve the global optimization. Therefore, each document is iteratively labeled based on its neighbors' class labels of the previous iteration until there is no change of label assignments or a pre-defined iteration number is reached.

3.3 Content-based clustering

Content-based clustering (based on documents' text content only) is important to link-based clustering since: (1) the initialization of link-based clustering is based on a content-based clustering; (2) using content information to label a document is also part of the probability framework (see term $\Pr(\tau_i \mid c_i)$ in equation (5)).

Algorithm: Model-based K-Means
Input: dataset $T = \{\tau_1,...,\tau_n\}$, and the desired number of clusters k.
Output: trained cluster models $\Lambda = \{\lambda_1,...,\lambda_k\}$ and the document assignment $C = \{c_1,...,c_n\}, c_i \in \{1,...k\}$
Steps:
1. Initialize document assignment C.
2. Model re-estimation: $\lambda_i = \arg\max_{\lambda} \sum_{\tau \in c_i} \log(\tau_i \mid \lambda)$
3. Sample re-assignment: $c_i = \arg\max_j \log p(\tau_i \mid \lambda_j)$
4. Stop if a pre-defined number of iterations is reached or if C does not change, otherwise go to step 2

Figure 1: Model based k-means algorithm

Theoretically, any content-based clustering algorithms can be used for the initialization of a link-based document clustering. However, when estimating the probability of term $\Pr(\tau_i \mid c_i)$, for a complete probabilistic approach, it is natural to employ a multinomial model instead of other non-probabilistic methods like spherical k-means.

Recent studies [23, 25] showed that model-based k-means clustering performs slightly worse than spherical k-means clustering. In this paper, we compare both schemes in our experiments. As indicated in [23, 25], smoothing techniques have a big impact on model-based document clustering algorithms; background collection smoothing outperforms Laplacian smoothing because Laplacian is only used for anti zero probability whereas background smoothing considers a word either generated from one of K cluster models $p_{ml}(w \mid c_i)$ or a background collection model $p(w \mid C)$ (see equations (6)(7)).

$$\log p(\tau \mid c_i) = \sum_{w \in V} c(w, \tau) \log p_b(w \mid c_i) \qquad (6)$$

$$p_b(w \mid c_i) = (1-\alpha)p_{ml}(w \mid c_i) + \alpha p(w \mid C) \qquad (7)$$

Therefore, we apply background smoothing techniques to both model-based k-means clustering and link-based k-means clustering.

3.4 Link-based Clustering

The proposed link-based clustering algorithm is described in figure 2. The entire clustering procedure is as follows. First, we run spherical k-means or model-based k-means until it converges. Then, we take the output class assignments of step 1 as the input of the relaxation labeling process. Next, the class model based on document content is re-estimated. Later on, the class label of each document is re-estimated based on its neighbors' class labels and its own content using equation (5). Last, the algorithm stops if it reaches a fix point or a pre-defined iteration number, otherwise it repeats model re-estimation and relaxation labeling (step 3 and 4).
Neighborhood Definition

As shown in equation (4), by default, the neighborhood of a given document is defined as its immediate in-neighbors and out-neighbors. In practice, the neighborhood definition can be more flexible. We may consider a radius-2 neighborhood, which can also include the neighbors of neighbors of a document. For instance, if we only consider the immediate out-neighbors of a document, equation (4) will be replaced by equation (8). But if we also include out-neighbors of out-neighbors of a document, then equation (8) can be rewritten as equation (9). We claim that the study the effects of the neighborhood-ranges can give us a global picture of different link structures' impacts on document clustering.

$$\Pr(N_i \mid c_i)\Pr(c_i) = \Pr(c_i)\prod_{d_k \in O_i} \Pr(c_k \mid c_i, e_{i \to k}) \qquad (8)$$

where document d_i's neighbors includes its immediate out-neighbor d_k

$$\Pr(N_i \mid c_i)\Pr(c_i) = \Pr(c_i)\prod_{d_k \in O_i} \Pr(c_k \mid c_i, e_{i \to k})\prod_{d_l \in O_k} \Pr(c_l \mid c_k, e_{k \to l}) \qquad (9)$$

where document d_i's neighbors includes not only its each immediate out-neighbor d_k, but also each immediate out-neighbor d_l of d_k.

Algorithm: Link-based K-Means
Input: dataset $T = \{\tau_1,...,\tau_n\}$, and the desired number of clusters k.
Output: trained cluster models $\Lambda = \{\lambda_1,...,\lambda_k\}$ and the document class assignment $C = \{c_1,...,c_n\}, c_i \in \{1,...k\}$
Steps:
1. Repeat content-based clustering such as spherical K-Means or model-based K-Means clustering until it reaches a fix point.
2. Initialize document assignment C using output class label assignment from step1.
3. Model re-estimation: $\lambda_i = \arg\max_{\lambda} \sum_{\tau c_i} \log(\tau_i \mid \lambda)$
4. Iteration Labeling using equation (5):
$c_i = \arg\max_{c_i} \log\{\Pr(N_i \mid c_i)\Pr(\tau_i \mid c_i)\Pr(c_i)\}$
where $\Pr(\tau \mid c_i) = \log p(\tau_i \mid \lambda_{c_i})$ (equation (5)(6))
5. Stop until a pre-defined iteration number is reached or if C does not change, otherwise go to step 3

Figure 2: Link-based k-means algorithm

3.5 Issue of Convergence

Chakrabarti et al [3] explained that the RL algorithms can find a fix point as long as the initialization confirms the link structure. However, the optimization of an initialization itself should be an active research topic, especially for clustering. In our experiments, our algorithm converged for most of the runs. For some runs, the algorithm did not converge, but the number of changing labels was decreased to a very small digit—usually less than 10, after 10 iterations. For clustering documents of over 1000, this is acceptable.

4. EXPERIMENTAL RESULTS

4.1 Dataset

4.1.1 WebKB4

The WebKB4 dataset contains web pages about university computer science departments. There are around 8,300 documents and they are divided into seven categories: student, faculty, staff, course, project, department and other. There are around 11,000 hyperlinks between these documents. Among these seven categories, student, faculty, course and project are the four most populous entity-representing categories. The associated subset is typically called WebKB4 (Table 1). For each document, the HTML tags are removed because these tags have negative effects on content-based clustering.

4.1.2 CORA—CORA7 and CORA18

Cora [14] is an online archive of computer science research papers. The archive was built automatically using a combination of smart spidering, information extraction, and statistical text classification from online papers in postscript format. These papers are then categorized into a Yahoo-like topic hierarchy with approximately 30,000 papers and over 1 million links to roughly 200,000 distinct documents. We selected two subsets of the Cora database: all 7 classes under the machine learning category (CORA7) and all 18 classes under the artificial intelligence category (CORA18) which includes CORA7 (Table 1).

4.1.3 TDT2_10, LATimes10, Reuters10 and 20NG

Pseudo link-based clustering experiments are conducted on four datasets: TDT2, LA Times (from TREC), Reuters-21578 and 20-newsgroups (20NG). The TDT2 corpus has 100 document classes, each of which reports a major news event. LA Times news is labeled with 21 unique section names, e.g., Financial, Entertainment, Sports, etc. Reuters-21578 contains 21587 documents covering 135 economic subcategories. The 20-Newsgroups dataset is collected from 20 different Usenet newsgroups, 1,000 articles from each. We selected 7,094 documents in TDT2 that have a unique class label, 18,547 documents from the top ten sections of the LA Times, 9467 documents from the top 20 categories of Reuters-21578 by excluding documents with multiple labels and all 19,997 documents in 20-newsgroups. The ten classes selected from TDT2 are *20001, 20015, 20002, 20013, 20070, 20044, 20076, 20071, 20012,* and *20023*. The ten sections selected from LA Times are *Entertainment, Financial, Foreign, Late Final, Letters, Metro, National, Sports, Calendar,* and *View*. The twenty classes from Reuters are the top twenty categories: *earn, acq, crude, trade, money-fx, interest, money-supply, ship, sugar, coffee gold, gnp, cpi, cocoa, jobs, copper, reserves, grain, alum,* and *ipi*. All 20 classes of 20NG are selected for testing.

4.1.4 DBLP3

The DBLP3 [1] dataset includes approximately 16000 scientific publications chosen from the DBLP database including three categories: "Database" (DB), "Machine Learning" (ML), and "Theory". These papers are labeled based on the conference where they were published. We use a paper title as document content and co-authorship information as an undirected link between documents (Table 1).

Table 1. Link Data sets

Datasets	# of classes	# of docs	#of in links	# of out links
WebkB4	4	4190	4104	6837
CORA7	7	4263	17072	16287
CORA18	18	10811	39318	37750
DBLP3	3	16809	359232	

4.2 Evaluation Methodology

Cluster quality is evaluated by three extrinsic measures: *F-score* [24], *purity* [24], and *normalized mutual information (NMI)* [25]. However, because of the space limitation, in some experiments, we only publish the result of the NMI, an increasingly popular measure of cluster quality. NMI is defined as the mutual information between the cluster assignments and a pre-existing labeling of the dataset normalized by the arithmetic mean of the maximum possible entropies of the empirical marginals, i.e.

$$NMI(X,Y) = \frac{I(X;Y)}{(\log k + \log c)/2} \tag{10}$$

where X is a random variable for cluster assignments, Y is a random variable for the pre-existing labels on the same data, k is the number of clusters, and c is the number of pre-existing classes. NMI ranges from 0 to 1. The bigger the NMI, the higher the quality of the clustering is.

4.3 Experimental Settings

In the following sections, we evaluate the performance of the spherical k-means, model based k-means and link-based k-means. As discussed in [23] [25], spherical k-means using the TFIDF (Term frequency * inverse document frequency) scheme always has the best performance, compared to TF (Term Frequency) and NormTF (normalized term frequency) schemes. It is also slightly better than model-based k-means. Similarly, we find link-based k-means using the outputs of spherical k-means TFIDF scheme (spherical link-based k-means) has the best performance on most of the datasets. Therefore, if not explicitly mentioned, we use it as the basic scheme.

Table 2. Clustering Schemes

Schemes	Explanation
spk_tfidf	Spherical k-means using TFIDF
spk_tfidf_l	Link k-means using spk_tfidf for initialization
mk_bkg	Model-based k-means using background smoothing
mk_bkg_l	Link k-means using mk_bkg smoothing for initialization

Table 2 shows the explanations of these scheme symbols. For example, mk_bkg stands for model-based k-means using multinomial model with background smoothing. The coefficient of background smoothing is set to 0.5, which means the probability of a word of a given document being generated from one of *K* class models is 50 % and from a background collection model is also 50 %. In our experiments, the coefficient of background smoothing is set to 0.5, which achieves the best performance for most of the datasets. Since the result of k-Means

clustering varies with the initialization, we run it ten times with random initializations and take the average as the result. During the comparative experiment, each run has the same initialization values.

In the following experimental result tables and figures, the symbols ** and * indicates the change is significant according to the paired-sample T-test at the level of p<0.01 and p<0.05, respectively. In terms of neighborhood settings, unless explicitly mentioned, a document's neighborhood is its immediate in- and out- neighbors. We use the dragon toolkit [26] to implement the corresponding algorithms and experiments.

4.4 Effects of Explicit Links

4.4.1 HyperLinks

We choose WebKB4 dataset to examine the effects of hyperlinks because it is in fact a vivid web community (computer science departments) where web pages from different classes are interwoven together. As shown in Table 3, the improvement of link-based k-means over spherical k-means is evaluated as significant by three evaluation measures—Fscore, Purity and NMI, whereas the improvement of mk_bkg_l over mk_bkg is trivial. But, we also notice that the performance of mk_bkg is much better than that of spk_tfidf, and very close to that of spk_tfidf_l. This indicates that the improvement of hyperlink-based k-means over content-based clustering is dependent on the performance of content-based clustering. Furthermore, this can also be arisen from the fact that the link graph of WebKB4 is sparse (see table 1) and therefore the influence of linkage on clustering is limited. However, the limited links still improves clustering significantly for one clustering method. Hyperlinks contain the most complicated patterns among the three compared link types because there are no strict requirements on the links. Here is a helpful example to explain this. The content of a student's homepage can be very close to a professor's if they have similar interests. Based only on content-based clustering, they are in the same cluster such as faculty cluster. However if the student's homepage also connects with many more students' homepages, it can be assigned to student cluster.

Table 3. WebKB4 Link-based vs. content-based clustering

	spk_tfidf	spk_tfidf_l	change	mk_bkg	mk_bkg_l	change
Fscore	0.485	0.524	+7%*	0.672	0.679	+1%
Purity	0.663	0.693	+4%*	0.687	0.690	+0%
NMI	0.328	0.367	+11%**	0.371	0.375	+0%

4.4.2 Citation Links

In table 4 and 5, citation link-based k-means significantly outperforms spherical k-means clustering and model-based k-means clustering (mk_bkg). For example, spk_tfidf_l has a 15% performance increase over spk_tfidf on CORA7 and a 17% on CORA18. For both datasets, spherical citation link-based kmeans(spk_tfidf_L) has the best performance. Moreover, note that the performance of mk_bkg is worse than spk_tfidf on both datasets and the same pattern holds for their corresponding link-based k-means. This infers that the improvement is dependent on not only the citation links, but also the output of the corresponding content-based clustering. Compared to hyperlinks, citation links are more helpful in improving content-based clustering performance. This can be resulted from that citation links usually have a stronger indication of relatedness than

hyperlinks since a scientific paper is usually serious about their choice of references and these references are often related to each other.

Table 4. CORA7 link-based vs. content-based clustering

	spk_tfidf	spk_tfidf_l	change	mk_bkg	mk_bkg_l	change
Fscore	0.607	0.647	+6%**	0.472	0.521	+9%**
Purity	0.614	0.656	+6%**	0.528	0.568	+7%**
NMI	0.379	0.448	+15%**	0.267	0.331	+19%**

Table 5. CORA18 link-based vs. content-based clustering

	spk_tfidf	spk_tfidf_l	change	mk_bkg	mk_bkg_l	change
Fscore	0.467	0.521	+10%**	0.404	0.441	+8%**
Purity	0.520	0.583	+11%**	0.466	0.509	+8%**
NMI	0.384	0.462	+17%**	0.348	0.401	+14%**

4.5 Effects of Implicit Links—Co-authorship

Table 6 shows the experimental results of implicit link-based (co-authorship) clustering. Compared to other link types, co-authorship link-based k-means clustering achieves a very significant improvement over its corresponding content-based clustering. Compared to that of hyperlinks and citation links, the improvement is the biggest. For example, the NMI score dramatically grows from 0.310 (spk_tfidf) to 0.677, a 54% increase. The main reason is that author tends to write papers on related topics and therefore the linkage can be a strong indication of similarity.

Table 6. DBLP3 link-based vs. content-based clustering

	spk_tfidf	spk_tfidf_l	change	mk_bkg	mk_bkg_l	change
Fscore	0.718	0.908	+21%**	0.719	0.855	+16%**
Purity	0.733	0.910	+19%**	0.733	0.860	+15%**
NMI	0.310	**0.677**	+54%**	0.338	**0.600**	+44%**

4.6 Pseudo Link vs. Content-based Clustering

Similarity links as Pseudo links between sentences was proved to be effective in text summarization [6]. It is interesting to evaluate their impacts on text clustering application. More importantly, the findings can be very indicative for clustering documents without explicit (citation linkage) and implicit (co-authorship) linkage information. In this experiment, we build similarity links between the text contents (vector of words) of two documents. The threshold is set to 0.4. Although we have tried other thresholds, the results are more or less the same, which indicates only very small amounts of links have effects on clustering.

Table 7. Pseudo link-based vs. content-based clustering

	Purity			NMI		
	spk_tfidf	spk_tfidf_l	change	spk_tfidf	spk_tfidf_l	change
TDT2	0.895	0.891	-0%*	0.726	0.719	-1%*
Reuters	0.513	0.516	+1%	0.514	0.517	+1%
LATimes	0.459	0.468	+2%**	0.344	0.357	+4%**
20NG	0.525	0.526	+0%	0.548	0.550	+0%

In table 7, we compare similarity link-based clustering with content-based clustering. Note that similarity link-based k-means clustering performs very similarly to content-based k-means with slight improvements on the LATimes. This is consistent with our observation that only a small number of documents' labels were changed for each run on four datasets. Therefore, these label changes do not have a big enough impact on affecting the clustering results. Moreover, pseudo links are based only on content-based similarity measures that do not contain external human knowledge that found in implicit and explicit links.

4.7 Comparison of Different Neighborhoods

Originally, MRF theory is built on undirected graphs. However, if a directed link is taken as an undirected link, then this link will be double-counted in the iteration labeling process. Moreover, a document's out-neighbors may have different impacts on clustering than its in-neighbors. Take a citation network as an example, a document's out-neighbors should be considered more important than its in-neighbors because an author of a scientific theory paper usually cites related theory papers while his or her paper can be cited by other applied science papers on different topics. Therefore, we differentiate a document's out-neighbors from its in-neighbors. Furthermore, we explore the radius-2 neighborhoods' effects. We argue that this comparison of different neighborhoods can be very indicative to both text clustering and other related applications.

Table 8. Neighborhood settings

Symbol	Explanation
I	Immediate in-neighbors only
O	Immediate out-neighbors only
I_O	Immediate in-neighbors and Immediate out-neighbors
II	Immediate in-neighbors and their in-neighbors
IO	Immediate in-neighbors and their out-neighbors
OO	Immediate out-neighbors and their out-neighbors
OI	Immediate out-neighbors and their in-neighbors
II_O	II and Immediate out-neighbors
IO_O	IO and Immediate out-neighbors
I_OI	Immediate in-neighbors and OI
I_OO	Immediate in-neighbors and OO
II_OO	II and OO
IO_OO	IO and OO
II_OI	II and OI
IO_OI	IO and OI

As shown in table 8, we present fifteen different neighborhood settings with I_O as the default setting (See equation (2)). Among these settings, there are three radius-1 neighborhood settings: I, O, I_O; the remaining are radius-2 neighborhood settings.

4.7.1 Hyperlinks (Explicit)

In figure 3, neighborhood settings including the immediate out-neighbors and their out-neighbor expansions (O, OO and I_OO) have a very slightly better performance than that of other settings. We also observe that all neighborhood settings are significantly better than spherical k-means using TFIDF scheme (spk_tfidf) and no one setting is significantly better than the others. This can be due to the trade-off between in- and out- neighbors and the relatively sparse connectivity of WEBKB4 dataset.
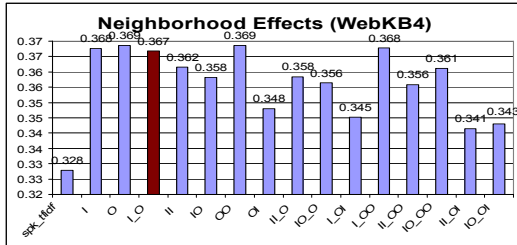

Figure 3: WebKB4: neighborhood effects on link-based clustering. Please refer to table 8 for the neighborhood symbols.

4.7.2 Citation links (Explicit)

Here, we only present the experimental results on CORA7 as they are very similar to that of CORA18. In figure 4, observe that with immediate in-neighbors (I) only or immediate out-neighbors (O) only, link-based k-means clustering achieves very comparable results to that of I_O. Furthermore, with O only is better than with I only. One main reason can be that a paper usually cites related papers while it can be cited by many other papers from various topics. We also find that OO, OI, I_OO and I_OI have much better performance than II, IO, II_O and IO_O. This shows that expanding a document's immediate out-neighbors is more helpful than expanding its in-neighbors for document clustering. Especially, the inclusion of out-neighbors of its immediate in-neighbors (IO, IO_O and IO_OO] is the worst scheme. For other radius-2 settings including II_OO, IO_OO and IO_OI, note that there are compensations between in-neighbors and out-neighbors and the results are comparable to the baseline settings (I_O).
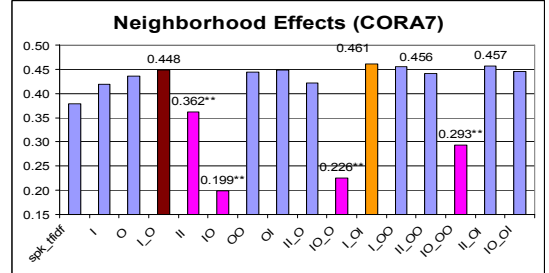

Figure 4:   CORA7: neighborhood effects on link-based clustering. Please refer to table 8 for the neighborhood symbols.

Table 9. DBLP3 neighborhood effects

	spk_tfidf_L_O	spk_tfidf_L_OO	Change
Fscore	0.908	0.883	-2%*
Purity	0.910	0.887	-2%*
NMI	0.677	0.633	-7%**

4.7.3 Co-authorship links (Implicit)

Since co-authorship graph is an undirected graph, all immediate neighbors of a document are treated as its immediate out-neighbors (O). Similarly, OO is used to represent a document's immediate neighbors and its immediate expansions. In table 9, OO performs worse than O in terms of all three metrics. This indicates that including the radius-2 expansion on a co-authorship graph has no positive impacts.
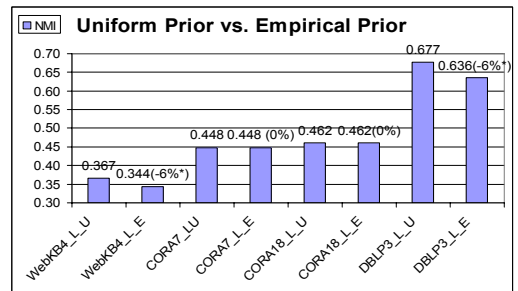

Figure 5. **Uniform priors vs. empirical priors.** The dataset name ending with "_L_U" means link-based clustering using uniform priors while the dataset name ending with "_L_E" indicates link k-means using empirical priors.

4.8 Uniform Priors vs. Empirical Priors

Theoretically, according to the basic model in section 3, we should use empirical priors ($\Pr(c_i)$) for RL, which are re-

calculated iteratively. However, there are no true priors to rely on for text clustering. The initial document labels of link-based k-means are based on content-based clustering algorithms, which may contain much noise. Thus, empirical priors may hurt the performance of link-based clustering and uniform priors may be a better choice. Therefore, we compare the performance of both priors. In figure 5, for link-based k-means with uniform priors and with empirical priors, there are no significant difference discovered on CORA7 and CORA18. However, with empirical prior, link-based k-means performs significantly worse than that with uniform priors on WebKB4 (-6 %*) and DBLP3 (-6 %*). This is consistent with our findings during the experiments. For instance, there are some clusters containing no documents for 7 out of 10 runs on WebKB4 dataset. All these findings confirm that uniform priors provide a more stable performance than empirical priors for MRF plus RL in text clustering. Therefore, we employ link-based k-means clustering using uniform priors for all the other experiments.

4.9 Pure Link vs. Pure Content

Departing from equation (5), it should be interesting to study link clustering based on only pure links. In this case, the text factor $\Pr(\tau_i | c_i)$ in equation (5) is not considered during the iterative labeling process. In table 10, the symbol "spk_tfidf_pl" means that it uses pure link-based k-means clustering.

From the experimental results in Table 10, pure link- based clustering performs very poorly on the WebKB4 dataset, which indicates that the complicated hyperlinked structure is not good itself for the relaxation labeling process; the content of web pages appear to be more crucial for hyperlink-based k-means clustering. Pure link-based clustering on CORA7 and CORA18 achieves very similar performance to content-based clustering. This shows that the more "careful" citation links, compared to the more "noisy" hyper links, are good for the global labeling optimization.

Table 10. Pure link-based vs. content-based clustering

	Purity			NMI		
	spk_tfidf	spk_tfidf_PL	change	mk	mk_bkg_PL	change
WebKB4	0.663	0.333	-99%**	0.328	0.066	-396%**
CORA7	0.614	0.562	-9%**	0.379	0.353	-1%
CORA18	0.520	0.504	-3%	0.384	0.381	-7%**
DBLP3	0.733	0.811	+10%	0.310	0.479	+35%**

Pure link-based clustering on DBLP3 does have a big improvement over content-based clustering (+10% and +35). This implies that co-authorship links have a very strong indication of the similarity of two documents.

4.10 Thresholding and Scaling Effects

In this subsection, we evaluate how heuristics such as thresholding and scaling affect link-based document clustering. Thresholding is filtering out links between two documents whose similarity value is below a pre-defined threshold. The scaling strategy is to scale equation (5) using the similarity score between two documents if there is a link between them:

$$c_i = \arg\max_{c_i} \log\{\Pr(N_i | c_i)\Pr(\tau | c_i)\Pr(c_i) \bullet S_{i,j}\} \quad (11)$$

where $S_{i,j}$ is the similarity score between document i and its immediate neighbor document j.

The motivation behind these heuristics is to filter out "irrelevant" links and to emphasize the effects of "useful" links [1, 2, and 17]. However, we argue that ignoring certain link information

improperly can cause information loss and therefore impose negative impacts to link-based clustering. Moreover, using heuristic scores such as the weights between two documents to scale the basic model may adversely affect the entire probabilistic model.
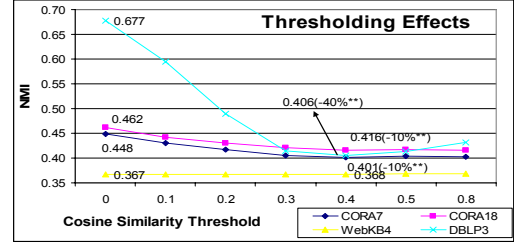

Figure 6. **Thresholding effects.** We set cosine similarity thresholds to 0, 0.1, 0.2, 0.3, 0.4, 0.5 and 0.8 respectively. "0" means there is no thresholding. If the threshold is set to other values such as 0.5, the link-based clustering will not use links below the threshold. If the threshold is set to 0.5, links below 0.5 will be filtered out and then the cosine similarity score will be used to scale the basic model.

In Figure 6, we observe a sharp performance decrease when increasing the similarity thresholds on two citation datasets. The clustering performance drops about 10 percent on both datasets from thresholds 0.0 to 0.4. An even more apparent pattern is observed on DBLP3 dataset. Increasing the threshold from 0.0 to 0.4 causes a 40% huge drop. This shows thresholding is a bad scheme for citation link and co-authorship link structures. One main reason is that the citation and co-authorship link structures contain very indicative reference relationships between documents, and therefore thresholding easily leads to a serious information loss. Moreover, observe that thresholding has no significant influence on WebKB4 dataset. These findings strongly indicate that ignoring links has negative or neutral impacts on document clustering.
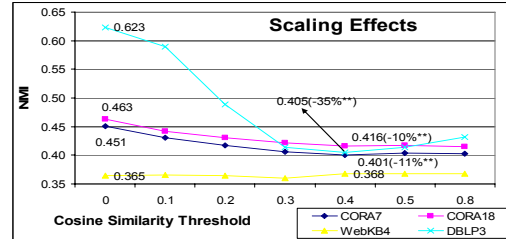

Figure 7. **Scaling Effects.** We set cosine similarity thresholds to 0, 0.1, 0.2, 0.3, 0.4, 0.5 and 0.8 respectively. "0" means scaling the basic spherical link-based k-means (spk_tfidf_l) without thresholding. If the threshold is set to 0.5, links below 0.5 is filtered out; then the cosine similarity scaling is applied to the remaining links during the re-estimation process (see equation 11).

Table 11. The comparison between spk_tfidf_l and spk_tfidf_l_scaling without thresholding

	spk_tfidf_l	spk_tfidf_l_scaling	change
CORA7	0.488	0.451	+0%
CORA18	0.462	0.463	+0%
DBLP3	0.677	0.623	-8%*
WEBKB4	0.367	0.365	-0%

To evaluate the scaling effects (see Figure 7), experiments are conducted both with thresholding (0.1, 0.2, 0.3, 0.4, 0.5, and 0.8) and without thresholding (0). As shown in table 11, scaling without thresholding has significant negative impacts on DBLP3 and neutral effects on the other three datasets. The result of scaling with thresholding is similar with using thresholding only (the

performance curves of four datasets are almost the same as those of thresholding experiments (Figure 6)). These results show that scaling is not a good strategy for improving link-based clustering. Moreover, thresholding and scaling tend to exaggerate the impact of the text similarity between documents which inevitably hurts the influence of link patterns. However, thresholding and scaling could be helpful for a not- well-structured "graph". For example, using co-actor as a link between two movies gives very little indication of the connection between two movies. In fact, this is not an issue of how links affect document clustering, but an issue of how to construct a graph.

5. CONCLUSION AND FUTURE WORK

In this paper, we adopt a RL-based algorithm to examine the impacts of different linkage types on link-based document clustering. We conduct extensive comparative studies in link-based clustering using explicit, implicit and similarity links on eight different datasets. In detail, we have the following interesting findings.

First, using explicit or implicit link information, link-based k-means exhibits significant improvement over spherical k-means and model-based k-means; by the NMI measure, implicit (co-authorship) link-based k-means achieves the best performance with a 54% increase; the performance of those using citation links stands in the middle with a 17% increase; the worst are those using hyperlinks with an 11% increase. These findings are consistent with those from the comparison between pure link-based k-means and spherical k-means: the performance of the k-means using pure hyperlink, citation link and co-authorship link is inferior, similar and superior to that of pure content-based clustering, respectively. We also find that link-based k-means using content similarity links performs slightly better, but not significantly better than spherical k-means. These results indicate that the positive impacts of links on clustering are affected by the degree of complication of the link patterns. Moreover, it infers that explicit and implicit links are more helpful for clustering documents than similarity links because they encode human knowledge. Another important finding is that using uniform priors is better than using empirical priors in improving clustering performance. Furthermore, we discover that: (1) for citation link and hyperlink structures, immediate out-neighbors of a document are more important than its immediate in- neighbors in improving clustering performance; (2) expanding in-neighbors of a document, especially the out-neighbors of its immediate in- neighbors, cause the worse result. Last, thresholding and scaling have very neutral or negative impacts on improving clustering performance.

For future work, we plan to study the use of link-based k-means on more hyperlink datasets and on other types of implicit links such as co-citation, co-conference and so on.

6. ACKNOWLEDGMENTS

7. REFERENCES

[1] Angelova, R. and Weikum, G. Graph-based text classification: learn from your neighbors. *SIGIR'06*.

[2] Angelova, R. and Siersdorfer, S. A neighborhood-based approach for clustering of linked document collecitons, *CIKM'06*.

[3] Chakrabarti,S., Dom, B. E. , and Indyk, P. Enhanced hypertext categorization using hyperlinks. In *SIGMOD'98*, 307–318

[4] Cohn, D. and Hofmann,T. The missing link - a probabilistic model of document content and hypertext connectivity. In *NIPS* 13, 2001.

[5] Eppstein, D. Finding the k shortest paths. In *IEEE Symp. On Foundations of Computer Science*, 154–165, 1994.

[6] Erkan, G., Radev, D.R.: LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. *J. Artif. Intell. Res. (JAIR)* 22: 457-479 (2004)

[7] Ghani, R., Slattery, S. and Yang, Y. Hypertext Categorization using Hyperlink Patterns and Meta Data, *ICML'01*.

[8] Kleinberg. J. Authoritative sources in a hyperlinked environment. In *Proc. Ninth Ann. ACM-SIAM Symp on Discrete Algorithms*, 1998.

[9] Halkidi, Ml, Nguyen, B., Varlamis, I., and Vazirgiannis M. THESUS: Organizing Web Document Collections based on Link Semantics. *The VLDB Journal* (2003) 12: 320-322.

[10] He, X., Zha, H, Ding, C. and Simon, H. Web document clustering using hyperlink structures, Tech. Rep. CSE-01-006, Dept. of CS and Eng., Pennsylvania State University, 2001.

[11] Lafferty, J., McCallum, A. and Pereira, F. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. of ICML*, 282–289, 2001.

[12] Pelkowitz, L. A continuous relaxation labeling algorithm for markov random fields. *IEEE transactions on Systems, Man and Cybernetics*, Vol 20 No.3:709-715, 1990.

[13] Lu,Q. and Getoor, L. Link-based classification. *ICML*, 2003.

[14] McCallum, A., Nigam, K., Rennie, J. and Seymore, K. A machine learning approach to building domain-specific search engines, *IJCAI1999*.

[15] Menczer, F. Lexical and Semantic Clustering by Web links. *JASIST*, 55(14): 1261-1269, 2004.

[16] Modha, D. S. and Spangler, W. S. 2000. Clustering hypertext with applications to web searching. *HYPERTEXT '00*.

[17] Oh, H.-J., Myaeng, S. H. and Lee, M.-H. A practical hypertext categorization method using links and incrementally available class information. *SIGIR*, 264–271, 2000.

[18] Page, L., Brin,S., Motwani, R., and Winograd,T. The PageRank citation ranking: Bringing order to the Web. Technical report, 1998.

[19] Slattery, S. and Mitchell, T. Discovering text set regularities in relational domains, *ICML'00*.

[20] Strehl, A., Ghosh, J. andMooney, R. J. Impact of similarity measures on web-page clustering. In *AAAI Workshop*, 2000.

[21] Wang, Y. and Kitsuregawa, M. 2002. Evaluating contents-link coupled web page clustering for web search results. *CIKM '02*.

[22] Weiss, R., Velez, B., Sheldon, M. A. et al. HyPursuit: a hierarchical network search engine that exploits content-link hypertext clustering. *HYPERTEXT'96*.

[23] Zhang X., Zhou X., and Hu X., Semantic Smoothing for Model-based Document Clustering, *ICDM'06*, 1193-1198.

[24] Zhao, Y. and Karypis, G. Criterion functions for document clustering: experiments and analysis, Technical Report, Department of Computer Science, Univ. of Minnesota, 2001

[25] Zhou X., Zhang X. and Hu X., Semantic Smoothing of Document Models for Agglomerative Clustering, *IJCAI 2007*, 2922-2927.

[26] Zhou, X., Zhang, X., and Hu, X., Dragon Toolkit: Incorporating Auto-learned Semantic Knowledge into Large-Scale Text Retrieval and Mining, *ICTAI'07*, 197-20