
Topic Models for Hypertext: How Many Words is a Single Link Worth ?

Amit Gruber

School of Computer Science and Engineering
The Hebrew University of Jerusalem
Jerusalem 91904 Israel
amitg@cs.huji.ac.il

Michal Rosen-Zvi

IBM Research Laboratory in Haifa
Haifa University, Mount Carmel
Haifa 31905 Israel
rosen@il.ibm.com

Yair Weiss

School of Computer Science and Engineering
The Hebrew University of Jerusalem
Jerusalem 91904 Israel
yweiss@cs.huji.ac.il

Abstract

Latent topic models have been successfully applied as an unsupervised learning technique on various types of data such as text documents, images and biological data. In recent years, with the rapid growth of the Internet, these models have also been adapted to hypertext data. Explicitly modeling the generation of both words and links has been shown to improve inferred topics and open a new range of applications for topic models. However, it remains unclear how to balance the information contributed by links with the information contributed by words.

In this paper we enrich the Latent Topic Hypertext Model [7] with a parameter that stands for importance of links in the model. Specifically, we quantitatively explore whether a single link is more indicative of the topic mixture in the document it points to than a single word. We show that putting an emphasis on the topics associated with links leads to better link prediction results.

1 Introduction

The World Wide Web provides instant access to an enormous repository of knowledge in many fields and to a wide range of applications. However, due to the vast size and dynamic character of the Internet, efficient use of it necessitates intelligent tools. The Internet has been in the focus of many machine learning studies, and one of the machine learning approaches to analyzing web data is the approach of topic models. Originally, Topic models such as LSA [4], PLSA [8] or LDA [1] were suggested in the context of text analysis. With the growing interest in the analysis of Internet data, these models have been extended in various ways ([3, 6, 5, 10, 7]) to model the generation of links in addition to words. In these models, each document is viewed as a mixture of latent topics or factors, and the factors, shared by the whole corpus, are related to the words and links appearing in the documents.

These studies have shown that incorporating link information into topic models indeed gives an advantage comparing to topic models that model only the words. However, most of these works do not consider the question of how much information regarding topics is carried by a single link, compared to the information carried by a single word. Should words and links be given same or different weight in semantic analysis ?

Links and connectivity of the Internet have been proven to play an important role in determining the importance and relevance of a document for information retrieval or the interest of a certain user in it [11, 2, 9]. Connectivity is also known to indicate importance and influence in other domains, such as citations of scientific papers. On the other hand, in both domains, Internet and scientific publications, the frequency of links is much smaller than the frequency of words. This may imply that a single link might carry more information than a single word.

Cohn and Hofmann suggested the PHITS model [3] which includes a weighting term, α , between links and words. The PHITS model is a bag of words model in which for each word first a topic is drawn from the topic mixture, then the word is drawn from a topic dependent distribution. Links are modeled similarly to words: for each link, first a topic is drawn from the topic mixture, then the target document of the link is drawn from a distribution specific to the topic that has been chosen. Links are not associated with words (or with their latent topics). The input to the algorithm is two matrices: a document-term co-occurrence matrix and a document-link co-occurrence matrix. The likelihood function of the model is a sum of two terms. The first term, with weight α , stands for the likelihood of words and the second term, with weight $1 - \alpha$, stands for the likelihood of links. The co-occurrence matrices are normalized to form frequency matrices and the weight α is the tradeoff between the information in all words versus the information in all links. Cohn and Hofmann test different values of α in a classification task using two datasets and find the optimal value, considering both datasets. The optimal value of α is found to be in the interval between 0.4 and 0.6 (see figure 1 in [3]) with a wide range of values giving similar performance in this task.

In this paper we ask how much information is carried by a single link and make use of a different approach to address this question. We employ the Latent Topic Hypertext Model (LTHM) [7] that differs from PHITS in several aspects. In LTHM the hidden topic on which the link depends is the same one used to generate the anchor word of that link. A link depends on the topic mixtures of both documents at both its ends, not only on the hidden topic in the source document. the non-existence of a link between two documents is an observation that serves as evidence for the difference between the topic mixtures of the documents.

We extend LTHM to include a new parameter, R , that increases the dependency between the hidden topic of the anchored word and the topics mixture of the document it points to and. Thus the dependency of topics mixture of a document on the links pointing to it is increased. We call this model Link Amplified LTHM (LALTHM). In the original LTHM the probability for a link depended on a match between a single topic was drawn from the target document mixture of topics and the topic of the anchor word. In the LALTHM the probability depends on a match between R subsequent topic draws from the target document mixture of topics and the topic of the anchor word. The outcome is that now instead of having topics affect the likelihood through links like a single word, they are now amplified R times. The question we ask is what is the optimal value for R , or in other words, how many words is a single link worth ?

2 Generative Model and Algorithm

Here we begin with a brief review of LTHM and then we extensively describe the link amplified extension.

2.1 The Latent Topic Hypertext Model

The generative process of LTHM consists of two stages: In the first stage, the document content (the words) is created using Latent Dirichlet Allocation (LDA). After the text of all the documents in the corpus has been created, The second stage of creating links takes place. This two stages process is a consistent generative model that allows for arbitrary topology of the links graph (including directed cycles and self loops).

2.1.1 Document generation using LDA

According to the LDA model, a collection of documents is generated from a set of K latent factors or topics. One of the main assumptions in the model is that for each topic z there is a single multinomial random variable β_z that defines the probability for a word given a topic for all documents in the

link at the second stage to be $\theta_d(z)$, and the overall probability to create a link from the i 'th word in document d' to document d given a topic z is

$$\Pr(\text{link}(d', i) \rightarrow d | z_{d', i}^W, \lambda_d, \theta_d) = \lambda_d \theta_d(z) \quad (1)$$

The second step can be described equivalently as drawing a topic $z^L \sim \theta_d$ and creating a link if $z^W = z^L$.

A slight simplification of this two stages process is illustrated in Figure 2.1.1a: for illustration purpose, it shows a scenario where links can be created from a certain document d' to any other document d in the corpus.

Formally, the generative process is:

1. Choose $D + 1$ dimensional $\lambda \sim \text{Dirichlet}(\gamma)$
2. For each document $d = 1, \dots, D$
 - For each word w_i , indexed by $i = 1, \dots, N_d$
 - Choose $\tau_i \in \{1 \dots D, \emptyset\} \sim \text{Multinomial}(\lambda)$
 - If $\tau_i \neq \emptyset$ choose a topic $z_L \sim \text{Multinomial}(\theta_{\tau_i})$
 - If $z^L = z_i^W$ create a link $L_i = \tau_i$ from word i to document τ_i

2.2 Link Amplified Latent Topic Hypertext Model

2.2.1 Parameterized Family of Models

The LALTHM extends the second phase of LTHM's generative process. In particular, the step at which a decision to make a link is carried out by using the z^W and the topic mixture of the target document θ_d is modified as follows: Instead of drawing a single topic $z^L \sim \theta_d$ (equation 2.1.2), we draw a series of R topics, z^{L_1}, \dots, z^{L_R} and create a link only if $z^W = z^{L_r}$ for all $1 \leq r \leq R$.

$$\Pr(\text{link}(d', i) \rightarrow d | z_{d', i} = z, \lambda_d, \theta_{d'}^L, \theta_d) = \lambda_d \theta_{d'}(z) \theta_d(z)^R \quad (2)$$

Equivalently, the probability of a link given the parameters is:

$$\Pr(\text{link}(d', i) \rightarrow d | \lambda_d, \theta_{d'}, \theta_d) = \lambda_d \sum_z \theta_{d'}(z) \theta_d(z)^R \quad (3)$$

Figure 2.1.1b illustrates the graphical model of LALTHM in the same scenario as figure 2.1.1a. It differs from figure 2.1.1a in that it contains an additional plate around z^L .

In the LTHM, each existing link with topic z pointing to document d contributed $\theta_d(z)$ to the likelihood, the same as a single word in d would do (if $\tau = d$, a non existing link contributes $1 - \theta_d(z)$, like a word with topic different from z). In LALTHM, the contribution of a single link to the likelihood is $\theta_d(z)^R$, the same as the contribution of R words, all with *the same topic* (if $\tau = d$, a non existing link contributes $1 - \theta_d(z)^R$, like R words with at least one of these word having a topic other than z). This topic is also shared with the anchor word in the source document. Notice that non-existing links will now be less indicative of the topical difference between target and source documents. The repeated topic drawing of z^{L_1}, \dots, z^{L_R} is not equivalent to simply replicating a link R times. In the latter case, each of the R copies could have a different topic.

As an alternative model, one could think of raising λ_d to the power of R to give more emphasis to the document popularity (or in-degree) in the link generation process. However, raising all λ_d , $d = 1, \dots, D$ to the power of R is the same as choosing a different parameter λ' with $\lambda'_d = \lambda_d^R$ and $\lambda'_\emptyset = 1 - \sum_d \lambda'_d$.

2.2.2 Approximate Inference Algorithm

Exact inference in the LALTHM is intractable as an immediate consequence of the intractability of the inference in LTHM. However, modifying the LTHM EM algorithm for approximate inference in LALTHM is straightforward.

The complete EM algorithm for LTHM is provided in [7] and the code can be downloaded from <http://www.cs.huji.ac.il/~amitg/lthm.html>. Here we describe only modifications made to adjust the algorithm for LALTHM.

E-step:

In the E-step, the computation of the link emission probabilities and accordingly the posterior distribution of topics need be changed. The link emission probabilities in LALTHM are:

$$\Pr(\text{link}(d', i) \rightarrow d | z_{d',i}^W = z) = \lambda_d \theta_d(z)^R \quad (4)$$

$$\Pr(\text{no-link}(d', i) | z_{d',i}^W = z) = 1 - \sum_d \lambda_d \theta_d(z)^R \quad (5)$$

This modified probability is substituted in equation (7) in [7] to compute the posterior of z^W .

The definition of the aggregations $U_{d,z}, V_{d,z}$ remains the same as in LTHM, but their computation changes:

$V_{d,z}$ stands for the number of occurrences of a topic z associated with any incoming link of document d , hence it should just be multiplied by R compared to LTHM.

$$\begin{aligned} E(V_{d,z}) &= \sum_{(d',i) \in A_d} R \Pr(z_{d',i}^{L,1} = \dots = z_{d',i}^{L,R} = z | O, P) \\ &= R \sum_{(d',i) \in A_d} \Pr(z_{d',i}^W = z | O, P) \end{aligned} \quad (6)$$

$U_{d,z}$ stands for the number of times $\tau_{d',i} = d$ but a link has not been created. This can happen for two reasons. The first is that not all R topics $z^{L,1}, \dots, z^{L,R}$ are the same. The second reason is that all of these R topics are the same, but they are different from $z_{d',i}^W$, the topic of the anchor word (the i th word in document d'). The probability $\Pr(z_{d',i}^W = z | O, P)$ is computed using Bayes law:

$$\Pr(z_{d',i}^W = z | O, P) \propto \theta_{d'}(z) \phi_z(w) \Pr(\text{link}(d', i) | z) \quad (7)$$

where $\Pr(\text{link}(d', i) | z)$ is given in equations (4)-(5).

$$\begin{aligned} E(U_{d,z}) &= N_{NL} \lambda_d \frac{\sum_{r=1}^{R-1} r \binom{R}{r} \theta_d(z)^R (1 - \theta_d(z))^{(R-r)}}{\Pr(\text{no-link}(d', i) | \hat{O}, P)} \\ &+ R \lambda_d(z) \theta_d(z)^R \left[\sum_{(d',i) \in B} \frac{(1 - \Pr(z_{d',i}^W = z | \hat{O}, P))}{\Pr(\text{no-link}(d', i) | \hat{O}, P)} \right] \end{aligned} \quad (8)$$

$$\Pr(\text{no-link}(d', i) | \hat{O}, P) = 1 - \sum_z [\sum_d \lambda_d \theta_d(z)] [\Pr(z_{d',i}^W = z | \hat{O}, P)] \quad (9)$$

where $B = \{(d', i) : \text{no-link}(d', i)\}$ is the set of all words without link in the corpus and N_{NL} is the number of words with no associated links in the corpus ($N_{NL} = \sum_B 1$). The probability $\Pr(z_{d',i}^W = z | \hat{O}, P)$ is the probability of the topic $z_{d',i}$ conditioned on the word $w_{d',i}$, but without having seen if there's an associated link or not:

$$\Pr(z_{d',i}^W = z | O, P) \propto \theta_{d'}(z) \phi_z(w) \quad (10)$$

In the M-step, the values of $E(V_{d,z})$ and $E(U_{d,z})$ computed using these formulas are plugged into the update rule for θ_d of LTHM.

The definition of T_d required for estimating λ in the M-step needs to be slightly modified. We define T_d to be the number of times that $\tau_{d',i} = d$ for any d' and any word i in it. For $R > 0$, $E(T_d) = \sum_z (E(V_{d,z}) + E(U_{d,z}))/R$. If $R = 0$ (i.e. topics do not affect link generation) then T_d is the number of links pointing to document d and the update rule depends only on the document in degree and the Dirichlet hyper parameter γ .

3 Experiments

We compare different values of R in the task of link prediction. For this comparison, we use two datasets of web pages ³: the webkb dataset (8282 documents with 12911 links) and a small dataset of Wikipedia web pages (105 documents with 790 links).

We split each data set into a training set consisting of 90% of the documents and a test set consisting of the remaining 10%. During training, the text of all the documents is provided to all the algorithms, but only the links originating from the documents in the training set are visible. Both dataset are learned with 20 hidden aspects. During test, for each test document d we sort all documents d' in the corpus according to the probability of having a link from d (outgoing from any word in it) to d' .

The values of R we compare are 0, 1, 2, 3, 4, 5 and 10. Note that when $R = 0$ then link creation depends only on the parameter λ and does not depend on the topic at all. Thus, λ_d is estimated during train to be the in-degree of the document d (up to the Dirichlet hyper parameter γ_d), and prediction is based only on the frequency of links targeting the document d in the training set. The case $R = 1$, degenerates to the simple LTHM. The higher R is, the more contribution existing links have when estimating topics during training. Similarly, the higher the R is, the higher the influence of topics when predicting links.

Figure 2 shows results for the wikipedia dataset and figure 3 shows results for the webkb dataset. Several measures of the performance of the different values of R are shown. The results shown in the graphs are the average of 10 train/test random partitions (10-fold cross validation). The first measure is the percentage of documents in the test set for which at least one link prediction among the top N is a true link. The motivation for this measure is the following question: Suppose we want to suggest to an author of a web page other documents to link to. If we show this author N suggestions for links, will s/he use at least one of them? The other measures we use are precision (Among the top N predictions, what is the percentage of true links?), recall (What percentage of the true links are included in the top N predictions?), the F-measure ($F = 2 \frac{P \cdot R}{P+R}$ where P is the precision and R is the recall) and precision vs. recall.

From the graphs, it is evident that in both datasets the optimal value of R for the different tasks is greater than 1. In the wikipedia dataset, optimal results are obtained for $R = 3$. In the webkb dataset the optimal value is $R = 2$. We see that $R = 0$ (not using topic information at all) falls way below all other values of R . Differences in the precision are more noticeable for small values of N because of the small average number of links in a document (less than 8 in the wikipedia dataset and 1.5 in webkb). Predicting many links necessarily leads to predicting wrong links that lower the precision rate. Also, differences in the recall rate are less noticeable for large N in the wikipedia dataset, as recall rates are high for all values of R . In the webkb dataset where recall rates are not as high, differences are more noticeable for $N = 30$.

Note that in [7] LTHM (i.e. LALTHM with $R = 1$) was shown to outperform the algorithms of Cohn and Hofmann [3] and Erosheva et al. [6] in the same tasks and on the same datasets we use here.

4 Discussion

In this paper we introduced an extension to the Latent Topic Hypertext Model that amplifies the influence of links in the topical analysis. We performed an empirical study to quantify how informative are links in topical analysis of documents compared with words and tested it on two datasets. Amplifying the link-topic dependency improves hits and recall rates in the task of link prediction, whereas ignoring topics performs much worse. The specific choice we made of how to increase link

³Both datasets are available online at: <http://www.cs.huji.ac.il/~amitg/lthm.html>

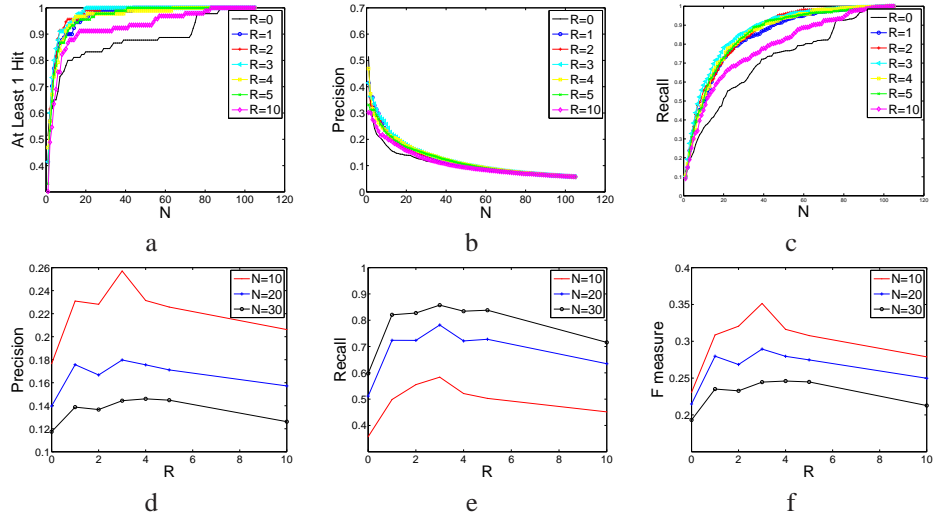


Figure 2: Comparison of link prediction in the Wikipedia test set between LTHM with different values of R . $R = 0$ is equivalent to the in-degree and $R = 1$ is the original LTHM. **a.** The percentage of text documents for which there is at least one true link among the first N predicted links. **b.** Average precision for the three methods. **c.** Average recall. **d.** Average precision as a function of R for selected values of N predicted links. **e.** Average recall as a function of R for selected values of N . **f.** Average F measure as a function of R for selected values of N .

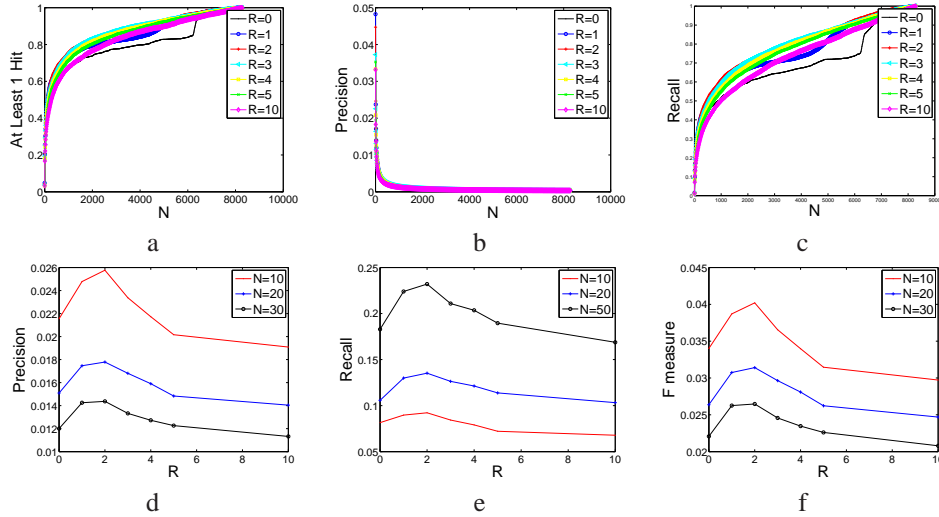


Figure 3: Comparison of link prediction in the Webkb test set between LTHM with different values of R . $R = 0$ is equivalent to the in-degree and $R = 1$ is the original LTHM. **a.** The percentage of text documents for which there is at least one true link among the first N predicted links. **b.** Average precision for the three methods. **c.** Average recall. **d.** Average precision as a function of R for selected values of N predicted links. **e.** Average recall as a function of R for selected values of N . **f.** Average F measure as a function of R for selected values of N .

influence of topic estimation affects mostly the topic mixture of the document that is being linked to. It would be interesting to compare with models that increase link influence also on the topic mixture of the source document.

Acknowledgements

Support from the Israeli Science Foundation is gratefully acknowledged.

References

- [1] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.
- [2] David Cohn and Huan Chang. Learning to probabilistically identify authoritative documents. In *Proc. 17th International Conf. on Machine Learning*, pages 167–174. Morgan Kaufmann, San Francisco, CA, 2000.
- [3] David Cohn and Thomas Hofmann. The missing link—a probabilistic model of document content and hypertext connectivity. In Todd K. Leen, Thomas G. Dietterich, and Volker Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 430–436, Cambridge, MA, 2001. MIT Press.
- [4] Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.
- [5] Laura Dietz, Steffen Bickel, and Tobias Scheffer. Unsupervised prediction of citation influences. In *International Conference on Machine Learning (ICML)*, 2007.
- [6] Elena Erosheva, Stephen Fienberg, and John Lafferty. Mixed-membership models of scientific publications. *Proceedings of the National Academy of Sciences*, 101:5220–5227, 2004.
- [7] Amit Gruber, Michal Rosen-Zvi, and Yair Weiss. Latent topic models for hypertext. In *Uncertainty in Artificial Intelligence (UAI)*, pages 230–240, 2008.
- [8] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual ACM Conference on Research and Development in Information Retrieval*, pages 50–57, New York, NY, 1999. ACM Press.
- [9] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.
- [10] Ramesh Nallapati and William Cohen. Link-plsa-lda: A new unsupervised model for topics and influence of blogs. In *International Conference for Weblogs and Social Media*, 2008.
- [11] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.