

Learning Probabilistic Models of the Web

Thomas Hofmann

Department of Computer Science, Box 1910

Brown University

Providence, RI 02912

th@cs.brown.edu, www.cs.brown.edu/~th

Abstract

In the World Wide Web, myriads of hyperlinks connect documents and pages to create an unprecedented, highly complex graph structure - the Web graph. This paper presents a novel approach to learning probabilistic models of the Web, which can be used to make reliable predictions about connectivity and information content of Web documents. The proposed method is a probabilistic dimension reduction technique which recasts and unites Latent Semantic Analysis and Kleinberg's Hubs-and-Authorities algorithm in a statistical setting.

This is meant to be a first step towards the development of a statistical foundation for Web-related information technologies. Although this paper does not focus on a particular application, a variety of algorithms operating in the Web/Internet environment can take advantage of the presented techniques, including search engines, Web crawlers, and information agent systems.

1 Introduction: Predictive Models of the Web

We use the phrase “predictive model” or “generative model” to refer to a parameterized statistical family that describes a (fictive) stochastic generation process which is assumed to underly the observed data. In the case of the Web, a generative model has to capture the textual content of documents as well as the inter-document connectivity by hyperlinks. Other document attributes like markup tags, multimedia content and anchor information can also be included, but we will focus on terms and links in the rest of the paper.

Abstracting away from the sequential order of term and link occurrences, each document is reduced to a vector of term counts and a vector of (outgoing) link counts, i.e., each document is characterized by its first order statistics of occurring terms and links.

In order to be more formal, we introduce the following notation: we assume that a document collection Δ (e.g., a subset of Web pages) and a vocabulary of terms Ω (such as words and phrases) is given. A generic document (page) is denoted by $\delta \in \Delta$ and a generic term by $\omega \in \Omega$. We model two types of random events: the occurrence of a term ω in a document δ and the existence of a link from document δ to document δ' . Thus, we have to estimate probabilities $P_\delta(\omega)$ that an arbitrary token in δ is of type ω as well as probabilities $P_\delta(\delta')$ that a link in δ will point to δ' . Since

the data has been reduced to term/link occurrence vectors, these estimates should be based on the counts $N_\delta(\omega)$ and $N_\delta(\delta')$ for the number of times a term ω and link to δ' , respectively, occurred in document δ .

A canonical way to estimate these probabilities is by maximum likelihood estimation. Denoting by $N_\delta(\Omega)$ the total number of word occurrences in δ and by $N_\delta(\Delta)$ the total number of links, the maximum likelihood estimator is simply given by the relative frequencies

$$\hat{P}_\delta(\omega) = N_\delta(\omega)/N_\delta(\Omega), \quad \hat{P}_\delta(\delta') = N_\delta(\delta')/N_\delta(\Delta). \quad (1)$$

Yet, what makes this estimation problem difficult is the size of the state space over which probabilities have to be estimated compared to the small number of observations that are typically available. As a consequence, the maximum likelihood estimator will be plagued by high variance. In particular, it fails to assign meaningful probabilities to events that might have not been observed at all in the (training) data.

To further illustrate this point we imagine a simple “game” where a single term ω or a link occurrence δ' in $\delta \in \Delta$ is hidden. The goal is to predict the identity of this term or link, based on all the other information that is available in Δ . The maximum likelihood estimator only considers the observed occurrences in δ , ignoring all other documents in $\Delta - \{\delta\}$. However, since documents are related to one another, it is possible to learn about a document δ from other documents in the collection. While this is only a conjecture in the general case, playing the above game, we can actually verify whether or not we did learn useful information from the document collection by evaluating the predictive performance of the model, for example, by using measures like the average log-probability or, equivalently, the perplexity on independent (hidden) test data.

2 Probabilistic Latent Semantic Analysis

In previous work, we have developed a technique called *Probabilistic Latent Semantic Analysis* (PLSA) [6] to address the above estimation problem for document collections without hyperlinks. PLSA has been inspired by Latent Semantic Analysis (LSA) [3], where the main idea is to reduce the dimensionality of documents represented in the vector space model. In the probabilistic setting of PLSA, the goal is to compute simultaneous estimates for the probability mass functions P_δ over Ω for all $\delta \in \Delta$. Formally, the PLSA model assumes that all P_δ can be represented in the following functional form

$$P_\delta(\omega) = \sum_{k=1}^K \tau_\delta^k \Phi_k(\omega), \quad \text{with } \tau_\delta^k \geq 0, \quad \sum_{k=1}^K \tau_\delta^k = 1. \quad (2)$$

Here Φ_k are “prototypical” probability mass functions and τ_δ are document-specific vectors of mixing proportions or weights. The PLSA model effectively restricts all P_δ to be in the $(K - 1)$ -dimensional convex hull of the Φ_k ’s. It has been shown that the latter typically capture “topics”, i.e., each Φ_k will assign high probabilities to terms that are likely to occur in the context of a particular topic. One can use the well-known Expectation Maximization (EM) algorithm [4, 9, 6] to fit this model from data (cf. Section 4).

PLSA has successfully been applied in *ad hoc* retrieval [6], where it is closely related to other recent approaches for retrieval based on document-specific language models [8, 1]. The common idea of these approaches is that a document-specific (unigram) language-model $P_\delta(\omega)$ can be used to compute for each document the probability to generate a given query. Documents that make a query “more likely” are then assumed to be also of higher relevance to the query, which can be justified formally using Bayes’ rule.

3 Joint Models for Content and Connectivity

In this paper, we aim at an extension of the PLSA model to include the additional hyperlink structure between documents. A natural extension of the model is to predict the existence of a link $\delta \rightarrow \delta'$, by associating with each “topic” not only a distribution Φ_k over terms, but also a distribution Ψ_k over Δ , i.e., over possible links to Web pages. In this case one gets in addition to (2),

$$P_\delta(\delta') = \sum_{k=1}^K \tau_\delta^k \Psi_k(\delta'). \quad (3)$$

In this model, each “topic” has some probability $\Psi_k(\delta')$ to link to a document δ' . The overall probability for a document δ to link to δ' is then again a convex combination, using the same weights τ_δ as before. The advantage of this joint modeling approach is that it integrates content- and link-information in a principled manner. Since the parameters τ_δ are “shared” in (2) and (3), the learned decomposition has to be satisfying with respect to both, the predictive performance on terms and the prediction of links between documents. Which importance one gives to predicting terms relative to predicting links may depend on the specific application. In general, we propose to maximize the following (normalized) likelihood function with a relative weight α ,

$$l(\Phi, \Psi, \tau) = \sum_{\delta \in \Delta} \left[\alpha \sum_{\omega \in \Omega} \frac{N_\delta(\omega)}{N_\delta(\Omega)} \log \sum_{k=1}^K \tau_\delta^k \Phi_k(\omega) + (1-\alpha) \sum_{\delta' \in \Delta} \frac{N_\delta(\delta')}{N_\delta(\Delta)} \log \sum_{k=1}^K \tau_\delta^k \Psi_k(\delta') \right], \quad (4)$$

with respect to all parameters Φ, Ψ, τ . The normalization ensures that each document gets the same weight irrespective of its length; $0 \leq \alpha \leq 1$ has to be specified a priori.¹

It is interesting to note that by ignoring the actual document content, i.e., the terms that occur in the document, one gets a decomposition of the Web graph adjacency matrix (including multiplicities) that is closely related to the Hubs-and-Authorities (HITS) algorithms proposed in [7]. HITS is based on an iterative re-scoring method that can be interpreted in terms of a Singular Value Decomposition (SVD).

¹One has to keep in mind that one might want to use different criteria for training and for testing on new data. For example, the only goal might be to predict the link structure between documents ($\alpha = 0$). Nevertheless, using information about the document content can be very helpful in predicting possible links and choosing $\alpha > 0$ during training might result in more accurate models.

The latter fact implies that HITS computes a low-rank approximation of the adjacency matrix that is optimal in the sense of the Frobenius norm (or L_2 matrix norm). As has been shown in [6], PLSA (and the extension presented here) can also be thought of in terms of a matrix decomposition. In contrast to SVD, the PLSA decomposition is based on a (multinomial) likelihood function and allows a rigorous probabilistic interpretation (cf. also [5]). As a limiting case of our model ($\alpha = 0$) we thus recover a probabilistic version of HITS. Of course, by setting $\alpha = 1$ one is back to the standard PLSA model in (2) which is a probabilistic version of LSA.

4 Model Fitting by EM

The Expectation Maximization (EM) algorithm [4] is a standard algorithm for maximizing likelihood functions of mixture models as the one in (4). In general, the log-likelihood function in (4) will have many local maxima and – depending on the randomized starting state – EM is only guaranteed to find a local maximum which may not be a global maximum.

EM is an iterative procedure that alternates two steps until convergence: an expectation (or E) step and a maximization (or M) step. On an intuitive level, what one needs to compute in the E-step is for each term and link occurrence (δ, ω) and (δ, δ') the probability that these are “explained” by the k -th topic Φ_k and Ψ_k , respectively. Formally, one gets the following posterior probability equations from Bayes’ rule

$$P^k(\delta, \omega) = \frac{\tau_\delta^k \Phi_k(\omega)}{\sum_{l=1}^K \tau_\delta^l \Phi_l(\omega)}, \quad P^k(\delta, \delta') = \frac{\tau_\delta^k \Psi_k(\delta')}{\sum_{l=1}^K \tau_\delta^l \Psi_l(\delta')}. \quad (5)$$

Qualitatively this states that one favors the k -th topic as an explanation, if τ_δ^k is large, i.e., the document “participates” in this topic and if $\Phi_k(\omega)$ is large, i.e., the term has a high probability of occurrence in the k -th topic; a similar relation holds for links.

In the M-step, one re-estimates the parameters based on the posterior probabilities computed in the E-step. A formal derivation leads to

$$\Phi_k(\omega) \propto \sum_{\delta \in \Delta} \frac{N_\delta(\omega)}{N_\delta(\Omega)} P^k(\delta, \omega), \quad (6)$$

$$\Psi_k(\delta') \propto \sum_{\delta \in \Delta} \frac{N_\delta(\delta')}{N_\delta(\Delta)} P^k(\delta, \delta'), \quad (7)$$

$$\tau_\delta^k \propto \sum_{\omega \in \Omega} \frac{\alpha N_\delta(\omega)}{N_\delta(\Omega)} P^k(\delta, \omega) + \sum_{\delta' \in \Delta} \frac{(1-\alpha) N_\delta(\delta')}{N_\delta(\Delta)} P^k(\delta, \delta'). \quad (8)$$

Here all parameters have to be normalized such that they fulfill the appropriate normalization constraints (sum to unity). As can be seen, the weight α only affects the estimation of the “shared” τ parameters. If α is large, more weight is put on the topics which explain the terms occurring in δ well, if α is small, more weight is put on topics which explain δ ’s outlinks well.

5 Experiments

In this section, some preliminary result of an ongoing experimental evaluation are reported. These experiments provide a proof of concept, but do not claim to substitute a full-scale quantitative analysis.

We have followed the method described in [7] to generate subsets of Web documents. The generation involves three steps: (i) a query is issued to a search engine (Altavista: www.altavista.com) and the top M matching pages (e.g., $M = 100$) are retrieved. (ii) For each of these M pages

Ulysses

ulysses 0.022082	grant 0.019197	page 0.020032
space 0.015334	s 0.017092	ulysses 0.013361
page 0.013885	ulysses 0.013781	new 0.010455
home 0.011904	online 0.006809	web 0.009060
nasa 0.008915	war 0.006619	site 0.009009
science 0.007417	school 0.005966	joyce 0.008430
solar 0.007143	poetry 0.005762	net 0.007799
esa 0.006757	president 0.005259	teachers 0.007236
mission 0.006090	civil 0.005065	information 0.007170
ulysses.jpl.nasa.gov/ 0.028583	www.lib.siu.edu/projects /usgrant/ 0.019358	http://www.purchase.edu /Joyce/Ulysses.htm 0.008469
helio.estec.esa.nl/ulysses 0.026384	www.whitehouse.gov /WH/glimpse /presidents /ug18.html 0.017598	http://www.bibliomania.com /Fiction/joyce/ulysses /index.html 0.007274
www.sp.ph.ic.ac.uk/Ulysses 0.026384	saints.css.edu/mkelsey /gppg.html 0.015838	http://teachers.net /chatroom/ 0.005082



Figure 1: Top probability words and links for 3 topics from a $K = 5$ topic decomposition of a set of 410 Web pages created from a search result using the query term “Ulysses”.

δ , we include all pages (up to a certain maximum number) for which we find links in δ . (iii) For each of the M pages δ , we also include all pages that point to δ (limiting again the maximum number of inlinks) using Altavista’s inverted link index. The Web pages fetched in steps (i)-(iii) are preprocessed by stripping away markup tags and removing stop words as well as any non-textual content. By matching URLs of Web pages with the URLs found on outlinks, a corresponding Web (sub-)graph is generated. Finally counts $N_\delta(\omega)$ and $N_\delta(\delta')$ are extracted.

In Figure 1 and 2, we show (a subset of) the “topics” that have been found for queries using the keyword “Ulysses” and “Jaguar”, respectively. The optimal number K has been chosen to maximize the predictive performance on hold-out data. Both queries are highly ambiguous: Just based on the query term, it is simply impossible to know what the “correct answer” would be. In both cases, the probabilistic decomposition reveals this ambiguity by identifying relevant “topics”, e.g., (i) “Ulysses”, the space probe, (ii) “Ulysses” S. Grant, a former US president, and (iii) “Ulysses” a book written by James Joyce.² The decomposition provides both, a concise description of topics by their most probable key words (terms ω for which $\Phi_k(\omega)$ is largest) as well as authoritative links (pages δ' for which $\Psi_k(\delta')$ is largest).

6 Conclusions and Future Work

We have presented a predictive model of the Web based on a probabilistic decomposition, along with a statistical model fitting procedure. The model can be directly used to derive quantitative predictions about term and link occurrences. In addition, the factors identified by the decomposition contain useful information about “topics”, i.e., a characterization by keywords and most authoritative Web pages. In contrast to other approaches to combine content and link analysis (e.g., [2]), our model combines terms and links in a principled way. As a data mining method, the probabilistic nature of our approach offers the advantage to validate the significance of the extracted “topics” in a principled manner.

We are currently working on a quantitative evaluation of our method and a systematic comparison with other methods (such as the HITS algorithm and LSA). In addition, we are working on a search engine post-processing tool that can disambiguate lists of URLs returned by standard search en-

²Unfortunately, the original (i.e., Homer’s) “Ulysses” was lost in the noise of the Web.

Jaguar

jaguar 0.022260	jaguar 0.030825	server 0.152747
atari 0.017393	parts 0.022084	url 0.109497
games 0.009168	jag 0.011407	requested 0.101342
page 0.007400	links 0.006485	file 0.083555
game 0.006924	used 0.006003	error 0.054760
homepage 0.006784	new 0.005837	htm 0.051896
links 0.006716	classic 0.005554	html 0.040400
gif 0.006412	bat 0.005510	port 0.027134
video 0.006297	club 0.005298	apache 0.027066
www.atarihq.com/ 0.052433	www.jag-lovers.org/ 0.055210	specified 0.017443
home.earthlink.net 0.022940	www.jagweb.com/ 0.054455	www.scottsdalejag.com/ 0.0
/~mfurdock/jaguar/jaguar.htm	www.xks.com/ 0.030477	www.jaguarpaw.com/ 0.0
homepage2.rconnect.com 0.019663	www.jcna.com/ 0.021773	www.bitcon.no 0.0
/forhan/jaguar.html		/~gunnar/sovereign.html
		masjaguar.com 0.0
		/jowr/jaguarring.htm



404 ERROR

Figure 2: Top probability words and links for 3 topics from a $K = 6$ topic decomposition of a set of 613 Web pages created from a search result using the query term “Jaguar”.

gines. We are also developing a tool for semi-automated document annotation by suggesting links to related Web pages.

References

- [1] A. Berger and J. Lafferty. Information retrieval as statistical translation. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1999.
- [2] K. Bharat and M. R. Henzinger. Improved algorithms for topic distillation in hyperlinked environments. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1998.
- [3] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407, 1990.
- [4] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Statist. Soc. B*, 39:1–38, 1977.
- [5] K. Hall and T. Hofmann. Learning curved multinomial subfamilies for natural language processing and information retrieval. In *Proceedings of the 17th International Conference on Machine Learning (ICML)*, 2000.
- [6] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd ACM-SIGIR International Conference on Research and Development in Information Retrieval, Berkeley, California*, pages 50–57, 1999.
- [7] J. Kleinberg. Authoritative sources in a hyperlinked environment. In *Proc. 9th ACM-SIAM Symposium on Discrete Algorithms*, 1998.
- [8] J.M. Ponte and W.B Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 275–281, 1998.
- [9] L. Saul and F. Pereira. Aggregate and mixed-order Markov models for statistical language processing. In *Proceedings of the 2nd International Conference on Empirical Methods in Natural Language Processing*, pages 81–89, 1997.