Link-based Classification

Qing Lu QINGLU@CS.UMD.EDU
Lise Getoor GETOOR@CS.UMD.EDU
Department of Computer Science/UMIACS, University of Maryland, College Park, MD 20742

Abstract

A key challenge for machine learning is tackling the problem of mining richly structured data sets, where the objects are linked in some way due to either an explicit or implicit relationship that exists between the objects. Links among the objects demonstrate certain patterns, which can be helpful for many machine learning tasks and are usually hard to capture with traditional statistical models. Recently there has been a surge of interest in this area, fueled largely by interest in web and hypertext mining, but also by interest in mining social networks, bibliographic citation data, epidemiological data and other domains best described using a linked or graph structure. In this paper we propose a framework for modeling link distributions, a link-based model that supports discriminative models describing both the link distributions and the attributes of linked objects. We use a structured logistic regression model, capturing both content and links. We systematically evaluate several variants of our link-based model on a range of data sets including both web and citation collections. In all cases, the use of the link distribution improves classification accuracy.

1. Introduction

Traditional data mining tasks such as association rule mining, market basket analysis and cluster analysis commonly attempt to find patterns in a data set characterized by a collection of independent instances of a single relation. This is consistent with the classical statistical inference problem of trying to identify a model given a random sample from a common underlying distribution.

A key challenge for machine learning is to tackle the problem of mining more richly structured data sets, for example multi-relational data sets in which there are record linkages. In this case, the instances in the data set are linked in some way, either by an explicit link, such as a URL, or a constructed link, such as join between tables stored in a database. Naively applying traditional statistical inference procedures, which assume that instances are independent, can lead to inappropriate conclusions (Jensen, 1999). Care must be taken that potential correlations due to links are handled appropriately. Clearly, this is information that should be exploited to improve the predictive accuracy of the learned models.

Link mining is a newly emerging research area that is at the intersection of the work in link analysis (Jensen & Goldberg, 1998; Feldman, 2002), hypertext and web mining (Chakrabarti, 2002), relational learning and inductive logic programming (Dzeroski & Lavrac, 2001) and graph mining (Cook & Holder, 2000). Link mining is potentially useful in a wide range of application areas including bio-informatics, bibliographic citations, financial analysis, national security, and the Internet.

In this paper we propose a statistical framework for modeling link distributions. Rather than an ad hoc collection of methods, the proposed framework extends classical statistical approaches to more complex and richly structured domains than commonly studied. It is similar in spirit to the work in conditional random fields (Lafferty et al., 2001), although here we focus on non-sequence data.

The framework we propose stems from our earlier work on link uncertainty in probabilistic relational models (Getoor et al., 2002). However in this paper, we *do not* construct explicit models for link existence. Instead we model link distributions, which describe the neighborhood of links around an object, and can capture the correlations among links. With these link distributions, we propose algorithms for link-based classification. In order to capture the joint distributions of the links, we use a logistic regression model for both the content and the links. A key challenge is structuring the model appropriately; simply throwing both links

Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003), Washington DC, 2003.

and content attributes into a 'flat' logistic regression model does not perform as well as a structured logistic regression model that combines one logistic regression model built over content with a separate logistic regression model built over links.

A second challenge is classification using a learned model. A learned link-based model specifies a joint distribution over link and content attributes and, unlike traditional statistical models, these attributes may be correlated. Intuitively, for linked objects, updating the category of one object can influence our inference about the categories of its linked neighbors. This requires a more complex classification algorithm. Iterative classification and inference algorithms have been proposed for hypertext categorization (Chakrabarti et al., 1998; Oh et al., 2000) and for relational learning (Neville & Jensen, 2000; Taskar et al., 2001; Taskar et al., 2002). Here, we also use an iterative classification algorithm. One novel aspect is that unlike approaches that make assumptions about the influence of the neighbor's categories (such as that linked objects have similar categories), we explicitly learn how the link distribution affects the category. We also examine a range of ordering strategies for the inference and evaluate their impact on overall classification accuracy.

The main contributions of our work are:

- A statistical framework is proposed for modeling link distributions, a link-based model, which integrates statistical features such as object descriptions with linkage information.
- Experimental results demonstrating that the use of link distributions clearly improves classification accuracy.
- An evaluation of an iterative categorization algorithm that makes use of a variety of inference strategies.

Related work is discussed in Section 2. Section 3 describes our link-based models. In Section 4, we describe how the parameter estimation is performed for the models, and in Section 5 how the link-based model is used for categorization. Experimental results are discussed in Section 6.

2. Related Work

There has been a growing interest in learning from structured data. By structured data, we simply mean data best described by a graph where the nodes in the graph are objects and the edges/hyper-edges in the graph are links or relations between objects. Tasks include hypertext classification, segmentation, information extraction, searching and information retrieval, discovery of authorities and link discovery. Domains include the world-wide web, bibliographic citations, criminology, bio-informatics to name just a few. Learning tasks range from predictive tasks, such as classification, to descriptive tasks, such as the discovery of frequently occurring sub-patterns.

Here, we describe some of the most closely related work to ours, however because of the surge in recent interest, and the wide range of venues where research is reported (including WWW, NIPS, ICML, SIGIR, SIGMOD, VLDB), our list is sure to be incomplete.

Probably the most famous example of exploiting link structure is the use of links to improve information retrieval results. Both the well-known page rank (Page et al., 1998) and hubs and authority scores (Kleinberg, 1999) are based on the link-structure of the web. Dean and Henzinger (1999) propose an algorithm based on co-citation to find related web pages. Kubica et al. (2002) have proposed a probabilistic model for link detection and modeling groups.

One line of work closely related to link-based classification is hypertext and web page classification. This work has its roots in the information retrieval community. A hypertext collection has a richer structure than a collection of text documents. In addition to words, hypertext has both incoming and outgoing links. Traditional bag-of-words model discard this rich structures of hypertext and do not make full use of link structure of hypertext. In the web page classification problem, the web is viewed as a large directed graph, and our objective is to label the category of a web page, based on features of the current page and features of linked neighbors. With the use of linkage information, anchor text and neighboring text around each incoming link, better categorization results can be achieved.

Chakrabarti et al. (1998) propose a probabilistic model to utilize both text and linkage information to classify a database of patents and a small web collection. They showed that naively incorporating words from neighboring pages reduces performance, while incorporating category information, such has hierarchical category prefixes, improves performance. Oh et al. (2000) reported similar results on a collection of encyclopedia articles: simply incorporating words from neighboring documents was not helpful, while making use of the predicted class of neighboring documents was helpful.

These results indicate that simply assuming that link documents are on the same topic, and incorporating the features of linked neighbors is not generally effective. Another line of works tries to construct features from related documents. The first example is the work of Slattery and Craven (1998). They propose a model which beyond using words in a hypertext docu-

ment, makes use of anchor text, neighboring text, capitalized words and alphanumeric words. Using these statistical features and a relational rule learner based on FOIL (Quinlan & Cameron-Jones, 1993), they propose a combined model for text classification. Popescul et al. (2002) also combine a relational learner with a logistic regression model to improve accuracy for document mining.

Another approach is to identify certain types of hypertext regularities such as encyclopedic regularity (linked objects typically have the same class) and co-citation regularity (linked objects may not share the same class, but objects that are cited by the same object tend to have the same class). Yang et al. (2002) gives an in-depth investigation of the validity of these regularities across several data sets and using a range of classifiers. They found that the usefulness of the regularities varied, depending on both the data set and the classifier being used.

Here, we propose a method that can learn a variety of different regularities among the categories of linked objects. However, unlike Joachims et al. (2001) in which the citation matrix is explicitly modeled, here we use a much simpler model that captures the topic distribution, rather than the identities of the particular documents.

Others have proposed generative probabilistic models for linked data. Cohn and Hofmann (2001) propose a probabilistic model for hypertext content and links. In earlier work (Getoor et al., 2002), we also proposed a generative model for relational data, both content and links. The model that we propose is not a generative model for the links; in the cases we examine, the links are always observed, so it is not required that we have a generative model for them.

In contrast to work which constructs relational features, we use a simpler model that tries to capture the link distribution. Other approaches such as relational Markov networks (Taskar et al., 2002) support arbitrary dependency. Here we assume a fixed model that combines two models built over both content and link information. Because we are combining predictions from two distinct models, there is some similarity with co-training (Blum & Mitchell, 1998), although the way in which we make use of unlabeled data is rather different.

Like Joachims et al. (2001); Taskar et al. (2002); Popescul et al. (2002), our approach is based on a logistic regression model. The Naive Bayes model has been used more extensively for text categorization, but for the data sets examined here, we found the logistic regression model consistently improved our accuracy. Ng and Jordan (2002) gives a comparison of the two models.

3. Link-based models

In our original work on link uncertainty (Getoor et al., 2002), we proposed two simple models for link uncertainty. We showed that these models improved predictive accuracy in a variety of domains, including a web page classification problem (Craven et al., 1998) and a scientific paper classification problem (McCallum et al., 2000). While the models we proposed improved performance, they were rather simplistic; similar to a Naive Bayes model, the existence of a link is independent of the other links—it does depend on attributes of related objects, however one cannot reason directly about groups of links.

Here we propose a general notion of a link-based model that supports much richer probabilistic models based on the distribution of links and based on attributes of linked objects.

3.1. Definitions

The generic link-based data we consider is essentially a directed graph, in which the nodes are objects and edges are links between objects.

- $\mathcal{O}$ - The collection of objects, $\mathcal{O} = \{X_1, \ldots, X_N\}$ where X_i is an object, or node in the graph. $\mathcal{O}$ is the set of nodes in the graph.
- $\mathcal{L}$ - The collections of links between objects. $L_{i \to j}$ is a link between object X_i and object X_j. $\mathcal{L}$ is the set of edges in the graph.
- $\mathcal{G}(\mathcal{O}, \mathcal{L})$ - The directed graph defined over $\mathcal{O}$ by $\mathcal{L}$.

Our model supports classification of objects based both on features of the object *and* on properties of its links. The object classifications are a finite set of categories $\{c_1, \ldots, c_k\}$ where $c(X)$ is the category c of object X. We will consider the neighbors of an object X_i via the following relations:

- $I(X_i)$ - the set of incoming neighbors of object X_i, $\{X_j \mid L_{j \to i} \in \mathcal{L}\}$.
- $O(X_i)$ - the set of outgoing neighbors of object X_i, $\{X_j \mid L_{i \to j} \in \mathcal{L}\}$.
- $Co(X_i)$ - The set of objects co-cited with object X_i, $\{X_j \mid X_j \neq X_i$ and there is a third object X_k that links to both X_i and $X_j\}$.

3.2. Object features

The attributes of an object provide a basic description of the object. Traditional classification algorithms are

based on object attributes. In a linked-based approach, it may also make sense to use attributes of *linked* objects. Furthermore, if the links themselves have attributes, these may also be used. [1] However, in this paper, we simply use object attributes, and we use the notation $OA(X)$ for the attributes of object X. As an example, in the scientific literature domain, the object features might consist of a variety of text information such as title, abstract, authorship and content. In the domains we examined, the objects are text documents the object features we use are word occurrences.

3.3. Link features

To capture the link patterns, we introduce the notion of link features as a way of capturing the salient characteristics of the objects' links. We examine a variety of simple mechanisms for doing this. All are based on statistics computed from the linked objects rather than the *identity* of the linked objects. Describing only the limited collection of statistics computed from the links can be significantly more compact than storing the link incidence matrix. In addition, these models can accommodate the introduction of new objects, and thus are applicable in a wider range of situations.

We use the notation $LD(X)$ for the link features of object X. We examine several ways of constructing link features. All are constructed based on statistics computed from the *categories* of the different sets of linked objects.

The simplest statistic to compute is a single feature, the mode, from each set of linked objects from the in-links, out-links and co-citation links. We call this the **mode-link** model.

We can use the frequency of the categories of the linked objects; we refer to this as the **count-link** model. In this case, while we have lost the information about the individual entity to which the object is connected, we maintain the frequencies of the different categories.

A middle ground between these two is a simple binary feature vector; for each category, if a link to an object of that category occurs at least once, the corresponding feature is 1; the feature is 0 if there are no links to this category. In this case, we use the term **binary-link** model. [2]

[1] Essentially this is a propositionalization (Flach & Lavrac, 2000; Kramer et al., 2001) of the aspects of the neighborhood of an object in the graph. This is a technique that has been proposed in the inductive logic programming community and is applicable here.

[2] The distinction between binary-link and count-link models is akin to the distinction between binary and multinomial Naive Bayes (McCallum & Nigam, 1998). However, we use a discriminative model (logistic regression) rather than a generative model (Naive Bayes).

4. Predictive model for object classification

Logistic regression (Hosmer & Lemeshow, 1989) has been used in statistical analysis for many years. Support Vector Machines (SVMs), which can be regarded as an approximation and limiting case of a generalized logistic regression family, have been successfully applied for text categorization (Joachims, 1998). Now logistic regression is gaining considerable interest in application for categorization. Here we present an approach to compute the conditional probability distribution $P(c \mid OA(X), LD(X))$ based on the logistic regression method.

A binary or dichotomous classification problem is to determine a label (dependent variable) $c \in \{-1, +1\}$ given an input vector (explanatory variable) x. Logistic regression is a discriminative method for solving this problem, which models the conditional probability $P(c = 1 \mid w, x)$ with a logit transformation:

$$g(P(c = 1 \mid w, x)) = w^T x$$

where $g(t) = \ln[\frac{t}{1-t}]$ is called a logit function.

So we have $P(c \mid w, x) = \frac{1}{exp(-w^T x c)+1}$, where $c \in \{-1, +1\}$. Given a training set of labeled data (x_i, c_i), where $i = 1, 2, \ldots, n$ and $c_i \in \{-1, +1\}$, to compute the conditional probability $P(c \mid w, x)$ is to find the optimal w for the discriminative function, which is equivalent to the following regularized logistic regression formulation (Zhang & Oles, 2001):

$$\hat{w} = \text{arginf}_w \frac{1}{n} \sum_{i=1}^{n} \ln(1 + exp(-w^T x_i c_i)) + \lambda w^2$$

where we use a zero-mean independent Gaussian prior for the parameter w: $P(w) = exp(\lambda w^2)$. A penalized likelihood method, which can be interpreted as an MAP estimator with prior $P(w)$ $exp(\lambda w^2)$, is used to optimize the parameters for our regularized logistic regression model.

For our predictive model, we compared a variety of regularized logistic regression models. The simplest is a *flat* model, which uses a single logistic regression model over both the object attributes and link statistics.

We also explore the use of a structured logistic regression model. Regularized logistic regression is used to compute the posterior probability $P(c \mid OA(X))$ and $P(c \mid LD(X))$ separately. We tuned the regularization parameter λ separately on a held-out validation set.

$$P(c \mid w_o, OA(X)) = \frac{1}{exp(-w_o^T OA(X) c) + 1}$$

$$P(c \mid w_l, LD(X)) = \frac{1}{exp(-w_l^T LD(X) c) + 1}$$

where w_o and w_l are the parameters for $P(c \mid OA(X))$ and $P(c \mid LD(X))$ respectively.

Now the MAP estimation for categorization becomes

$$\hat{C}(X) \quad = \quad \text{argmax}_{c \in C} P(c \mid OA(X)) P(c \mid LD(X))$$

For multi-class categorization, we trained one-against-others logistic regression model for each category and during testing, picked the category with the highest posterior probability.

5. Link-based classification

Our proposed model is learned from a fully labeled training data set. In order to make use of it for prediction, we are in a more challenging situation than classical prediction. Here, we have an unlabeled graph over a collection of objects. The object attributes and the links are observed, only the categories are unobserved. We need to be able to predict the categories of all of the objects at once—clearly each of these predictions depends on neighboring predictions. In our model, to predict the category for one object, we need the categories of object's neighbors, which will also be unlabeled.

In the spirit of Chakrabarti et al. (1998) and Neville and Jensen (2000), we propose an iterative classification algorithm (ICA). The general approach has been studied in numerous fields, including relaxation-labeling in computer vision (Hummel & Zucker, 1983), inference in Markov random fields (Chellappa & Jain, 1993) and loopy belief propagation in Bayesian networks (Murphy & Weiss, 1999). Here, the challenging aspect is the non-regular structure of the inference, and the novelty of our approach is our attempt to make use of this in the inference.

Our ICA has two stages: bootstrap and iteration. At the beginning of the prediction phase, all of the objects are unlabeled, and thus the link statistics are unknown. A bootstrap stage is used to assign an initial category to each object, based solely on the object attributes (which are observed). This simplified model is used to give an initial classification for each object. During the iteration phase, the full version of our model is used to update the classifications of the objects. The algorithm terminates when it converges (there are no longer any updates to the categories) or a maximum number of steps has been reached.

Step 1: (Bootstrap) Using only the object attributes, assign an initial category for each object in the test set.

Step 2: (Iteration) Iteratively apply the full model to classify each object until the termination criterion have been satisfied. For each object,

1. Compute the link statistics, based on the current assignments to linked objects
2. Compute the posterior probability for the category of this object.
3. The category with the largest posterior probability is chosen as a new category for current object.

In the iterative step there are many possible orderings for objects. One approach is based simply on the number of links; Oh et al. (2000) report no significant improvement using this method. Neville and Jensen (2000) propose an iterative classification algorithm where the ordering is based on the inference posterior probability of the categories. They report an improvement in classification accuracy. We explore several alternate orderings based on the estimated link statistics. We propose a range of link-based adaptive strategies which we call *Link Diversity*. Link diversity measures the number of different categories to which an object is linked. The idea is that, in some domains at least, we may be more confident of categorizations of objects with low link-diversity, in essence, the object's neighbors are all in agreement. So we may wish to make these assignments first, and then move on to the rest of the pages. In our experiments, we evaluate the effectiveness of different ordering schemes based on link diversity.

6. Experimental Results

We evaluated our link-based classification algorithm on two standard data sets Cora (McCallum et al., 2000) and WebKB (Craven et al., 1998) and a data set that we constructed from citeseer entries (Giles et al., 1998), which we call CiteSeer. In all three domains, document frequency (DF) is used to prune the word dictionary. Words with DF values less than 10 are discarded.

The Cora data set contains 4187 machine learning papers, each of which is categorized into one of seven possible topics. We consider only the 3181 papers which are cited by or cite other papers. There are 6185 citations in the data set. After stemming and removing stop words and rare words, the dictionary contains 1400 words. We split the data set into three separate equally sized parts.

The CiteSeer data set has approximately 3600 papers from six categories: Agents, Artificial Intelligence, Database, Human Computer Interaction, Machine Learning and Information Retrieval. There are 7522 citations in the data set. After stemming and removing stop words and rare words, the dictionary for CiteSeer contains 3000 words. Similar to the Cora data set, we split the data set into three splits with roughly equal size.

Table 1. Summary of average accuracy, precision, recall and F1 measure using different link-based models on Cora, CiteSeer and WebKB. The random iteration ordering strategy is used.

Cora							
	Content-Only	Flat-Mode	Flat-Binary	Flat-Count	Mode-Link	Binary-Link	Count-Link
Avg. Accuracy	0.674	0.649	0.74	0.728	0.717	0.754	**0.758**
Avg. Precision	0.662	0.704	0.755	0.73	0.717	0.747	**0.759**
Avg. Recall	0.626	0.59	0.689	0.672	0.679	0.716	**0.725**
Avg. F1 Measure	0.643	0.641	0.72	0.7	0.697	0.731	**0.741**
CiteSeer							
	Content-Only	Flat-Mode	Flat-Binary	Flat-Count	Mode-Link	Binary-Link	Count-Link
Avg. Accuracy	0.607	0.618	0.634	0.644	0.658	0.664	**0.679**
Avg. Precision	0.551	0.55	0.58	0.579	**0.606**	0.597	0.604
Avg. Recall	0.552	0.547	0.572	0.573	0.601	0.597	**0.608**
Avg. F1 Measure	0.551	0.552	0.575	0.575	0.594	0.597	**0.606**
WebKB							
	Content-Only	Flat-Mode	Flat-Binary	Flat-Count	Mode-Link	Binary-Link	Count-Link
Avg. Accuracy	0.862	0.848	0.832	0.863	0.851	0.871	**0.877**
Avg. Precision	0.876	0.86	0.864	0.876	0.878	**0.879**	0.878
Avg. Recall	0.795	0.79	0.882	0.81	0.772	0.811	**0.83**
Avg. F1 Measure	0.832	0.821	0.836	0.84	0.82	0.847	**0.858**

The WebKB data set contains web pages from four computer science departments, categorized into topics such as faculty, student, project, course and a catch-all category, other. In our experiments we discard pages in "other" category, which generates a data set with 700 pages. After stemming and removing stop words, the dictionary contains 2338 words. For WebKB, we use the standard split along different schools.

On Cora and CiteSeer, for each experiment, we take one split as a test set, and the remaining two splits are used to train our model: one for training and the other for a validation set used to find the appropriate regularization parameter λ. On WebKB, we learned models for a variety of λ; here we show the best result.

6.1. Experiments

In our first set of experiments, we compared several baseline models—**content-only**, **flat-mode**, **flat-binary** and **flat-count**—with our models—**mode-link**, **binary-link** and **count-link**. In a second set of experiments, we examined the individual effects of the different categories of links: in-links, out-links and co-links (short for co-citation links). In a third set of experiments, we compared a variety of ordering schemes for the iterative categorization algorithm.

6.2. Results

Table 1 shows a summary of our results using four different metrics (accuracy, precision, recall and F1 measure) on three different data sets. Significance results are reported for paired t-test on the F1 measure. In this first set of experiments, all of the links (in-links, out-links and co-links) are used and we use a random ordering for the iterative classification algorithm.

In all three domains, **binary-link** and **count-link** outperform content-only at the 95% significance level. **Mode-link** outperforms **content-only** at the 95% significance level on Cora and CiteSeer, while the difference on WebKB is not statistically significant.

We also compare the structured logistic regression model with the corresponding flat models. The difference between the two is that for the flat models, all the features are used is a single regression model, while for our link-based models separate logistic regression models, with different λs, are learned for object attributes and link features. We found the following: **mode-link** outperforms **flat-mode** at the 95% significance level on Cora and CiteSeer; **binary-link** outperforms **flat-binary** at the 90% significance level for Cora, CiteSeer and WebKB; and **count-link** also outperforms **flat-count** at the 95% significance level on all three data sets.

The conclusions about the best link-based model are mixed. On all of the data sets, **count-link** and **binary-link** outperform **mode-link**, however the improvements are statistically significant at the 95% significance level for only Cora and CiteSeer. So making use of link statistics beyond the mode is clearly effective. However, the choice between the two models, **binary-link** versus **count-link** is less clear; while **count-link** gives better F1 measures than **binary-link**, the difference is not statistically significant.

In the next set of experiments, we investigated which type of link features are more predictive: in-links, out-links or co-links. Table 2 shows the results on three data sets. For **binary-link** and **count-link**, using all the links(in+out+co) always gives better results than using any in-links, out-links or co-links separately and

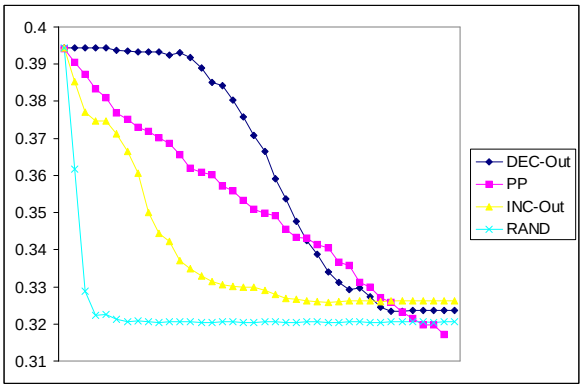

Figure 1. The convergence rates of different iteration methods on the CiteSeer data set.

out-links seems to contribute greatest to the improved performance.

In the last set of experiments, we examined various ICA ordering strategies. Our experiments indicate that final test errors with different ordering strategy have a standard deviation around 0.001. There is no significant difference with various link diversity to order the predictions. We also compared with an ordering based on the posterior probability of the categories as done in Neville and Jensen (2000), denoted PP. On Cora and WebKB, the random ordering outperforms PP and all other orderings with various link diversity, while PP gives the best result on CiteSeer. However, the differences between the results were not statistically significant.

While the different iteration schemes converge to about the same accuracy, their convergence rate varies. To understand the effect of the ordering scheme at a bit finer level of detail, Figure 1 shows an example of the test errors of the different iteration schemes for the CiteSeer data set (to make the graph readable, we show only ordering by increasing diversity of out-links (INC-Out) and decreasing diversity of out-links (DEC-Out); the results for in-links and co-links are similar). Our experiments indicate that order by increasing link diversity converges faster than ordering by decreasing link diversity, and the RAND ordering converges the most quickly at the start. Results on the Cora data set are consistent.

7. Conclusions

Many real-world data sets have rich structures, where the objects are linked in some way. Link mining targets data mining tasks on this richly-structured data. One major task of link mining is to model and exploit the link distributions among objects. Here we focus on using the link structure to help improve classification accuracy.

In this paper we have proposed a simple framework for modeling link distributions, based on link statistics. We have seen that for the domains we examined, a combined logistic classifier built over the object attributes and link statistics outperforms 1) a simple content-only classifier and 2) a single flat classifier over both the content and link attributes. More surprisingly, the mode of the link statistics is not enough to capture the dependence. Actually modeling the distribution of the link categories at a finer grain is useful.

Acknowledgments

This study was supported in part by the Advanced Research and Development Activity (ARDA) under Award Number NMA401-02-1-2018. The views, opinions, and findings contained in this report are those of the author(s) and should not be construed as an official Department of Defense position, policy, or decision unless so designated by other official documentation.

References

Blum, A., & Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. *COLT: Proceedings of the Workshop on Computational Learning Theory, Morgan Kaufmann Publishers.*

Chakrabarti, S. (2002). *Mining the web.* Morgan Kaufman.

Chakrabarti, S., Dom, B., & Indyk, P. (1998). Enhanced hypertext categorization using hyperlinks. *Proc of SIGMOD-98.*

Chellappa, R., & Jain, A. (1993). *Markov random fields: theory and applications.* Boston: Academic Press.

Cohn, D., & Hofmann, T. (2001). The missing link - a probabilistic model of document content and hypertext connectivity. *Neural Information Processing Systems 13.*

Cook, D., & Holder, L. (2000). Graph-based data mining. *IEEE Intelligent Systems, 15,* 32–41.

Craven, M., DiPasquo, D., Freitag, D., McCallum, A., Mitchell, T., Nigam, K., & Slattery, S. (1998). Learning to extract symbolic knowledge from the world wide web. *Proc. of AAAI-98.*

Dean, J., & Henzinger, M. R. (1999). Finding related pages in the World Wide Web. *Computer Networks, 31,* 1467–1479.

Dzeroski, S., & Lavrac, N. (Eds.). (2001). *Relational data mining.* Berlin: Kluwer.

Feldman, R. (2002). Link analysis: Current state of the art. *Tutorial at the KDD-02.*

Flach, P. A., & Lavrac, N. (2000). The role of feature construction in inductive rule learning. *Proc. of the ICML2000 workshop on Attribute-Value and Relational Learning: crossing the boundaries.*

Table 2. Average accuracy using in-links, out-links, co-links separately, and all (in+out+co) links with **mode-link**, **binary-link** and **count-link** models on Cora, CiteSeer and WebKB

	Mode-Link				Binary-Link				Count-Link			
data set	in	out	co	all	in	out	co	all	in	out	co	all
Cora	0.687	**0.717**	0.668	**0.717**	0.695	0.732	0.686	**0.754**	0.694	0.729	0.688	**0.758**
CiteSeer	0.632	0.651	0.628	**0.658**	0.629	0.659	0.624	**0.664**	0.631	0.644	0.636,,	**0.679**
WebKB	0.853	**0.857**	0.843	0.851	0.857	0.847	0.857	**0.871**	0.866	0.863	0.868	**0.877**

Getoor, L., Friedman, N., Koller, D., & Taskar, B. (2002). Learning probabilistic models with link uncertainty. *Journal of Machine Learning Research.*

Giles, C. L., Bollacker, K., & Lawrence, S. (1998). CiteSeer: An automatic citation indexing system. *ACM Digital Libraries 98.*

Hosmer, D., & Lemeshow, S. (1989). *Applied logistic regression.* New York: Wiley.

Hummel, R., & Zucker, S. (1983). On the foundations of relaxation labeling processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39,* 267–287.

Jensen, D. (1999). Statistical challenges to inductive inference in linked data. *Seventh International Workshop on Artificial Intelligence and Statistics.*

Jensen, D., & Goldberg, H. (1998). *AAAI fall symposium on AI and link analysis.* AAAI Press.

Joachims, T. (1998). Text categorization with support vector machines: learning with many relevant features. *Proc. of ECML-98.*

Joachims, T., Cristianini, N., & Shawe-Taylor, J. (2001). Composite kernels for hypertext categorisation. *Proc. of ICML-01.*

Kleinberg, J. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM, 46,* 604–632.

Kramer, S., Lavrac, N., & Flach, P. (2001). Propositionalization approaches to relational data mining. In S. Dzeroski and N. Lavrac (Eds.), *Relational data mining,* 262—291. Kluwer.

Kubica, J., Moore, A., Schneider, J., & Yang, Y. (2002). Stochastic link and group detection. *Proc. of AAAI-02.*

Lafferty, J., McCallum, A., & Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proc. of ICML-01.*

McCallum, A., & Nigam, K. (1998). A comparison of event models for naive bayes text classification. *AAAI-98 Workshop on Learning for Text Categorization.*

McCallum, A., Nigam, K., Rennie, J., & Seymore, K. (2000). Automating the construction of internet portals with machine learning. *Information Retrieval, 3,* 127–163.

Murphy, K., & Weiss, Y. (1999). Loopy belief propagation for approximate inference: an empirical study. *Proc. of UAI-99.* Morgan Kaufman.

Neville, J., & Jensen, D. (2000). Iterative classification in relational data. *Proc. AAAI-2000 Workshop on Learning Statistical Models from Relational Data.* AAAI Press.

Ng, A. Y., & Jordan, M. I. (2002). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. *Neural Information Processing Systems 14.*

Oh, H.-J., Myaeng, S. H., & Lee, M.-H. (2000). A practical hypertext categorization method using links and incrementally available class information. *Proc. of SIGIR-00.*

Page, L., Brin, S., Motwani, R., & Winograd, T. (1998). *The pagerank citation ranking: Bring order to the web* (Technical Report). Stanford University.

Popescul, A., Ungar, L., Lawrence, S., & Pennock, D. (2002). Towards structural logistic regression: Combing relational and statistical learning. *KDD Workshop on Multi-Relational Data Mining.*

Quinlan, J. R., & Cameron-Jones, R. M. (1993). FOIL: A midterm report. *Proc. of ECML-93.*

Slattery, S., & Craven, M. (1998). Combining statistical and relational methods for learning in hypertext domains. *Proc. of ILP-98.*

Taskar, B., Abbeel, P., & Koller, D. (2002). Discriminative probabilistic models for relational data. *Proc. of UAI-02* (pp. 485–492). Edmonton, Canada.

Taskar, B., Segal, E., & Koller, D. (2001). Probabilistic classification and clustering in relational data. *Proc. of IJCAI-01.*

Yang, Y., Slattery, S., & Ghani, R. (2002). A study of approaches to hypertext categorization. *Journal of Intelligent Information Systems, 18,* 219–241.

Zhang, T., & Oles, F. J. (2001). Text categorization based on regularized linear classification methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 5–31.