

# Context-Based Classification for Link Data\*

Yonghong Tian<sup>1,2</sup>, Wen Gao<sup>1,2</sup>, Tiejun Huang<sup>2</sup>

<sup>1</sup> Institute of Computing Technology, Chinese Academy of Sciences,  
Beijing 100080, P. R. China

<sup>2</sup> Graduate School of Chinese Academy of Sciences,  
Beijing 100039, P. R. China  
{Yhtian, Tjhuang, Wgao}@ict.ac.cn

**Abstract.** In Web-based e-learning, an up-to-date catalogue of subject-specific Web resources can effectively offer inexperienced students with an advanced academic portal on the Web. To automatically construct such academic Web resource catalogue, a key issue is how to classify the collected Web pages. However, existing link-based classification methods treat all neighbors equally in the categorization of the target objects. In this paper, we propose a context-based classification approach that can scale well for noisy and heterogeneous link data such as Web pages. We quantitatively measure the contextually topical dependencies between linked objects using the dependence functions, which are then exploited to classify the target objects in the link structure. Experimental results show that the proposed classification model can better capture the link regularities and can facilitate better categorization of linked objects.

**Keywords.** Management of learning resource, link-based classification, context model

## 1 Introduction

The Web has become a most important information source and knowledge base for scientific, educational and research applications. In Web-based e-learning, an up-to-date catalogue of subject-specific Web resources can effectively offer inexperienced students with an advanced academic portal on the Web. To automatically construct such academic Web resource catalogues, a key issue is how to classify the collected Web pages. In this paper, Web page categorization is viewed as an application of link-based classification methods, in which the topical correlation between linked pages should be effectively utilized.

In general, the link data can be described by a graph in which the nodes are objects and the edges are links between objects. Links among the objects may demonstrate certain patterns, which may be helpful for classifying the linked objects (e.g., [1], [2]). However, many link data such as Web pages are in essence heterogeneous, often accompanied with much noise, it is important to design link-based classification

---

\* This work was supported by Science and Technology Tackle Key Problem of the National “Tenth Five-year” Plan of China under Grand. 2001BA101A07.

methods that can scale well in the context of noise and heterogeneity. Existing link-based classification methods treat all neighbors equally in the categorization of the target objects. Obviously, the class information or link features from such noisy neighbors would increase the uncertainty of the categories of the target objects. Ideally, we expect a classifier to recognize such noisy objects and effectively reduce their influence on the classification. Towards this end, we develop a rich context-based probabilistic model to capture the topical dependencies between linked objects.

An innovative aspect of our model is the explicit use of the context. Generally speaking, two objects are said to be contextually dependent if the interpretation of one object is highly related to the other [3]. Therefore, we can model contextual dependencies between objects in the link data. More importantly, we can exploit the learned context models to facilitate classification. In this paper, the contextual dependence is measured by three mechanisms, i.e., class dependency, link structure, and link features. Thus we define a feasible quantitative measure of contextual dependencies, which is called *dependence function*. Using this measure, all linked neighbors are no longer equally treated by the classifier in the categorization of the target objects. Furthermore, we formally describe how to exploit the dependency functions for classification. Experimental results demonstrate that the proposed *context-based link classification* approach can provide better prediction for the attributes of linked objects.

Starting with a simple discussion of the existing linked-based classification models in Section 2, we describe the proposed context-based link classification approach in Section 3. Experimental results are presented in Section 4. We conclude in Section 5.

## 2 Problem Formulation

Formally, the typical link-based classification problem is modeled as follows: Given the link data represented as an object graph  $G = (O, L)$ , assign a class label  $c_i \in \mathbf{C} = \{c_1, \dots, c_{|\mathbf{C}|}\}$  to  $O_i \in O$ . For simplicity, this paper uses the same notation  $O_i$  for the identity of the object and its content features such as textual keywords.

Generally speaking, for classifying a given linked object, we need to consider its content features and the information from its neighbors. Here we restrict the neighborhood to within a radius of one or two, since exploring larger neighborhoods can be futile and dangerous [1]. Let  $N(O_i)$  to denote the linked neighbors of  $O_i$ ,  $C_{N(O_i)}$  to represent the set of class values of objects in  $N(O_i)$ . Assuming that there is a probability distribution on the link graph, we want to choose  $c_i$  to maximize  $\Pr[c_i | O_i, C_{N(O_i)}]$  (i.e., *maximum a posterior* estimation, MAP):

$$\hat{c}_i = \arg \max_{c_i} \Pr[c_i | O_i, C_{N(O_i)}]. \quad (1)$$

Under this optimization framework, different probabilistic models are developed to calculate the category posterior probabilities of the target object given its contextual objects, i.e.,  $\Pr[c_i | O_i, C_{N(O_i)}]$ . One oft-used *neighboring-class* model is to assume that given  $C_i = c_i$ , the content features of  $O_i$  are independent of all the classes of all

its neighbors  $N(O_i)$  (e.g., [1], [2]). Instead of exploiting the class information from neighbors, Lu and Getoor [4] also proposed a *link-distribution* model that describes the linkage information between an object and a set of its neighbors, and then supports discriminative models describing both the link features and the attributes of linked objects.

In a complex link environment such as in the Web, Web pages often contain many noisy objects and noisy links (e.g., advertisements, navigational bar) that are irrelevant to their main content. In above two models, however, all neighbors are assumed to have the same influence on classifying the target object. Obviously, the class information of the noisy objects would increase the uncertainty of the categorization of the target objects. Ideally, we expect a link-based classifier to recognize such noisy objects or effectively reduce their influence on the classification of the target objects. Towards this end, we need to develop a rich context-based probabilistic model to capture the dependencies between objects, which will be addressed in the next section.

### 3 Context-Based Link Classification Model

#### 2.1 Modeling Context on Link Graph

Generally speaking, two objects are said to be contextually dependent if the interpretation of one object is highly related to the other. This means there is some *dependence function* between the two objects. In particular, we represent the context of a given object as its most relevant neighbors in the link structure. Based on the linkage information, there are many ways to quantify the dependence function. The simplest way is to set the dependence function value between two linked pages  $u$  and  $v$  to the constant *one* [8]. To reduce the influence of nepotistic links, Dean and Henzinger [9] showed that connectivity and content analysis should be integrated to capture the more complex correlations among linked objects.

Thus in this paper, we argue that the dependence function should be measured by three mechanisms, i.e., class dependency, link structure and link features. The rationales of the three context parameters are as follows: For the class dependency, we refer it to the correlation between the categories of linked objects. It is naturally measured in terms of the mutual information. By this measure, the higher the mutual information between two objects, the easier it is to estimate the target object given the other object, or vice versa. For the link structure, we consider that two objects (e.g., Web pages) that are close to are generally more informative about each others' categories. Therefore, the link structure can be measured by the link distance between two objects. For the link features, we refer them to the indicators of "*link tightness*" between two objects. For example, a higher weight can be assigned to the neighbor with more links to the target object than that with a single link. More formally, we have:

**Definition 1** Given an object graph  $G = (O, L)$ , a function  $\sigma(\cdot, \cdot)$  of two variables is called *dependence function* between two objects  $O_i \in O$  and  $O_j \in O$  if it satisfies

- (1)  $\sigma(O_i, O_j) \geq 0$ , and
- (2)  $\sum_{O_j \in N(O_i)} \sigma(O_i, O_j) = 1$ , and
- (3) for  $\forall O_k \in N(O_i), O_k \neq O_j, I(O_i; O_j) \geq I(O_i; O_k) \Rightarrow \sigma(O_i, O_j) \geq \sigma(O_i, O_k)$ ,  
 $d(O_i, O_j) \leq d(O_i, O_k) \Rightarrow \sigma(O_i, O_j) \geq \sigma(O_i, O_k)$ ,  
 $\varphi(O_i, O_j) \geq \varphi(O_i, O_k) \Rightarrow \sigma(O_i, O_j) \geq \sigma(O_i, O_k)$ ,

where  $I(O_i; O_j)$ ,  $d(O_i, O_j)$  and  $\varphi(O_i, O_j)$  are the mutual information, link distance and link feature between  $O_i$  and  $O_j$  respectively.  $I(O_i; O_j)$  can be calculated directly according to the definition of the mutual information [7]. For  $d(O_i, O_j)$ , we use the Euclidian distance  $D_2(O_i, O_j)$ . For  $\varphi(O_i, O_j)$ , we can use the frequency of links between  $O_i$  and  $O_j$  to the total number of links between  $O_i$  and its neighborhood. We call this the **link-count** model. However, this model will induce additional bias since the nepotistic links are rampant on the Web today. To further reduce the influence of such noisy links, we use the frequency of different link modes co-existing between  $O_i$  and  $O_j$ . In this case, we use the term **mode-count** model. For simplicity, we use  $\sigma_{i,j}$  to denote  $\sigma(O_i, O_j)$ , use  $\sigma_i$  to denote the *dependence vector* of object  $O_i$ . Note that for each object  $O_i$ ,  $M_i = |\sigma_i|$ .

There are many choices of embodied forms of the *dependence function* between two objects  $O_i$  and  $O_j$ . Here we use the following form:

$$\sigma_{i,j} = \exp\left(-\beta_1 \frac{d(O_i, O_j)}{\varphi(O_i, O_j)}\right) I(O_i; O_j), \quad (2)$$

where  $\beta_1$  is a parameter to control the sensitivity of the dependence function value to the ratio  $\varphi(O_i, O_j)/d(O_i, O_j)$ , which is set as follows: for the mode-count model,  $0.8 \leq \beta_1 \leq 1$ ; for the link-count model,  $0.2 \leq \beta_1 \leq 0.3$ . Thus, instead of using the assumption that the class variables of neighbors are independent given the class label  $c_i$  of  $O_i$ , we have the following assumption:

$$\Pr[c_i | C_{N(O_i)}] = \sum_{O_{ik} \in N(O_i)} \sigma_{i,ik} \Pr[c_i | C_{ik} = c_j]. \quad (3)$$

Here different neighbors have different influences on the categorization of  $O_i$ . Thus the resulting classifier can be written as:

$$\hat{c}_i = \arg \max_{c_i} \Pr[c_i | O_i, C_{N(O_i)}] = \arg \max_{c_i} \frac{\Pr[c_i | O_i] \Pr[c_i | C_{N(O_i)}]}{\Pr[c_i]}. \quad (4)$$

## 2.2 Context Optimization

However, for each  $O_i \in \mathcal{O}$ , there may be a large number of neighbors. A typical academic paper, a patent in the Patent Database, and an average Web page all have typi-

cally more than ten citations or out-links [1]. Naïvely restricting the expansion to out-links within the site or domain would miss many valuable links at the same time. Hence to avoid the “context dilution” problem, a more sophisticated method should be used to reduce the objects’ contextual space which is often initialized as their neighborhoods.

The context optimization on the link data can be modeled as follows: For a given object  $O_i$  in a link graph  $G=(O, L)$ , find a subset  $NC(O_i)$  of  $N(O_i)$  that minimizes the conditional entropy  $H(C_i | O_i, C_{NC(O_i)})$ . Here  $NC(O_i)$  denotes the reduced neighborhood of  $O_i$ , i.e., its *de facto* context. For simplicity, we use  $K_i = |NC(O_i)|$ . In general,  $K_i \leq M_i$ . In this paper, the context optimization problem can be solved using the dependence function. Intuitively, some neighbors with relatively low dependence function values would be removed from the context space of the target object. Formally, the context optimization for a given object  $O_i$  is equivalent to find  $K_i$  neighbors in  $N(O_i)$  with maximal  $\sigma(\cdot, \cdot)$  values, i.e.,

$$NC(O_i) = \underset{\substack{NC(O_i) \subseteq N(O_i), \\ |NC(O_i)| = K_i}}{\operatorname{argmin}} H(C_i | O_i, C_{NC(O_i)}) = \underset{\substack{NC(O_i) \subseteq N(O_i), \\ |NC(O_i)| = K_i}}{\operatorname{argmax}} \sum_{O_{ik} \in NC(O_i)} \sigma_{i,ik}. \quad (5)$$

Without loss of generality, let  $\sigma_{i,i1} \geq \sigma_{i,iM_i}$ , then the optimized context of  $O_i$  is  $NC(O_i) = \{O_{i1}, \dots, O_{iK_i}\}$ .

Theoretically, if the neighbor  $O_{ik}$  is irrelevant with  $O_i$ , then  $I(O_i; O_{ik}) \rightarrow 0$ , which will approximately result in  $\sigma_{i,ik} \rightarrow 0$ . Thus we set a threshold  $\sigma_{\perp}$  for the dependence function values, i.e., if  $\sigma_{i,ik} \geq \sigma_{\perp}$ , then  $O_{ik} \in NC(O_i)$ . Furthermore, the threshold can be determined easily. For example,  $\sigma_{\perp} = 0.5\bar{\sigma}_i$  or  $\sigma_{\perp} = 0.01$ .

After the context optimization and the re-normalization of the dependence function values (i.e., let  $\sum_{O_{ik} \in NC(O_i)} \sigma_{i,ik} = 1$ ), Eq (3) can then be rewritten as

$$\Pr[c_i | C_{N(O_i)}] = \sum_{O_{ik} \in NC(O_i)} \sigma_{i,ik} \Pr[c_i | C_{ik} = c_j]. \quad (6)$$

The results calculated from this equation instead of from Eq. (3) can be applied in Eq. (4) to assign a new class label to  $O_i$ .

### 2.3 The Classification Framework

Generally, the neighbors of the linked objects may be partly or fully unlabeled. Inspired by the works in [1], [4], this paper also uses the iterative inference procedure. Let  $O = O^L \cup O^U$  where  $O^L$  and  $O^U$  denote the objects in labeled and unlabeled sub-datasets respectively. Firstly, a bootstrap step is used to assign an initial class label to each object  $O_i$  in  $O^U$ , only using the content features. Then an iterative step is used to refine the classification of the objects in  $O^U$  until the algorithm terminates, which includes several sequential operations such as context modeling, influence propagation and termination check. Note that in each iteration step, the dependence

vector  $\sigma_i$  for each object  $O_i$  will be re-calculated, thus the corresponding context  $NC(O_i)$  will be optimized again, based on the current assignments to linked objects. The framework of the algorithm is as follows:

**Algorithm 1. (The Context-Based Link Classification)**

**Input:** Link data  $G = (O, L)$ , let  $O = O^L \cup O^U$ .

**Given:** A predefined class taxonomy  $C = \{c_1, \dots, c_{|C|}\}$  for objects.

**Step 1 (Bootstrap):** For each  $O_i$  in  $O^U$ , assign an initial class label to the object  $O_i$  only using its content features..

**Step 2 (Iteration):** Iteratively classify each object  $O_i$  in  $O^U$  until termination:

Find the initial neighborhood for  $O_i$ :  $N(O_i) = \{O_j \mid O_j \in O, 0 < d(O_i, O_j) < 2\}$ .

**2.1 (Context Modeling)** Based on the current assignments to linked objects, calculate  $I(O_i, O_{ik})$ ,  $d(O_i, O_{ik})$ ,  $\phi(O_i, O_{ik})$  and  $\sigma_{i,ik}$  for all  $O_{ik} \in N(O_i)$ .

Select  $O_{ik}$  with  $\sigma_{i,ik} \geq \sigma_{\perp}$  to form the current  $NC(O_i)$ .

**2.2 (Influence Propagation)** Calculate the current  $\Pr[c_i \mid C_{N(O_i)}]$  by Eq. (6) using all the class information of its contextual objects  $O_j \in NC(O_i)$ , and update the class label of  $O_i$  by Eq. (4).

**2.3 (Termination Check)** If the predefined convergence criteria are met, output final results.

**Output:** The final class label for each object  $O_i$  in  $O^U$ .

Several criteria are used to determine whether the iteration process will be terminated, including the convergence of the average entropies over all unlabeled (or target) objects, the consistency of MAP estimates of object categories between two consecutive iterations, and the iteration upper. In this paper, 5 iterations were sufficient to ensure convergence, thus the iteration upper was set to 5.

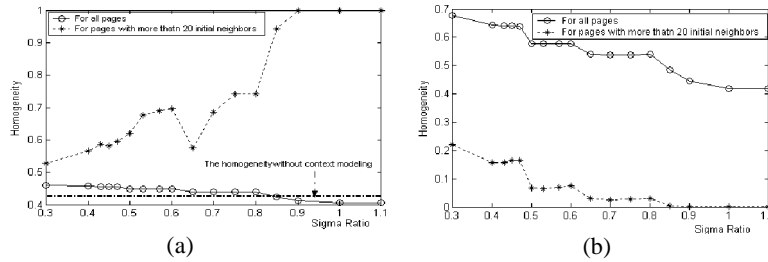
## 4 Experiments and Results

Two experiments were designed to evaluate the proposed classification algorithm on the standard WebKB dataset [5], [6]. The WebKB dataset contains approximately 4100 pages from four computer science departments, with a seven-valued attribute representing their types (i.e., faculty, student, staff, department, project, course and other), and 10,900 links between pages. The first experiment dealt with the evaluation of context modeling capability. The second experiment evaluated various link-based classification algorithms with and without context modeling.

As for *context modeling capability*, we denote whether the additional context modeling can reduce the high-dimensional context space (called *dimensionality reduction ability*), and can automatically identify as more linked objects of the same or highly relevant categories as possible from the objects' neighborhood (called *homogeneity*). For simplicity, this paper uses the frequency of contextual objects that are of the same category with the target object to measure the homogeneity of its context space, and uses the ratio between the dimensionalities of context spaces before and after context optimization to measure the dimensionality reduction ability.

Figure 1 depicts the curves of the average homogeneity and the curves of the average dimensionality ratio under different  $\sigma_{\perp} / \bar{\sigma}_i$  on WebKB dataset. Here we high-

light the curves for the pages with more than 20 initial neighbors in the dataset. Note that the homogeneity ratios without context modeling are only 0.41 and 0.34 respectively for all pages and for pages with more than 20 initial neighbors. And after the context optimization process, these ratios become 0.43 and 0.75. Clearly, the context modeling can significantly increase the context homogeneity and reduce the high-dimensional context space for those pages with relatively more linked neighbors. Namely, we can effectively purify the pages' neighborhood via context modeling and optimization.



**Fig. 1.** The curves for (a) the average homogeneity and (b) the average dimensionality ratio of context space under different  $\sigma_{\perp} / \bar{\sigma}_i$  on the WebKB dataset.

Then we tested the conjecture that by link context modeling we can improve the prediction of descriptive attributes. We evaluated the following set of models:

- **Baseline:** The Naïve Bayes model uses only textual features of the page to predict the category of the page. This model is also used for the basic text classification tasks in the following link-based classification procedures.
- **Neighboring-Class Model [1]:** This model equally utilizes the class information of all linked neighbors to predict the category.
- **Link-Distribution Model [4]:** Based on the counts link statistics for in-links, out-links and co-citations, a logistic regression model is built for link features between the target pages and its neighborhood.
- **Context-Based Models:** We may use mode-count model and link-count model to capture the link feature between the target page and each of its neighbors. Hence there are two choices of the context-based classification models.

*Table 1.* Comparison of average accuracy, precision, recall and F1 scores on each of the classification tasks using different link-based models.

	Without context modeling			With context modeling	
	Baseline (Content-Only)	Neighboring Class Model	Link-Distri. Model	Link-Count Model	Mode-Count Model
Avg. Accuracy	74.3	85.4	86.1	86.6	<b>89.1</b>
Avg. Precision	75.3	84.1	85.4	86.7	<b>88.2</b>
Avg. Recall	70.6	81.2	82.8	82.3	<b>82.9</b>
Avg. F1 Measure	72.9	82.6	84.0	84.4	<b>85.5</b>

Table 1 summarizes the average results of 4-fold cross-validation tests using the five classification models on the WebKB dataset. We see that both models of context-based classification significantly improve the average accuracy, precision, recall and

F1 scores, although mode-count based models seem to be superior. Hence in the other experiments throughout this paper, the context-based classification utilized the mode-count model to characterize the link features.

In practice, we applied the context-based classification model on the automatic construction of academic Web resource catalogue. In this process, a focused crawler analyzes the hyperlink structure of web pages in a given seed set to find the related pages or sites, which then are assigned to different labels by context-based classifiers. All these newly discovered pages or sites were verified and labeled by domain experts, and then will be added partially to the Phy-Math Portal of Chinese Science Digital Library project (<http://phymath.csdl.ac.cn>).

## 5 Conclusion

Many link data such as Web pages are in essence heterogeneous, often accompanied with much noisy information. Hence this paper has explored how to model the contextual dependencies between objects in the link data, and exploited the learned context models to facilitate classification. It should be noted that the link context modeling technique could also be used to other Web-related applications such as enhancing Web search. We plan to further investigate its new applications in the future.

## References

1. Chakrabarti, S., Dom, B., & Indyk P.: Enhanced hypertext categorization using hyperlinks. In: Proceedings of SIGMOD'98. Seattle, Washington, USA: ACM Press. (1998) 307-318.
2. Oh, H. J., Myaeng, S. H. & Lee, M.-H.: A practical hypertext categorization method using links and incrementally available class information. In: Proceedings of 23<sup>rd</sup> ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR-00). Athens, Greece: ACM Press, New York, NY, USA. (2000) 264-271.
3. Brzillon P. Context in problem solving: a survey. *The Knowledge Engineering Review*, 14(1), 1999, 1-34.
4. Lu, Q. & Getoor, L.: Link-based Classification. In: Proceedings of 12th Int. Conf. on Machine Learning (ICML-2003), Washington DC, AAAI Press, Menlo Park, US. (2003).
5. Craven, M., DiPasquo, D., Freitag, D. McCallum, A., Mitehell, T., Nigam, K., & Slattery, S.: Learning to extract symbolic knowledge from the world wide web. In: Proceedings of the AAAI-98. (Madison, US): AAAI Press (1998) 509-516.
6. Slattery, S., Craven, M.: Combining statistical and relational methods for learning in hypertext domains. In D. Page (Ed.): Proceedings of 8th Int. Conf. on Inductive Logic Programming (ILP-98), no. 1446 in Lecture Notes in Computer Science, (Madison, US), Springer Verlag, Heidelberg, DE. (1998) 38-52.
7. Gray, R. M. *Entropy and Information Theory*. New York, NY: Springer-Verlag. (1990).
8. Kleinberg, J. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46, 1999, 604-632.
9. Dean, J. & Henzinger, M. Finding related pages in the World Wide Web. In: Proceedings of 8th international World Wide Web Conference (WWW8), Toronto, Canada, Elsevier Science B.V. (1999) 389-401.