Review of the web page classification approaches and applications

Luu Ngoc Do, Quang Nhat Vo
School of Electronic and Computer Engineering
Web Mining Class
doluungoc@gmail.com, vqnhat@gmail.com

***Abstract*—The growth of Internet and World Wide Web is directly proportional with the amount of sharing information in the world. Therefore, the Web information retrieval tasks such as crawling, searching, maintaining Web directories, knowledge base extracting, are necessary to control the nature of webs content. Especially, the classification of web page plays a vital role in these tasks. In this report, we will review the applications and state-of-the-art algorithms and specific features for web page classification. We also implement a webpage classification system using text feature and Naive Bayes model. The system is trained and tested with WebKB dataset for four classes of webpage.**

I. Introduction

The web page classification is the process of categorizing a Web page to one or more labels. Classification is traditionally posed as a supervised learning problem in which a set of labeled data is used to train a classifier which can be applied to label future examples. The general problem of Web page classification can be divided into more specific problems: subject classification, functional classification, sentiment classification, and other types of classification. Subject classification is concerned about the subject or topic of a Web page. For example, judging whether a page is about arts, business, or sports is an instance of subject classification. Functional classification cares about the role that the Web page plays. For example, deciding a page to be a personal homepage, course page or admission page is an instance of functional classification. Sentiment classification focuses on the opinion that is presented in a Web page, that is, the authors attitude about some particular topic. In this report, we only focus on the subject classification. Based on the number of classes in the problem, classification can be divided into binary classification and multiclass classification, where binary classification categorizes instances into exactly one of two classes, and multiclass classification deals with more than two classes. Based on the number of classes that can be assigned to an instance, classification can be divided into single-label classification and multi-label classification. In single-label classification, one and only one class label is to be assigned to each instance, while in multi-label classification, more than one class can be assigned to an instance. Based on the organization of categories, Web page classification can also be divided into flat classification and hierarchical classification. In flat classification, categories are considered parallel, that is, one category does not supersede another, while in hierarchical classification the categories are organized in a hierarchical tree-like structure, in which each category may have a number of subcategories.

II. Applications of web page classification

A. Improving quality of search result

Query ambiguity is critical problem that affect the quality of search results. Various approaches have been proposed to improve retrieval quality by disambiguating query terms. Chekuri et al. [1997] studied automatic Web page classification in order to increase the precision of Web search. A statistical classifier, trained on existing Web directories, is applied to new Web pages and produces an ordered list of categories in which the Web page can be placed. At query time the user is asked to specify one or more desired categories so that only the results in those categories are returned, or the search engine returns a list of categories under which the pages would fall. This approach works when the user is looking for a known item. Approaches proposed by Dumais [2000] and Kaki [2005] classify search results into a predefined hierarchical structure and presents the categorized view of the results to the user. Their user study demonstrated that the category interface is liked by the users better than the result list interface, and is more efficient for users to find the desired information.

B. Building Focused Crawler

When only domain-specific queries are expected, performing a full crawl is usually inefficient. Chakrabarti et al. [1999] proposed an approach called focused crawling, in which only documents relevant to a predefined set of topics are of interest. In this approach, a classifier is used to evaluate the relevance of a Web page to the given topics so as to provide evidence for the crawl boundary.

C. Extracting Knowledge Base

A knowledge base (KB) is a technology used to store complex structured and unstructured information from the World Wide Web to make a computer understandable environment. Craven et al. [1998] provided a knowledge base extracting system that contains three steps: recognize class instances by classifying webs content, recognize relation instances by classifying chains of hyperlinks and recognize class and relation instances by extracting small fields of text.

III. Feature Selection

Not only the textual contents but also the additional components of the web such as HTML tags, hyperlinks and anchor text can used as the informative features for classification. These features can be divided into two classes: on-page

features, which are directly located on the page to be classified, and features of neighbors, which are found on the pages related in some way to the page to be classified.

A. On-page Features

The textual content is the most straightforward feature that one may use. However, due to the variety of uncontrolled noises in Web pages, directly using a bag-of-words representation for all terms may not achieve top performance. N-gram representation is another method that has been found to be useful. Mladenic [1998] suggested an approach to automatic Web page classification based on the Yahoo! hierarchy. In this approach, each document is represented by a vector of features, which includes not only single terms, but also up to five consecutive words. The advantage of using n-gram representation is that it is able to capture the concepts expressed by a sequence of terms (phrases), which are unlikely to be characterized using single terms. Imagine a scenario of two different documents. One document contains the phrase New York. The other contains the terms new and york, but the two terms appear far apart. A standard bag-of-words representation cannot distinguish between them, while a 2-gram representation can. However, an n-gram approach has a significant drawback: it usually generates a space with much higher dimensionality than the bag-of-words representation does. One obvious feature that appears in HTML documents but not in plain text documents is HTML tags. It has been demonstrated that using information derived from tags can boost the classifiers performance. Golub and Ardo [2005] derived significance indicators for textual content in different tags. In their work, four elements from the Web page were used: title, headings, metadata, and main text. They showed that the best result was achieved from a well-tuned linear combination of the four elements. Rather than deriving information from the page content, Kan and Thi [2005] demonstrated that a Web page can be classified based on its URL. While not providing ideal accuracy, this approach eliminates the necessity of downloading the page and therefore reduces the processing time. Sujatha et al. [2013] used the main text, anchor text and URL for a co-training approach to classify irrelevant pages on current-day academic websites. Their goal is adapting a classifier trained on a labeled dataset of web pages to a related environment containing newer types of web pages in the context of focused crawling for researcher homepages. This approach can effectively incorporate unlabeled data to improve the classification performance.

B. Features of Neighbors

When exploring the features of neighbors, some assumptions are implicitly made in existing work. Usually, it is assumed that, if pages **pa** and **pb** belong to the same category, pages neighboring them in the Web graph share some common characteristics. This assumption does not require that the neighboring pages belong to the same category as **pa** and **pb** do. This assumption is referred as the weak assumption. Under the weak assumption, a classifier can be derived from the features of the neighboring pages of training examples, and used to predict the categories of testing examples based on the features of their neighbors. In subject classification, a stronger assumption is often made that a page is much more likely to be surrounded by pages of the same category. In other words, the presence of many sports pages in the neighborhood of **pa** increases the probability of **pa** being in sports. This assumption is referred as the strong assumption. Existing research has mainly focused on pages within two steps of the page to be classified. At a distance no greater than two, there are six types of neighboring pages according to their hyperlink relationship with the page in question: parent, child, sibling, spouse, grandparent, and grandchild, as illustrated in Figure 1.

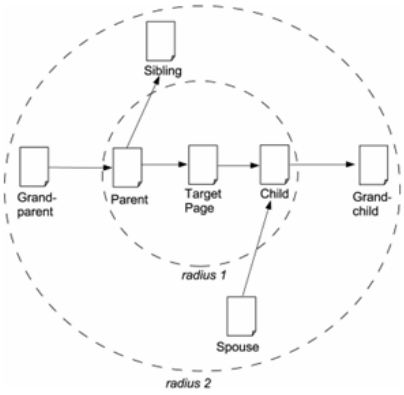

Fig. 1. Neighbor pages

In general, Chakrabarti et al. [1998] showed that directly incorporating text from parent and child pages into the target page is not good because parent and child pages are likely to have different topics than the target page. Oh et al. [2000] required the content of neighbors to be sufficiently similar to the target page. Using a portion of content on parent and child pages, especially the content near enough to the hyperlink that points to the target page can reduce the influence from the irrelevant part of neighboring pages. Usually, title, anchor text, and the surrounding text of anchor text on the parent pages are found to be useful. Sibling pages are even more useful than parents and children. This was empirically demonstrated by Qi and Davison [2006].

IV. Algorithms

The k-Nearest Neighbor classifiers require a document dissimilarity measure to quantify the distance between a test document and each training document. Most existing kNN classifiers use cosine similarity or inner product. Based on the observation that such measures cannot take advantage of the association between terms, Kwon and Lee [2000] developed an improved similarity measure that takes into account the term co-occurrence in documents. The intuition is that frequently co-occurring terms constrain the semantic concepts of each other. The more co-occurring terms two documents have in common, the stronger the relationship between the two documents. Their experiments showed performance improvements of the new similarity measure over cosine similarity and inner product measures.

Text classication is the task of classifying documents by their content: that is, by the words of which they are comprised. This problem is a great practical application for webpage classification because of the massive volume of online text

available through the Web pages, Internet news feeds, electronic mail, corporate databases, medical patient records and digital libraries. The state-of-art algorithm for classifying the text content is Nave Bayes with Bernoulli model discussed by Shimodaira [2014]. In the Bernoulli document model, a document is represented by a feature vector with binary elements taking value 1 if the corresponding word is present in the document and 0 if the word is not present. If we have a vocabulary V containing a set of $|V|$ words, then the t^{th} dimension of a document vector corresponds to word w_t in the vocabulary. Let bi be the feature vector for the i^{th} document D^i, then the t^{th} element of b_i, written bit, is either 0 or 1 representing the absence or presence of word w_t in the I_{th} document. The document likelihood $P(D^i|C)$ is written:

$$P(D^i|C)\ P(b_i|C) = \prod_{t=1}^{|V|}[b_{it}P(w_t|C) + (1 - b_{it})(1 - P(w_t|C))] \quad (1)$$

Finally, to classify an unlabeled document D^j, we estimate the probability for each class and select the maximum probability as the class of D^j:

$$\begin{aligned} P(C|D^i) &= P(C|b_j) \propto P(b_j|C)P(C) \\ &\propto P(C)\prod_{t=1}^{|V|}[b_{jt}P(w_t|C) + (1 - b_{jt})(1 - P(w_t|C))] \end{aligned} \quad (2)$$

Co-training, introduced by Blum and Mitchell [1998], is an approach that makes use of both labeled and unlabeled data to achieve better accuracy. In a binary classification scenario, two classifiers that are trained on different sets of features are used to classify the unlabeled instances. The prediction of each classifier is used to train the other. Compared with the approach which only uses the labeled data, this co-training approach is able to cut the error rate by half.

V. Experimental results with content-based web classification

We implement a webpage classification system based on the Bernoulli document model and nave bayes classifier. The WebKB data set (Craven et al., 1998) contains 8145 web pages gathered from university computer science departments. The collection includes the entirety of four departments, and additionally, an assortment of pages from other universities. The pages are divided into seven categories: student, faculty, staff, course, project, department and other. We select four most populous non-other classes: student, faculty, course and project. The training is performed on webpages belong from university of Cornell, Washington, Texas and miscellaneous pages collected from other universities. The tested pages is from Wisconsin university. We limit the vocabulary to the 300 words that have the most variance of appearance probability in 4 classes. We define the classification accuracy in each class as:

$$acc = \frac{\#ofcorrectedclassificationineachclass}{\#ofwebpageineachclass} \quad (3)$$

The classification result showed in Table I demonstrate the effectiveness of this content-based model.

TABLE I. Classification accuracy and the number of training and testing pages

Classes	Faculty	Course	Student	Project
# of training pages	1082	845	1485	479
# of testing pages	42	85	156	25
accuracy	0.8182	0.8851	0.7595	0.8148

References

[1] BLUM, A. AND MITCHELL, T. 1998. Combining labeled and unlabeled data with co-training. In Proceedings of the 11th Annual Conference on Computational Learning Theory (COLT). ACM Press, New York, NY, 92100.

[2] CHAKRABARTI, S., VAN DEN BERG, M., AND DOM, B. 1999. Focused crawling: A new approach to topic-specific Web resource discovery. In Proceeding of the 8th International Conference on World Wide Web (WWW). Elsevier, New York, NY, 16231640.

[3] CHAKRABARTI, S., DOM, B. E., AND INDYK, P. 1998. Enhanced hypertext categorization using hyperlinks. In Proceedings of the ACM SIGMOD International Conference on Management of Data. ACM Press, New York, NY, 307318.

[4] CHEKURI, C., GOLDWASSER, M., RAGHAVAN, P., AND UPFAL, E. 1997. Web search using automated classification. In Proceedings of the Sixth International World Wide Web Conference (Santa Clara, CA). Poster POS725.

[5] CRAVEN, M., DIPASQUO, D., FREITAG, D.,MCCALLUM, A.,MITCHELL, T., NIGAM, K., AND SLATTERY, S. 1998. Learning to extract symbolic knowledge from the World Wide Web. In Proceedings of the Fifteenth National Conference on Artificial Intelligence. AAAI Press, Menlo Park, CA, 509516.

[6] DUMAIS, S. AND CHEN,H. 2000. Hierarchical classification ofWeb content. In Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM Press, New York, NY, 256263.

[7] GOLUB, K. AND ARDO, A. 2005. Importance ofHTMLstructural elements and metadata in automated subject classification. In Proceedings of the 9th European Conference on Research and Advanced Technology for Digital Libraries (ECDL). Lecture Notes in Computer Science, vol. 3652. Springer, Berlin, Germany, 368378.

[8] HIROSHI SHIMODAIRA, Text Classication using Naive Bayes, Undergraduate Course: Informatics 2B - Algorithms, Data Structures, Learning (INFR08009), The University Of Edinburgh, 11 February 2014.

[9] KAKI, M. 2005. Findex: Search result categories help users when document ranking fails. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI). ACM Press, New York, NY, 131140.

[10] KAN, M.-Y. AND THI, H. O. N. 2005. Fast Webpage classification using URL features. In Proceedings of the 14th ACM International Conference on Information and Knowledge Management (CIKM). ACM Press, New York, NY, 325326.

[11] KWON, O.-W. AND LEE, J.-H. 2000. Web page classification based on k-nearest neighbor approach. In Proceedings of the 5th International-Workshop on Information Retrieval with Asian Languages (IRAL). ACM Press, New York, NY, 915.

[12] MLADENIC, D. 1998. Turning Yahoo into an automatic Web-page classifier. In Proceedings of the European Conference on Artificial Intelligence (ECAI). 473474.

[13] OH, H.-J., MYAENG, S. H., AND LEE, M.-H. 2000. A practical hypertext catergorization method using links and incrementally available class information. In Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM Press, New York, NY, 264271

[14] QI, X. AND DAVISON, B. D. 2006. Knowing a Web page by the company it keeps. In Proceedings of the 15th ACM International Conference on Information and Knowledge Management (CIKM). ACM Press, New York, NY, 228237.

[15] SUJATHA DAS G., CORNELIA CARAGEA, PRASENJIT MITRA,

C.LEE GILES. Researcher Homepage Classification Using Unlabeled Data. WWW 2013.