A critical review for URL-based web page classification proposals

Binbin He

Lei Cao

Abstract

There are many interesting applications of web page classification, which refers to classify a web page a class where web pages have same contents. With regard to one of the applications, enterprise web information integration, URL-based web page classification proposals are relatively appropriate. This article presents eight Web page classifiers building on URLs and compares those classifiers from three aspects: (i) lightweight crawling, (ii) unsupervision, and (iii) classify without downloading. By analyzing those classifiers, we found that that none of the proposals in the article fulfills the three requirements which have been identified before. An unsupervised proposal which performs a lightweight crawling and classify web page without downloading previously is anticipated.

Key words: Web page classification, URL classification

Introduction

With the continuous growth in the scale and use of the Internet, it has attracted many scholars to do research to develop the value of web. Web page classification is one of the essential techniques of web data mining, because classifying Web pages of an interesting class is often the first step of mining the Web pages.

There are a variety of techniques to learn web page classifiers; they can be broadly classified into the following categories: URL-based proposals [1,3,4,8], Contents-based proposals [7], visual-based proposals [14], structure-based proposals [2,4], and link-based proposals [11]. The proposals in the first category rely on features of the URLs, which, in principle, contain potential information of web pages, and are relatively easier to realize the classification of the web page, comparing to analyze a whole web page.

Scope and method

This article mainly focuses on the URL-based proposals. In order to have a better knowledge of the related work of this topic, we searched the subject web page classification, URL classification on the Internet, and find a number of journal and conference papers with high quality.

In the process of analyzing these proposals, we found that there are many techniques to learn a web page classifier. A subset of web pages are required downloading for most techniques, which is commonly referred to as training set, and analyses some of their features. These features can be external to the pages such as features in their URLs or the context, or internal such as words or HTML structure.

Some techniques are supervised, i.e., a person is required to pre-classify the pages in the training set, while others are unsupervised, i.e., their results must be interpreted by a person.

In order to measure whether a learning or classification technique is useful in the context of enterprise web information integration, it must fulfill three requirements:

1. **Lightweight crawling:** It is essential to provide a training dataset for learning a classifier, which is gathered by performing a crawl of the site being analyzed. Such a crawl should be as lightweight as possible since it should not interfere with the normal operation of the site. Furthermore, web sites change frequently [7] and it is not uncommon that these changes render the classifier obsolete. Therefore, the classifier must be learnt not once but several times and performing an extensive crawling that covers a large portion of a web site becomes unfeasible [9].
2. **Unsupervision:** It is important that a person does not need to pre-classify each page in the training set individually, for this is an effort-consuming and error-prone task; neither should he or she provide any additional training information. In other words, it should be seriously considered that the techniques used to learn a classifier is unsupervised or, otherwise, they shall not scale well to the Web [12].
3. **Classify without downloading:** It would increase load on the server to takes time to download a web page, and consumes bandwidth. That is, it is important that a classifier relies exclusively on external features of a page in

order to classify it, since it would be inefficient for practical purposes otherwise [8]

Body of the review

In this section, a number of proposals of URL-based web page classifiers will be analyzed. Firstly, a naive and basic approach to classify web pages is to be introduced; This approach rely on a distance function and return a number of classifications that verify that the inter-distance is maximum, whereas the intra-distance is minimum. String distance would be used in the distance function, because the URLs can be naturally represented as strings. Unfortunately, it has been noticed that using classic string distances does not work well to classify URLs [4] since it may happen that two close URLs may provide information about two different classes, whereas distant URLs may be related to web pages of the same class. On the contrary, URLs are far more distant but belong to the same class. It remains unexplored whether using non-classic distances might improve the results.

To improve the previous naive approach, eight kind of techniques are proposed by other authors that are summarized in Table 1. In the following paragraphs of this part, additional details on each proposal will be provided.

Kan and Thi [8] represented a supervised technique to classify web pages building exclusively on URLs' features. The URLs are tokenized by using the standard RFC 3985 format for URIs, and those tokens and their positions are used as features; Besides, the lexical kind of token (e.g., if it represents a word, a number, or a non-alphabetical symbol) and the length of the URL are computed out as features.

Next, these features are used as input to an entropy maximization algorithm, a well-known machine learning approach that is usually applied to text classification [9, 11]. In order to build the classifier, large training sets of URLs are used to achieve good precision and recall. As a result, this method requires a previous extensive crawling of the sites that are being analyzed.

Brin [14] proposed the DIPRE, a supervised approach to extract structured information from web pages. The Web pages are considers as a database of unstructured information, and can be extracting tuples (e.g., glasses). The input of this proposal is a set of sample tuples. The approach consisting of the following steps performs an incremental process: first of all, it looks for occurrences of the sample tuples in the Web, i.e., it looks for web pages where the attributes of one of the tuples occur near to each other. For each occurrence, the URL of the web page on which it appears and the text that surrounds it are considered the context of the occurrence. Afterwards, these contexts can be used to generate patterns that match occurrences with a similar context. The same URL prefix is included in these patterns, which is the longest common prefix to the URLs of the occurrences, and a text pattern, which is a regular expression that matches the text surrounding the occurrences. Finally, it looks for tuples matching the new patterns in the Web. The iterative process until enough patterns being generated. It should be noted that DIPRE requires to downloading extensive Web pages to gather as many tuples of the target relation as possible

Baykan et al. [3, 2] proposed a supervised web page classification that exclusively based on URLs. First, the URLs are tokenized as feature to create feature

Table 1

Comparison of current URL-based proposals. (R1 = Lightweight crawling; R2= Unsupervision; R3 = Classify without downloading)

Proposals	R1	R2	R3
Brin [14]	No	No	Yes
Kan and Thi [8]	No	No	Yes
Vidal et al. [13]	No	No	Yes
Bar-Yossef et al. [1]	No	Yes	Yes
Baykan et al. [3]	No	No	Yes
Koppula et al. [10]	No	Yes	Yes
Baykan et al. [2]	No	No	Yes
Blanco et al. [4]	No	Yes	No

vectors. Then they build a support vector machine and a naive-bays classifier based on those features. In their experiments, they use extensive training sets of URLs, and they require the user to provide a list of words and URLs which should be representative of every class; Furthermore, they also require a sample set of URLs that are not representative of each class.

Vidal et al. [13] proposed a supervised web pages classifier building on their URL. Their proposal takes a sample page as input, and returns a set of URL patterns that match the URLs of Pages that are structurally similar to the sample page. It based on following two steps: site mapping and pattern generation. Site mapping consists in building a map of the web site, which requires to crawl the entire site starting from its home page and following every possible path. They keep a record of the paths in the

map that lead (directly or indirectly) to pages that are similar to the sample page. The similarity is measured using a tree-edit distance between the DOM trees underlying the pages. Then, pattern generation consists of generalizing the URLs of the pages in the former paths using regular expressions, and then selecting the path that leads to the largest number of target pages.

Bar-Yossef et al. [1] proposed a supervised technique to detect web pages with different URLs that have the same contents. Crawling efficiency would suffer a negative impact from this. To solve this problem, it classifies that URLs according to the contents of their target, and regular expressions is built to define each class of URLs. Furthermore, those URLs are normalized using a rule mining algorithm. A large collection of URLs is required to achieve good results, which can be inferred that a previous extensive crawling of the web site must be performed to gather them. Besides, Koppula et al. presented a similar proposal [10].

Blanco et al. [4] proposed an unsupervised algorithm to classify web pages which is combining external features and optional internal features. Their proposal can be summarized into the idea that every web site is created by populating a number of HTML templates with data from a database, and that the URLs of those pages are created by populating a URL template with data from the same database. Therefore, pages created from the same HTML template have similar contents and URLs generated from the same URL template link to pages with similar contents. An algorithm is proposed that combines web page contents and its URL as features to cluster web pages so that each cluster contains pages that were created using a certain

template. Their algorithm is based on the well-known minimum description length method [7]. In order to crawl the entire site in their experiments, they require a large training set. It should be mentioned that for improving the classification efficiency, internal features is essential part, which means that the page must be downloaded previously in some cases.

Interpretation and conclusion

From the previous paragraphs, a set of web pages classifiers based on URLs has been analyzed. It can be inferred that none of the techniques in the review body fulfills the three requirements, although, most of them have advantages, There is a kind of proposals expected, which are not require the quantity of web site which need to be crawled, but only a small set of hubs. Besides, it is completely unsupervised, and can classify a page without downloading it before.

Words count: 1,855

References

[1] Ziv Bar-Yossef, Idit Keidar, Uri Schonfeld, Do not crawl in the DUST: Different URLs with similar text, TWEB 3 (1) (2009) 3, http://dx.doi.org/10.1145/1462148.1462151

[2] Eda Baykan, Monika Henzinger, Ludmila Marian, Ingmar Weber, A comprehensive study of features and algorithms for URL-based topic classification, TWEB 5 (3) (2011) 15, http://dx.doi.org/10.1145/1993053.1993057

[3] Eda Baykan, Monika Rauch Henzinger, Ludmila Marian, Ingmar Weber, Purely URL-based topic classification, in: WWW, 2009, pp. 1109–1110, http://dx.doi.org/10.1145/1136709.1136880.

[4] Lorenzo Blanco, Nilesh Dalvi, Ashwin Machanavajjhala, Highly efficient algorithms for structural clustering of large websites, in: WWW, 2011, pp. 437–446, http://dx.doi.org/10.1145/1961005.1961068.

[5] Sergey Brin, Extracting patterns and relations from the World Wide Web, in: WebDB, 1998, pp. 172–183, http://dx.doi.org/10.1007/10704656_11.

[6] Dennis Fetterly, Mark Manasse, Marc Najork, Janet L. Wiener, A large-scale study of the evolution of web pages, Softw. Pract. Exper. 10 (2) (2004) 213–237, http://dx.doi.org/10.1002/spe.577.

[7] Peter Grünwald, Advances in Minimum Description Length: Theory and Applications, MIT Press, 2005.

[8] Min-Yen Kan, Hoang Oanh Nguyen Thi, Fast web page classification using URL features, in: CIKM, 2005, pp. 325–326, http://dx.doi.org/10.1145/1099554.1099649.

[9] Wallace Koehler, An analysis of web page and web site constancy and permanence, JASIS 50 (2) (1999) 162–180. doi: 10.1002/(SICI)1097-881 4571(1999) 50:2<162::AID-ASI7>3.0.CO; 2-B.

[10] Hema Swetha Koppula, Krishna P. Leela, Amit Agarwal, Krishna Prasad Chitrapura, Sachin Garg, Amit Sasturkar, Learning URL patterns for webpage de-duplication, in: WSDM, 2010, pp. 381–390, http://dx.doi.org/10.1145/1718487.1718535.

[11] Kamal Nigam, Andrew Kachites Mccallum, Sebastian Thrun, Tom Mitchell, Text classification from labeled and unlabeled documents using EM, Mach.Learn. 39 (2–3) (1999) 103–110,http://dx.doi.org/10.1023/A:1007692713085.

[12] Hassan A. Sleiman, Rafael Corchuelo, Trinity: on using trinary trees for unsupervised web data extraction, IEEE Trans. Knowl. Data Eng. (99) (2013), http://dx.doi.org/10.1109/TKDE.2013.161. 1-1, Preprint, ISSN 1041-4107.

[13] Márcio L.A. Vidal, Altigran Soares da Silva, Edleno Silva de Moura, João M.B.Cavalcanti, Structure-based crawling in the Hidden Web, J. UCS 14 (11) (2008) 1857–1876, http://dx.doi.org/10.3217/jucs-014-11-1857.

[14] Jianping Zhang, Jason Qin, Qiuming Yan, The role of URLs in objectionable web content categorization, in: Web Intelligence, 2006, pp. 277–283, http://dx.doi.org/10.1109/WI.2006.170.