A Comparative Study of Web Page Classification Techniques

Mr. Anirudhdha Nayak

(anirudhdha.nayak@git.org.in)

Abstract— Internet provides millions of web pages for each and every search term and it is a powerful medium for communication between computers and accessing online documents but tools like search engines assist users in locating and organizing information. Classification is a data mining technique used to predict group membership for data instances. In Web classification, web pages are assigned to pre-defined categories mainly according to their content (content mining). In this paper, we present the some basic classification techniques like decision trees, k-nearest neighbor, naïve bayes and support vector machine. Web page classification is one of the essential techniques for Web mining because classifying Web pages of an interesting class is often the first step of mining the Web. The goal of this paper is to provide a comprehensive review of different classification techniques.

Index Terms— Data Mining, Web Mining, Web page classification, Classification Techniques.

I. Introduction

The growth of information sources available on the World Wide Web has made it necessary for users to utilize automated tools in finding the desired information resources. There is a necessity of creating intelligent systems for servers and clients that can effectively mine for knowledge. Web mining can be broadly defined as the discovery and analysis of useful information from the World Wide Web by the extraction of interesting and potentially useful patterns and implicit information from artifacts or activity related to the World Wide Web.

There are roughly three knowledge discovery domains that pertain to web mining: Web Content Mining, Web Structure Mining, and Web Usage Mining. Web content mining is the process of extracting knowledge from the content of documents or their descriptions. Web structure mining is the process of inferring knowledge from the World Wide Web organization and links between references and referents in the Web. Finally, web usage mining is the process of extracting interesting patterns in web access logs [1].

Web content mining is an automatic process that goes beyond keyword extraction. Since the content of a text document presents no machine readable semantic, some approaches have suggested restructuring the document content in a representation that could be exploited by machines. The usual approach to exploit known structure in documents is to use techniques to map documents to some data model. There are two web content mining strategies: those that directly mine the content of documents and those that improve on the content search of other tools. From a Data Mining point of view, Web mining, has three areas of interest: clustering (finding natural groupings of users, pages etc.), associations (which URLs tend to be requested together), and classification (characterization of documents).

Web page classification is the process of assigning a Web page to one or more predefined category labels [2]. Classification can be understood as a supervised learning problem in which a set of labelled 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 is concerned about the subject or topic of a Web page. Functional classification cares about the role that the Web page plays. Sentiment classification focuses on the opinion that is presented in a Web page, that is, the author’s attitude about some particular topic. Other types of classification include genre classification, search engine spam classification. This paper focuses on subject and functional classification.

II. The Web Data: Features Of Web Pages

Web pages are what make up the World Wide Web. A Web page is a document or information resource written usually in HTML (hypertext markup language) and translated by your Web browser. Web pages are formed by a variety of information, such as: images, videos or other digital assets that are addressed by a common URL (Uniform Resource Locator). These pages are typically written in scripting languages such as PHP, Perl, ASP, or JSP. The scripts in the pages run functions on the server that return things like the date and time, and database information. All the information is returned as HTML code, so when the page gets to your browser, all the browser has to do is translate the HTML, interpreting it and displaying it on the computer screen. Since all pages share the same language and elements it is possible to characterize each of them accurately in an automatic way. All web pages are different, in order to classify them, data extracted from the HTML code will be used. Pages can then be classified according to multiple categories. Throughout this document, the following labels have been used: blog, video, images and news.

- Blogs: short for weblog is a personal online journal with reflections, comments provided by the writer. Blogs are frequently updated and intended for general public consumption. Blogs generally represent the personality of the author or reflect the purpose of the Web site that hosts the blog. Blogs can be distinguished by their structure: a series of entries posted to a single page in reverse-chronological order. Blogs contents are basically text, occasionally images and videos are included.
- Video: these web pages provide a venue for sharing videos among friends and family as well as a showcase for new and experienced videographers. Videos are streamed to users on the web site or via blogs and other Web sites. Specific codes for playing each video are embedded on the Web page.
- Image: a photo gallery on a website is collection of images or photos that is uploaded to a website and available for website visitors to view. These web pages hardly have any text and generally carry a navigator that allows the visitor to move around the images.
- News: an online newspaper is a web site which provides news on a basis which is close to real time. A good news site updates its content every few minutes.

III. Classification Techniques

Classification means given a collection of records called training set, and each record contains a set of attributes, one of the attributes is the class. The technique is to find a model for class attribute as a function of the values of other attributes. It will assign a class as accurately as possible from the previously unknown records. The several classification techniques, summarized as follows:

A. *Decision Trees*

Decision trees are trees that classify instances by sorting them based on feature values. Each node in a decision tree represents a feature in an instance to be classified, and each branch represents a value that the node can assume. Instances are classified starting at the root node and sorted based on their feature values.

The basic algorithm for decision tree induction is a greedy algorithm that constructs decision trees in a top-down recursive divide-and-conquer manner.

The algorithm, summarized as follows:

1. create a node N;
2. if samples are all of the same class, C then
3. return N as a leaf node labeled with the class C;
4. if attribute-list is empty then
5. return N as a leaf node labeled with the most common class in samples;
6. select test-attribute, the attribute among attribute-list with the highest information gain;
7. label node N with test-attribute;
8. for each known value ai of test-attribute
9. grow a branch from node N for the condition test-attribute= ai;
10. let si be the set of samples for which test-attribute= ai;
11. if si is empty then,
12. attach a leaf labeled with the most common class in samples;
13. else attach the node returned by Generate_decision_tree.

Decision trees are usually unvaried since they use based on a single feature at each internal node. Most decision tree algorithms cannot perform well with problems that require diagonal partitioning. Decision trees can be significantly more complex representation for some concepts due to the replication problem. A solution is using an algorithm to implement complex features at nodes in order to avoid replication.

To sum up, one of the most useful characteristics of decision trees is their comprehensibility. People can easily understand why a decision tree classifies an instance as belonging to a specific class. Since a decision tree constitutes a hierarchy of tests, an unknown feature value during classification is usually dealt with by passing the example down all branches of the node where the unknown feature value was detected, and each branch outputs a class distribution. The output is a combination of the different class distributions that sum to 1. The assumption made in the decision trees is that instances belonging to different classes have different values in at least one of their features. Decision trees tend to perform better when dealing with discrete/categorical features.

Wen-Chen Hu [3] has performed a modified decision trees for web page classification, which facilitates web page search. By choosing keywords and descriptions for the web pages and keywords for web categories, the level of accuracy can be improved significantly. V. Estruch et. al. [4] have proposed web categorization using distance-based decision trees. It can be seen as a general and flexible web categorization framework which is potentially able to deal with any kind of information (belonging to the content, the structure or the usage of the Web pages) in a uniform way.

B. *K-Nearest Neighbor*

Nearest neighbor classifiers are based on learning by analogy. The training samples are described by n dimensional numeric attributes. Each sample represents a point in an n-dimensional space. In this way, all of the training samples are stored in an n-dimensional pattern space. When given an unknown sample, a k-nearest neighbor classifier searches the pattern space for the k training samples that are closest to the unknown sample. "Closeness" is defined in terms of Euclidean distance, where the Euclidean distance, where the Euclidean distance between two points, X=(x1,x2,……,xn) and Y=(y1,y2,….,yn) is

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (1)$$

The unknown sample is assigned the most common class among its k nearest neighbors. When k=1, the unknown sample is assigned the class of the training sample that is closest to it in pattern space.

The *k-nearest neighbors' algorithm* is amongst the simplest of all machine learning algorithms. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its *k* nearest neighbors. *k* is a positive integer, typically small. If *k* = 1, then the object is simply assigned to the class of its nearest neighbor. In binary (two class) classification problems, it is helpful to choose *k* to be an odd number as this avoids tied votes.

The neighbors are taken from a set of objects for which the correct classification is known. This can be thought of as the training set for the algorithm, though no explicit training step is required. In order to identify neighbors, the objects are represented by position vectors in a multidimensional feature space. It is usual to use the Euclidian distance, though other distance measures, such as the Manhattan distance could in principle be used instead. The *k*-nearest neighbor algorithm is sensitive to the local structure of the data.

Juan Zhang et. al. [5] has proposed web document classification based on fuzzy k-NN algorithm to increase the accuracy in the classification results. They used TF/IDF for select features of a document. Performance was better than k-NN and SVM but speed was bit slow than k-NN.

C. *Support Vector Machine*

The Support Vector Machine (SVM) is a classification technique based on statistical learning theory that was applied with great success in many challenging non-linear classification problems and on large data sets. Support Vector Machines (SVM) is a classification technique for finding a decision boundary that separates the labeled data set with the widest margin. For simplicity, the assumption is that the data set is composed of data points and each data point is made up of a set of attribute-value pairs. Furthermore, each data point is assigned a label for the purpose of training the SVM. We distinguish two cases of data set in this section.

SVM was first bought forward by Cortes and Vapnik [6] as a learning algorithm for classification and regression. It tried to maximize the margin of confidence of classification on the training data set, which could use the linear, polynomial or radial basis function (RBF) kernels. In the case of support vector machine, an object is viewed as n-dimensional vector and we want to separate such objects with n-1 dimensional hyper-plane. This is called a linear classifier. There are many hyper-planes that might classify the data. The goal of SVM is try to address the nearest distance between a point in one class and a point in the other class being maximized and draw a hyper-plane to classify two categories as clearly as possible.

D. *Naïve Bayes*

The Naïve Bayes classifiers (Lewis 1992) are known as a simple Bayesian classification algorithm. Under the Bayes independent assumption premise, it simulates each kind of class condition joint probability distribution separately, and then constructs posterior classifier with Bayes theorem. Regarding web document categorization problem, a document $d \in D$ corresponds to a data instance, where *D* denotes the training document set. Each document *d* is associated with a class label $c \in C$, where C denotes the class label set [7].

A Bayesian Network (BN) is a graphical model for probability relationships among a set of variables features. The Bayesian network structure *S* is a directed acyclic graph (DAG) and the nodes in *S* are in one-to-one correspondence with the features *X*. Typically, the task of learning a Bayesian network can be divided into two subtasks: initially, the learning of the directed cyclic graph (DAG) structure of the network, and then the determination of its parameters. Probabilistic parameters are encoded into a set of tables, one for each variable, in the form of local conditional distributions of a variable given its parents. Given the independences encoded into the network, the joint distribution can be reconstructed by simply multiplying these tables.

The methods for learning Bayesian belief network (BN) based predictive models for classification was investigated by Jie Cheng and Russell Greiner [8]. They studied two types of unrestricted BN classifiers – general Bayesian networks and Bayesian multi-nets. The results show that BN learning algorithms are very efficient, and the learned BN classifiers can give really good prediction accuracy. By checking and modifying the learned BN predictive models, domain experts can study the relationships among the attributes and construct better BN predictive models. The most interesting feature of BNs, compared to decision trees or neural networks, is most certainly the possibility of taking into account prior information about a given problem, in terms of structural relationships among its features. This prior expertise, or domain knowledge, about the structure of a Bayesian network can take the following forms:

1. Declaring that a node is a root node, i.e., it has no parents.
2. Declaring that a node is a leaf node, i.e., it has no children.
3. Declaring that a node is a direct cause or direct effect of another node.
4. Declaring that a node is not directly connected to another node.
5. Declaring that two nodes are independent, given a condition-set.
6. Providing partial nodes ordering, that is, declare that a node appears earlier than another node in the ordering.
7. Providing a complete node ordering.

IV. Evaluations Measures

A. *Precision & Recall*

For classification tasks, the terms **true positives**, **true negatives**, **false positives**, and **false negatives** compare the results of the classifier under test with trusted external judgments. The terms *positive* and *negative* refer to the classifier's prediction (sometimes known as the *observation*), and the terms *true* and *false* refer to whether that prediction corresponds to the external judgment (sometimes known as the *expectation*).

B. *F-measure*

A measure that combines precision and recall is the harmonic mean of precision and recall, the traditional F-measure or balanced F-score. This is also known as the F_1 measure, because recall and precision are evenly weighted.

V. Conclusion

We have surveyed some existing classification techniques. The goal of classification result integration algorithms is to generate more certain, precise and accurate system results. Decision trees and Bayesian Network generally have different operational profiles, when one is very accurate the other is not and vice versa. Classification methods are typically strong in modeling interactions. However, a straightforward application of classification methods to large numbers of markers has a potential risk picking up randomly associated markers. Although or perhaps because many methods of web page classification have been proposed, there is yet no clear picture of which method is best.

References

[1] Dunham M. H., “Data Mining: Introductory and Advanced Topics”, Prentic Hall, New Jersey, 2003.

[2] Qi X and Davison B.D., “Web Page Classification: Features and Algorithms ACM Computing Surveys”, Vol. 41, No. 2, Article 12, 2009.

[3] Wen-Chen Hu, “WebClass: Web document classification using modified decision- trees”, The Fifth International Conference on Computer Science and Informatics, September 1999.

[4] V. Estruch, C. Ferri, J. Hernandez-Orallo and M.J. Ramirez-Quintana, “Web Categorisation Using Distance-Based Decision Trees”, Published by Elsevier Science B. V., 2005.

[5] Juan Zhang, Yi Niu and Huabei Nie, “Web Document Classification Based on Fuzzy k-NN Algorithm”, International Conference on Computational Intelligence and Security, pp. 193-196, 2009.

[6] C. Cortes and V. Vapnik, “Support Vector Networks,” Machine Learning, vol. 30 No. 3, pp. 273-297, 1995.

[7] Jensen, F., “An Introduction to Bayesian Networks”, Springer, 1996.

[8] Cheng, J. & Greiner, R., “Learning Bayesian Belief Network Classifiers: Algorithms and System”, In Stroulia, E. & Matwin, S. (ed.), *AI 2001*, 141-151, LNAI 2056, 2001.

[9] Bernhard E. B., Isabelle M. G., Vladimir N. V., “A Training Algorithm for Optimal Margin Classifiers”, In Proceedings of International Conference on Computational Learning Theory, pp. 144-152, 1992.

[10] E. Glover, K. Tsioutsiouliklis, S. Lawrence, D. Pennock, and G. Flake, “Using web structure for classifying and describing web pages”, In Proc. of the WWW2002, Hawaii, USA, May 2002.