An Intelligent Web-Page Classifier with Fair Feature-Subset Selection

Hahn-Ming Lee, Chih-Ming Chen, and Chia-Chen Tan

Department of Electronic Engineering,
National Taiwan University of Science and Technology
Taipei , Taiwan
hmlee@et.ntust.edu.tw

Abstract

The explosion of on-line information has given rise to many manually constructed topic hierarchies (such as Yahoo!!). But with the current growth rate in the amount of information, manual classification in topic hierarchies results in an immense information bottleneck. Therefore, developing an automatic classifier is an urgent need. However, the classifiers suffer from the enormous dimensionality, since the dimensionality is determined by the number of distinct keywords in a document corpus. More seriously, most classifiers are either working slowly or they are constructed subjectively without learning ability. In this paper, we address these problems with a fair feature subset selection algorithm and an adaptive fuzzy learning network (AFLN) for classification. The fair feature subset selection algorithm is used to reduce the enormous dimensionality. It not only gives fair treatment to each category but also has ability to identify useful features, including both positive and negative features. On the other hand, the AFLN provides extremely fast training and testing and, more importantly, it has the ability to learn the human knowledge. Experimental results show that our proposed fair feature subset selection algorithm is effective in recognizing useful keywords for classification. It indeed can be used to reduce a surprising number of dimensions in classification models. Besides, experimental results also show the adaptive fuzzy learning network for classification with high-speed classification and high accuracy rate.

1. Introduction

In recent years, information is grown rapidly, especially on the World Wide Web [1]. It is estimated that the web now contains more than twenty million different content areas, presented on more than 320 million web pages, and one million web servers, and it is double every nine months [2]. Many researchers have devoted themselves to the study of developing tools, such as search engines [3], net directories [4] and metasearch engines [5], aiming at helping users to acquire their desired documents.

Net directories are useful and suitable tools for browsing the web pages on WWW, especially in those situations mentioned above, so more and more search tools combine the functions of search engines with net directories. To fulfill this task, a traditional method is to classify documents into the existing topic structure manually every day. However, it is a time-consuming job and is almost impossible to keep up with the amount of new documents. Therefore, developing an automatic classifier is essential.

Moreover, many researchers pay attention to design lots of mathematical functions to extract human knowledge for the importance of features. For example, one popular scheme, known as TF×IDF [6], assigns a weight to each feature in a particular category to indicate that how frequently a feature term occurs in the entire document corpus and how often a feature term appears in the category. However, a difficulty arises in such knowledge acquisition is that the human knowledge cannot be completely expressed as criteria in mathematics. Therefore, in order to acquire human knowledge about importance of features as possible as we can for classification, an adaptive fuzzy learning network (AFLN) is presented to perform classification task with learning ability. Furthermore, a fair feature subset selection algorithm is proposed to select useful keywords to serve as used features in document classification system. Experimental results show that our proposed fair feature subset selection algorithm can be used to reduce a surprising number of dimensions for keyword extraction. Besides, the adaptive fuzzy learning network has high-speed classification and high accuracy rate.

2. System Architecture

In what follows, we will describe all procedures of our proposed system architecture in detail.

2.1 Keywords Extraction Stage

In order to process documents into a vector representation with reduced dimensionality, three processing components are essential to generate keywords in the keyword extraction stage: HTML parser, word extractor and keyword controller. The progress of this stage is shown in Figure 1.

2.2 Fair Feature Subset Selection Algorithm

In this work, fair feature subset selection algorithm involves three stages as follows:

<u>1. Local Scoring</u>

We apply two local scoring functions, keyword frequency and document frequency, to measure the

0-7803-7078-3/01/$10.00 (C)2001 IEEE.

degree of representative of a keyword in some category.

a. Keyword Frequency (KF) Function

For the keyword i, the objective of this function is to compute the average occurrence frequency of this keyword per document for documents in each category and it can be written as follows:

$$l_i^k = \frac{\sum_{j=1}^{n_k} kf_i^j}{n_k}, \qquad (1)$$

where l_i^k is the local score of keyword i in category k, kf_i^j is the occurrence frequency of keyword i in the jth document and n_k is the number of training documents constituting the category k.

b. Document Frequency (DF) Function

It measures the frequency of documents containing a specific keyword in a given category. Namely, the document frequency for keyword i in category k can be written as

$$l_i^k = \frac{\sum_{\substack{j=1,\ldots,n_k; \\ kf_i^j \neq 0}} 1}{n_k}. \qquad (2)$$

where l_i^k is the local score of keyword i in category k, kf_i^j is the occurrence frequency of keyword i in the jth document and n_k is the number of training documents constituting the category k.

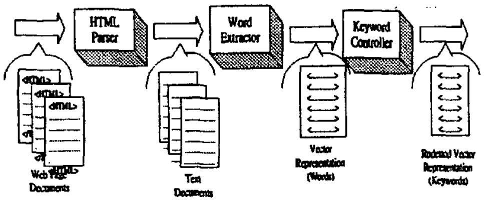

Figure 1. Keyword extraction stage

2. Normalization

The local scores computed by the local scoring function are the proper indicator acting as the degrees of representative of keywords. However, it is inappropriate since these local scores are valid only in their corresponding categories and thus they cannot be compared to each other directly. Thus, we present a normalization process to convert the local scores into global stamp values to make them valid everywhere without losing the ability of expressing the degree of representative.

Consider converting the local score of keyword i in category k, l_i^k, into its stamp value, s_i^k. Within category k, we can first rank keywords with non-zero local scores according to their corresponding local scores in the category. Then, based on the rank of position in the ranking list, we assign stamp values to keywords. These operations can be expressed as follows:

$$s_i^k = \begin{cases} 0 \ ; \ r_i^k = 0 \\ \left(r_{\max}^k + 1 - r_i^k\right) \times \frac{s_{\max}}{r_{\max}^k} \ ; \ r_i^k \neq 0 \end{cases} ; \quad i = 1, \cdots, p \qquad (3)$$

where s_i^k is the stamp value of keyword i in category k, $s_{\max}$ is predefined largest stamp values, r_i^k is the rank of position of keyword i in category k, $r_{\max}^k$ is the largest rank position within category k, and r_i^k and $r_{\max}^k$ can be computed by Eqs. (4) and (5) respectively as

$$r_{\max}^k = \max_{i=1}^{p}\left(r_i^k\right) \qquad (4)$$

$$r_i^k = \begin{cases} 0 \ ; \ l_i^k = 0 \\ \textit{the rank of position in the list} \ ; \ l_i^k \neq 0 \end{cases}, \qquad (5)$$

where l_i^k is the local score of keyword i in category k and p is the number of keywords identified in keyword extraction stage. To summarize, using Eqs.(3)-(5), if a keyword is more representative in some category, a higher local score and larger stamp value would be assigned to it.

3. Global Scoring

The local scoring function cannot provide complete information to perform fair feature subset selection because the contribution among categories for a given keyword is ignored. To handle it, the function for keyword i is expressed as

$$G_i = \sum_{k=1}^{q}\left[\max_{j=1}^{q}\left\|s_i^k - s_i^j\right\|\right]. \qquad (6)$$

where G_i is the global score of keyword i, q is the number of predefined categories in training documents and s_i^k, computed by Eq. (3), is the stamp value of keyword i in category k. In this equation, for each given category, we first compute maximum difference between this category and other categories made by keyword i. Then we sum up all theses differences to estimate the discriminating power contributed by this keyword so as to reveal the usefulness of the keyword in classification.

In summary, our proposed fair feature subset selection consists of applying two essential scoring functions (i.e., local and global scoring functions) and a normalization procedure. By these procedures, we can obtain a good feature selection results because our approach can make each category being treated equally in feature selection process.

2.3 Weight Assignment Stage

In this work, we use the occurrence frequency of features to indicate the representative of features in a given document. It suggests that if a keyword appears in a particular document frequently, then it is highly representative in this document. Thus, in this stage, we simply compute the weight of feature i in each document vector d as

$$d_i = kf_i, \tag{7}$$

where kf_i represents the occurrence frequency of feature i in the document d and d_i is the weight of feature i in document d.

2.4 Adaptive Fuzzy Learning Network (AFLN)

The algorithm used to construct the AFLN consists of two separate phases, i.e., membership function setting and parameter learning phases. In membership function setting phase, we determine all initial membership functions used in AFLN. Then, in parameter learning phase, in order to enhance the accuracy rate of AFLN, we train the membership functions by means of optimally adjusting their parameters for desired outputs.

2.4.1 AFLN Structure

In adaptive fuzzy learning network (AFLN), each document and the kth category are represented by corresponding vectors, $\boldsymbol{d}$ and $\boldsymbol{c}^k$, as follows:

$\boldsymbol{d} = (d_1, d_2, \cdots, d_p)$

and $\boldsymbol{c}^k = (c_1^k, c_2^k, \cdots, c_p^k)$,

where p is the number of features used in classification, d_i, computed by Eq.(7), is the weight of the ith dimension in $\boldsymbol{d}$ to indicate the representative degree of feature i in the document. c_i^k, calculated using Eq.(11), is the weight of the ith dimension in $\boldsymbol{c}^k$ to reveal the representative degree of feature i in the kth category. Consider classification of an input document vector $\boldsymbol{d}$ into q predefined categories, the proposed network structure can be drawn as Figure 2.

In Figure 2, the AFLN has a total of four layers. The nodes in layer 1 are input nodes that represent input linguistic variables and transmit the input document vector to the next layer directly. Each node in layer 2 acts as a membership function to measure the degree of similarity between the input document and a particular category in some dimension. Thus, in the hope that $\mu_{M_i^k}(d_i)$ increases, d_i is closer to c_i^k. A bell-shaped (Gaussian) function is adopted to express the membership function, and this function for the ith node can be expressed as

$$\mu_{M_i^k}(d_i) = \begin{cases} h_i^k \times \exp\left[-\left(\dfrac{d_i - c_i^k}{\sigma_i^k} \right)^2 \right] & ; \quad d_i > 0 \\ 0 & ; \quad else \end{cases} \tag{8}$$

where $\mu_{M_i^k}(d_i)$ indicates the similarity degree between $\boldsymbol{d}$ and h_i^k in the ith dimension, c_i^k, σ_i^k and h_i^k are, respectively, the center, width and height of this bell-shaped function.

Furthermore, nodes in layer 3 are aggregation nodes to accumulate a set of similarity degrees to draw a conclusion. For example, the kth node in layer 3 sums those similarity degree between $\boldsymbol{d}$ and $\boldsymbol{c}^k$ in individual dimensions to induce an output as

$$y_k = \sum_{i=1}^{p} \mu_{M_i^k}(d_i), \tag{9}$$

where p is the number of features, $\mu_{M_i^k}(d_i)$ is used to measure the degree of similarity between $\boldsymbol{d}$ and $\boldsymbol{c}^k$ in the ith dimension and y_k is the output of the kth node in layer 3 indicating the degree that the document $\boldsymbol{d}$ belongs to the kth category.

Finally, these outputs in layer 4 are synthesized in layer 5 to yield an output vector y* as

$$\boldsymbol{y}^* = (y_1, y_2, \ldots, y_q), \tag{10}$$

which indicates the degrees that the input document $\boldsymbol{d}$ belong to corresponding categories. As a result, we are able to classify each unseen document into proper categories based on this vector.

2.4.2 Membership Function Setting

In this stage, we seek to set each membership function in layer 2. From Eq.(8), we know that, for each $\mu_{M_i^k}(d_i)$, there are three parameters, h_i^k, c_i^k and σ_i^k, to determine. First, c_i^k implies the weight of feature i in the kth category vector, and it is computed as the average of the non-zero weight of feature i in those document vectors belonging to the kth category. That is,

$$c_i^k = \frac{1}{N_i^k} \sum_{d \in category\ k} d_i \tag{11}$$

where d_i is the weight of feature i in $\boldsymbol{d}$ and N_i^k is the number of document vectors constituting category k in which the ith weight is not zero. Of course, the value of c_i^k computed by Eq.(11) might not be appropriate. As a result, it will be further tuned in fuzzy rule training phase.

Moreover, h_i^k is used to model the importance of features in each category that can be extracted by mathematic method and is computed as

$$h_i^k = s_i^k \times \frac{s_i^k}{s_i} \qquad (12)$$

$$s_i = \sum_{k=1}^{q} s_i^k \qquad (13)$$

where q is the number of predefined categories and s_i^k, computed by Eqs.(3)-(5), is the stamp value of feature i in the kth category. The idea of Eqs. (12) and (13) is similar to TF×IDF [6] as: the value of h_i^k increases the more representative the feature i is in category k and decreases the more representative the feature i is in all other categories.

Furthermore, since not all of human knowledge can be expressed as criteria in mathematics and extracted by h_i^k, σ_i^k is designed to acquire human knowledge as to the importance of features to improve the accuracy of the classification system in parameter leaning phase. Therefore, in this phase, σ_i^k is assigned a temporary constant directly and then learned in parameter learning phase.

2.4.3 Parameter Learning

After the membership functions have been generated, the AFLN network structure is established. The network then enters the second learning phase to adjust the parameters of the membership functions. Whereby such learning, each category vector can be adjusted optimally as c_i^k increases and decreases. Besides, the human knowledge also can be easily incorporated in AFLN as well by the adjusting each σ_i^k. This is due the fact that an increase of σ_i^k decreases the slop of the membership function $\mu_{M_i^k}(d_i)$ or increases the value of $\mu_{M_i^k}(d_i)$ and thus the degree of importance of feature i in category k decreases.

Let us now derive the parameter learning algorithm. Assume that the category o is the desired output for some input training document $\boldsymbol{d}$. To begin with, we define a performance function to be maximized as

$$E = y_o . \qquad (14)$$

where y_o is the output of the o th node in layer 3 as shown in Figure 2.

By substituting Eqs. (7) and (9) into Eq. (14), we have

$$E = \sum_{i=1}^{p} \mu_{M_i^o}(d_i) = \sum_{i=1}^{p} \exp\left[-\left(\frac{d_i - c_i^o}{\sigma_i^o}\right)^2\right]. \qquad (15)$$

According to the gradient descent method, the c_i^o and σ_i^o can be updated by

$$\Delta c_i^o = \eta \frac{\partial E}{\partial c_i^o} \qquad (16)$$

$$\Delta \sigma_i^o = \eta \frac{\partial E}{\partial \sigma_i^o} \qquad (17)$$

where η is learning rate.

Hence the learning rules can be written as follows:

$$c_i^o(t) = c_i^o(t-1) + \eta \frac{\partial E}{\partial c_i^o} \qquad (18)$$

$$\sigma_i^o(t) = \sigma_i^o(t-1) + \eta \frac{\partial E}{\partial \sigma_i^o} \qquad (19)$$

where $c_i^o(t)$ and $c_i^o(t-1)$ are the corresponding centers of membership function at time t and previous time $t-1$ for the i th feature in the o th output node. $\sigma_i^o(t)$ and $\sigma_i^o(t-1)$ are the corresponding variances of membership function at time t and previous time $t-1$ for the i th feature in the o th output node. Also, $\frac{\partial E}{\partial c_i^o}$ and $\frac{\partial E}{\partial \sigma_i^o}$ can be computed by the following two equations:

$$\frac{\partial E}{\partial \sigma_i^d} = \left(2\, e^{-(\frac{d_i - c_i^d}{\sigma_i^d})^2} . \frac{d_i - c_i^d}{\sigma_i^d} \right) \qquad (20)$$

$$\frac{\partial E}{\partial \sigma_i^d} = \left(2\, e^{-(\frac{d_i - c_i^d}{\sigma_i^d})^2} . \frac{(d_i - c_i^d)^2}{(\sigma_i^d)^3} \right) \qquad (21)$$

We iterate the learning performing Equations (16)-(21) until the classification accuracy rate is satisfied. Finally, an efficient AFLN for classification is produced successfully.

3. Experiments

In order to evaluate the performance of our proposed feature subset selection algorithm and the adaptive fuzzy learning network (AFLN) for document classification, a news data set is applied to evaluate our methods.

3.1 Data Sets

The daily documents published from 1999/05/01 to 1999/05/15 at China Times Site [7] were collected as our data set. The numbers of the training and testing documents in each category are summarized in Table 1.

Table 1. The summary of data sets

Data set Category	# of training documents	# of testing documents	Total
Sports News	106	106	212
Economy News	110	110	220
News about Taiwan, Hong Kong and Mainland China	62	62	124
Entertainment News	54	54	108
International News	66	66	132
Political News	144	144	288
Social News	60	60	120
Star News	106	106	212
Total	708	708	1416

3.2 Feature Subset Selection

Initially we extract keywords that have complete semantic meanings from training documents by MMSEG program [8] and thus 21490 different keywords are produced. We then apply various feature selection algorithms to select useful features from these 21490 keywords and use these features to construct their corresponding AFLNs for document classification.

In the experiment, we incorporate two different local scoring functions, term frequency (TF) function and document frequency (DF) function, into our feature subset selection algorithm to select useful keywords from the 21490 original keywords individually so that two kinds of feature set, denoted TF-feature set and DF-feature set, are produced. Then we use these two feature sets to construct their corresponding AFLNs and compare their accuracy rates. The training and testing results of these experiments by different feature sizes from 50 to 2000 are reported in Figures 3 and 4, respectively. From the results of Figures 3 and 4, it is evident that the classifier constructed using DF-feature set produces better results than TF-feature set does. This is especially important when we realize that the feature set was reduced to less than 1% of its original size when using DF-feature set.

3.3 Classification Experiments

In this subsection, we apply the following methods for comparison with AFLN:

a. SBC − (Similarity-Based Classifier −): a traditional automatic classifier, which uses vectors to represent documents and categories. Binary weights are used in these vectors.

b. SBC + (Similarity-Based Classifier +): an advanced similarity-based classifier with typical keyword-weighting scheme, TF×IDF [6].

Figures 5 and 6 compare the training and testing results of different methods. For reference, we also include the probability of correct assignment by a random choice, i.e., the average precision of random assignment. In the experiments, the AFLN outperforms SBC + significantly, which in turn is better than SBC−. The improvement of SBC+ over SBC− comes from the use of TF×IDF weights instead of the binary weights of SBC−. The improvement of AFLN over SBC+ and SBC− comes from the learning ability of proper category vector and human knowledge as to the importance of features.

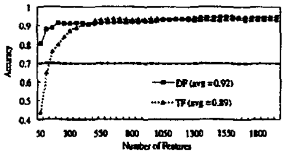

Figure 3. Training results using different local scoring functions

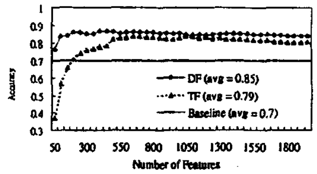

Figure 4. Testing results using different local scoring functions

4. Conclusion

We presente a fair feature subset selection algorithm based on designing an efficient scoring procedure to assign high scores to keywords deemed useful and lower scores to less useful keywords. This algorithm overcomes many of problems with existing methods [9]: it has a sound theoretical foundation; it is effective in recognizing useful keywords to classification; and most importantly, keywords in different categories are treated with justice by this algorithm.

Besides, we also develope an adaptive fuzzy learning network (AFLN) for classification with high-speed classification and high accuracy rate. More important, the AFLN actually has learning ability so that human knowledge can be acquired and proper category vectors are induced successfully. The empirical results show that the accuracy rate of our classifier achieves 92% and 86% in training set and testing set respectively, when we reduce the feature set to less than 1% of its original size (from 21490 to 200) using our methods.

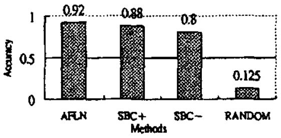

Figure 5. Training results of different methods

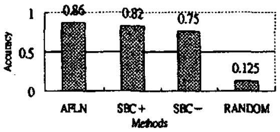

Figure 6. Testing results of different methods

5. References

[1] Hao-Kuang Ku, Automatic Network Documents Classification, *Master Thesis, Department of Information Management, National Taiwan University, Taipei, Taiwan*, June 1997.

[2] R. E. Filman and S. Pant, "Searching the Internet," *In IEEE Internet Computing*, vol. 2, no. 4, pp. 21-23, Aug. 1998.

[3] Alta Vista. : http://www.altavista.digital.com/

[4] Yam Web site : http://www.yam.com.tw/

[5] Inference Find Web site : http://www.infind.com.tw/

[6] G. Salton, *Automatic Text Processing. The Transformation, Analysis and Retrieval of Information by Computer*, Addison-Wesley, 1989.

[7] China Times Web site : http://www.chinatimes.com.tw/

[8] C. H. Tsai, "MMSEG : A World Identification System for Mandarin Chinese Text Based on Two Variations of the Maximum Matching Algorithm," *Web Publication*, May 1996, http://casper.beckman.uiuc.edu/~c-tsai4/chinese/wordseg/mmseg.html

[9] Yu-Jung Chang, Scalable Feature Selection for Web Page Classification by Fuzzy Ranking Analysis, Master Thesis, Department of Electronic Engineerin,, National Taiwan University of Science and Technology, Taipei, Taiwan, 1998.

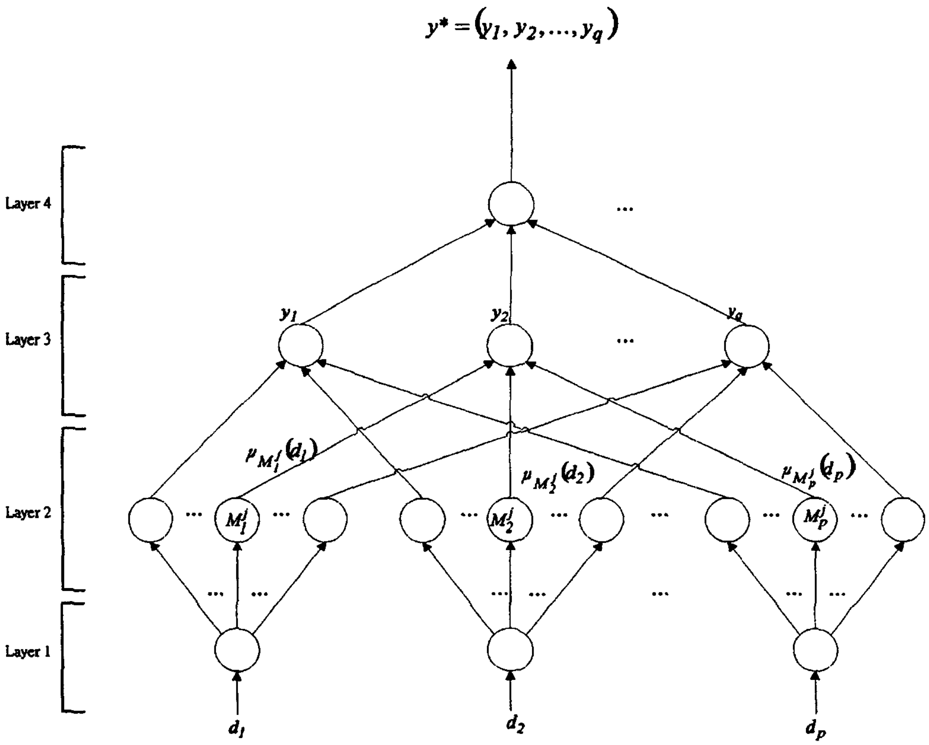

Figure 2. Structure of AFLN