Classification of Web Documents using Fuzzy Logic Categorical Data Clustering

George E. Tsekouras, Christos Anagnostopoulos, Damianos Gavalas, and Economou Dafni
University of the Aegean, Department of Cultural Techonoly and Communication, Laboratory of Intelligent Multimedia, Sapfous 5, Mytilene, Lesvos Island, 81100, Greece,
gtsek@ct.aegean.gr
WWW home page: http://www.aegean.gr/culturaltec

Abstract. We propose a categorical data fuzzy clustering algorithm to classify web documents. We extract a number of words for each thematic area (category) and then, we treat each word as a multidimensional categorical data vector. For each category, we use the algorithm to partition the available words into a number of clusters, where the center of each cluster corresponds to a word. To calculate the dissimilarity measure between two words we use the Hamming distance. Then, the classification of a new document is accomplished in two steps. Firstly, we estimate the minimum distance between this document and all the cluster centers of each category. Secondly, we select the smallest of the above minimum distance and we classify the document in the category that corresponds to this distance.

1 Introduction

The continuous growth in the size and use of the Internet creates difficulties in searching information. One of the most important functions of the Internet is the information retrieval [1]. The main problem involved in information retrieval is that the web pages are diverse, with an enormous number of ill-structured and uncoordinated data sources and a wide range of content, format and authorships [2]. New pages are being generated at such rate that no individual or organization is capable of keeping track of all of them, organizing them or presenting adequate tools for managing, manipulating and accessing the associated information.

In order to build efficient information retrieval systems, a solution is to perform web document classification under certain similar characteristics [2]. The steps to classify web documents involve the utilization of already classified documents in combination with specially designed algorithms to extract words and phrases usually called *items* [3]. These items and their synonyms form collections where indices are

used to indicate which item is related to a specific class. Moreover, the collections along with the respective indices carry information about how strong each item is associated with a specific class [4]. The task of assigning a new document to a class is accomplished thought the definition of appropriate similarity or dissimilarity measure.

One of the most efficient approaches to classify web documents is to use cluster analysis. Clustering is an unsupervised learning method that partitions a set of patterns into groups (clusters), where elements (patterns) that belong to the same group are as similar as possible, while elements belonging to different groups are as dissimilar as possible [5]. We distinguish two main categories of clustering algorithms. The first category is called hierarchical clustering and it produces nested partitions generated by sequential agglomerative of divisive algorithms, which are based on distance measures between clusters (such as single link, average link, complete link, etc) [6]. A major drawback of sequential algorithms is their strong dependence on the order in which the patterns are elaborated. The second category is referred to the so-called partitional clustering algorithms [7]. Their implementation is based on the alternating optimization of a certain objective function. Many clustering algorithms assume that the patterns are real vectors, called numerical patterns. However, in the web we usually consider non-numerical patterns. These patterns can be categorized into two types: (a) web documents presented in a specific document formats like HTML containing control strings and text [8], and (b) web server log files containing access sequences of web pages visited by specific users [9]. Relations between non-numerical patterns can be obtained by using the well-known Hamming distance or the Levenshtein distance [10].

In this paper we propose a systematic approach to cluster web documents. The basic idea is to define a number of web page categories and to download a number of pages each of which is related to a category. For each category we extract a number of words, which are treated as categorical data vectors. Then, we apply a novel algorithm to cluster these words into a number of clusters. The resulted cluster centers are words from the original data set. Finally, each web document is classified to a category based on the minimum Hamming distance.

2 Categorical Data Clustering

Categorical data clustering (CDC) is an important operation in data mining. A well-known categorical data clustering approach is the fuzzy *c*-modes [11]. Next, we describe the basic design steps of this algorithm.

Let $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n\}$ be a set of categorical objects. Each object is described by a set of attributes $A_1, A_2, ..., A_p$. The *j*-th attribute A_j $(1 \le j \le p)$ is defined on a domain of q_j categories. Thus, the *k*-th categorical object $\boldsymbol{x}_k$ $(1 \le k \le n)$ is described as: $\boldsymbol{x}_k = [x_{k1}, x_{k2}, ..., x_{kp}]$.

Let $\boldsymbol{x}_k = [x_{k1}, x_{k2}, ..., x_{kp}]$ and $\boldsymbol{x}_l = [x_{l1}, x_{l2}, ..., x_{lp}]$ be two categorical objects. Then, the matching dissimilarity between them is defined as [11],

$$D(\boldsymbol{x}_k, \boldsymbol{x}_l) = \sum_{j=1}^{p} \delta(x_{kj}, x_{lj}) \quad (1 \le k \le n,\ 1 \le l \le n,\ \ k \ne l) \tag{1}$$

where

$$\delta(x_{kj}, x_{lj}) = \begin{cases} 0, & if \quad x_{kj} = x_{lj} \\ 1, & otherwise \end{cases} \tag{2}$$

The fuzzy *c*-modes algorithm is based on minimizing the following objective function,

$$J = \sum_{k=1}^{n} \sum_{i=1}^{c} (u_{ik})^m \, D(\boldsymbol{x}_k, \boldsymbol{v}_i) \tag{3}$$

subject to the following equality constraint,

$$\sum_{i=1}^{c} u_{ik} = 1 \tag{4}$$

where n is the number of categorical objects, c is the number of clusters, $\boldsymbol{x}_k$ $(1 \le k \le n)$ is the k-th categorical object, $\boldsymbol{v}_i$ $(1 \le i \le c)$ is the i-th cluster center, u_{ik} is the membership degree of the k-th categorical object to the i-th cluster, and $m \in (1, \infty)$ is a fuzziness parameter to adjust the membership degree weighting effect.

The membership degrees that solve the above constrained optimization problem are given as [11],

$$u_{ik} = 1 \Bigg/ \left[\sum_{j=1}^{c} \left(\frac{D(\boldsymbol{x}_k, \boldsymbol{v}_i)}{D(\boldsymbol{x}_k, \boldsymbol{v}_j)} \right)^{1/(m-1)} \right] \quad (1 \le i \le c,\ 1 \le k \le n) \tag{5}$$

On the other hand, the locations of the cluster centers (modes) that minimize the objective function in (3) are determined as [11]:

$\boldsymbol{v}_i = [v_{i1}, v_{i2}, ..., v_{ip}]$, where

$$v_{ij} = a_j^r \in DOM(A_j) \text{ with, } \sum_{k, x_{kj} = a_j^r} (u_{ik})^m \ge \sum_{k, x_{kj} = a_j^t} (u_{ik})^m \qquad (1 \le t \le q_j, r \ne t) \tag{6}$$

3 The Proposed Fuzzy Categorical Data Clustering Algorithm

The proposed fuzzy clustering algorithm utilizes a number of steps each of which is described within the next subsections.

3.1 Potential Based Clustering

The potential of the k-th categorical object is defined as follows,

$$P_k = \sum_{l=1}^{n} \exp\{-a \, D(\boldsymbol{x}_k, \boldsymbol{x}_l)\} \tag{7}$$

where $D(\boldsymbol{x}_k, \boldsymbol{x}_l)$ is given in (1) and $a \in (0,1)$. Observing that an object with a high potential value is a good nominee to be a cluster center, the potential-based clustering algorithm is given next:

Select values for the design parameter $a \in (0,1)$ and for the parameter $\beta \in (0,1)$. Initially, set the number of clusters equal to c=0.

Step 1) Using eq. (7) determine the potential values for all data vectors $\boldsymbol{x}_k$ $(1 \le k \le n)$

Step 2) Set c=c+1

Step 3) Calculate the maximum potential value $P_{\max} = \max_k \{P_k\}$ and select the object $\boldsymbol{x}_{\max}$ that corresponds to $P_{\max}$ as the center element of the c-th cluster: $\boldsymbol{v}_c = \boldsymbol{x}_{\max}$

Step 4) Remove from the set $\boldsymbol{X}$ all the categorical objects having similarity with $\boldsymbol{x}_{\max}$ greater than β and assign them to the c-th cluster

Step 5) If $\boldsymbol{X}$ is empty stop; Else turn the algorithm to step 2.

3.2 Cluster Merging

In the first place we use (1) to calculate the matching dissimilarities between all pairs of cluster centers. Then, we compute the weighted matching dissimilarities between pairs of clusters according to the following formula,

$$D_w(\boldsymbol{v}_i, \boldsymbol{v}_j) = D(\boldsymbol{v}_i, \boldsymbol{v}_j) \sqrt{\sum_{k=1}^{n} u_{ik} \sum_{k=1}^{n} u_{jk} \Big/ \left(\sum_{k=1}^{n} u_{ik} + \sum_{k=1}^{n} u_{jk} \right)} \quad (1 \le i, j \le c;\ i \ne j) \tag{8}$$

where $\sum_{k=1}^{n} u_{ik}$ and $\sum_{k=1}^{n} u_{jk}$ are the fuzzy cardinalities of the i-th and j-th cluster. To decide which clusters are similar enough to be merged, we use the following similarity relation between two distinct clusters:

$$S_{ij} = \exp\{-\theta D_w(\boldsymbol{v}_i, \boldsymbol{v}_j)\} \quad (1 \le i, j \le c,\ \ i \ne j) \ \text{ and } \ \theta \in (0,1) \tag{9}$$

3.3 Validity Index

Cluster validity concerns the determination of the optimal number of clusters. In this section we use the cluster validity index developed in [12]. This validity index is given by the following equation,

$$G = \frac{\sum_{i=1}^{c} \dfrac{\sum_{k=1}^{n} (u_{ik})^m D(\boldsymbol{v}_k, \boldsymbol{v}_i)}{n_i}}{\sum_{i=1}^{c} \sum_{\substack{j=1 \\ j \ne i}}^{c} (u_{ij})^m D(\boldsymbol{v}_i, \boldsymbol{v}_j)} \tag{10}$$

where

$$n_i = \sum_{k=1}^{n} u_{ik} \quad , \quad 1 \le i \le c$$

and

$$ì_{ij} = \frac{1}{\sum_{\substack{l=1 \\ l \neq j}}^{c} \left(\frac{D(\boldsymbol{v}_j, \boldsymbol{v}_i)}{D(\boldsymbol{v}_j, \boldsymbol{v}_l)} \right)^{1/(m-1)}} \quad (i \neq j)$$

is the membership degree between the *i*-th and the *j*-th cluster center taking into account the rest of the cluster centers. The basic idea is to select the partition that corresponds to the minimum value of the index $G_{\min}$.

3.4 The Algorithm

Step 1) Apply the potential-based clustering algorithm to obtain a number of cluster centers $\boldsymbol{v}_i$ $(1 \le i \le c)$

Step 2) Using (5) calculate the u_{ik} $(1 \le i \le c, 1 \le k \le n)$ and determine the value of J in (3)

Step 3) Set $J^{old} = J$

Step 4) Using (6) update the cluster centers

Step 5) Using (5) calculate the u_{ik} $(1 \le i \le c, 1 \le k \le n)$ and determine the value of J in (3).
If $\left| J_m(\boldsymbol{U}, \boldsymbol{V}) - J_m^{old}(\boldsymbol{U}, \boldsymbol{V}) \right| \le \varepsilon$ go to step 6, else go to step 3

Step 6) Calculate the value of the validity index G in (10)

Step 7) Calculate all the similarities S_{ij} $(1 \le i, j \le c)$ in (9) and select the maximum: $S_{\max} = \max_{i,j} \{ S_{ij} \}$. The clusters that correspond to $S_{\max}$ are denoted as: i_0 and j_0

Step 8) If $S_{\max} > \delta$ then
merge the clusters i_0 and j_0 into a new cluster l_0 as follows,
$u_{l_0 k} = (u_{i_0} + u_{j_0}) / 2$
and set $c = c - 1$.
Using (6) determine the new cluster centers $\boldsymbol{v}_i$ $(1 \le i \le c)$,
calculate the new value of J and go to step 3
Else
Stop
EndIf

Step 9) Select the partition that corresponds to the minimum valued of the validity index ($G_{\min}$)

4 Web Document Classification

To apply the algorithm we choose the following 10 web page categories: (1) process engineering, (2) organic chemistry, (3) inorganic chemistry, (4) material analysis and design, (5) electrical engineering, (6) hardware, (7) software, (8) mechanical engineering, (9) civil engineering, and (10) marine science. We searched the yahoo.com and collected 28678 pages for all of the categories. Then, we used the 20000 to train the algorithm and the rest 8678 to test its performance.

Table 1: Number of clusters obtained by the algorithm for each category.

Category	Number of Clusters
Process Engineering	28
Organic Chemistry	33
Inorganic Chemistry	24
Material Analysis and Design	28
Electrical Engineering	43
Hardware	29
Software	35
Mechanical Engineering	46
Civil Engineering	52
Marine Science	38

The web documents are described in HTML format and texts are marked by the HTML tags. For each website category we downloaded the documents using a web crawler and we removed all the HTML tags in order to extract the texts from the web pages. We used a dictionary to separate the nouns from other words, since there is stronger relation between nouns and the web page theme. In the next step, we applied the algorithm developed in [13] to filter out insignificant nouns and certain types of word endings, like “ing” and “ed”. Then, we selected the 1000 most frequently reported words, using the inverse document frequency (IDF) for each word. The IDF is given by the following equation [14],

$$fr_{IDF} = \frac{f_w}{f_{w\_\max}} \log\left(\frac{P}{P_w}\right)$$

where f_w is the frequency of occurrence of the word in the category’s document collection, $f_{w\_\max}$ is the maximum frequency of occurrence of any word in the category’s collection, P is the number of documents of the whole collection, and P_w is the number of documents that include this word.

From the set of the 10000 words we found the word with the maximum number of characters, which defines the dimensionality of the discrete space and is denoted as p. Then, we represented the rest of the words in a sequence of p characters inserting where it is necessary the blank character. Therefore for each category, the k-th word can be described as:

$$\boldsymbol{x}_k = [x_{k1}, x_{k2}, ..., x_{kp}] \qquad (1 \le k \le 1000)$$

To this end, all the words have been transformed into categorical data vectors and thus, we can use the clustering approach presented in the previous section to partition the available 1000 words into a number of clusters for each category. Table

1 presents the categories and the respective number of clusters (i.e. words) obtained by the algorithm. For example, for the category "process engineering" the cluster centers (keywords) in alphabetical order are:

absorption, balance, batch, column, component, conservation, distillation, dryer, energy, equilibrium, evaporation, exchanger, fluid, furnace, heat, kinetics, liquid, mass, pressure, pump, reactor, saturation, separation, steam, temperature, thermodynamic, valve, viscosity

Table 2: Classification results of the proposed algorithm.

Category	Fuzzy *c*-Modes	*c*-Modes
Process Engineering	65.78%	57.86%
Organic Chemistry	67.11%	68.01%
Inorganic Chemistry	61.56%	60.23%
Material Analysis and Design	67.63%	63.10%
Electrical Engineering	60.89%	59.12%
Hardware	62.03%	58.85%
Software	66.34%	52.71%
Mechanical Engineering	60.09%	66.58%
Civil Engineering	69.90%	62.08%
Marine Science	66.55%	59.32%

Keywords extracted from the proposed approach can be used to automatically classify unknown texts. For this purpose we used the 8678 pages mentioned previously. Each document consists of a sequence of words $(\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_r)$ and each word has a minimum distance to one of the cluster centers of each of the 10 categories. Thus, for each document we determine 10 minimum distances. Then, the document is assigned to the category that appears the smallest of the above 10 distances. The results of this simulation are presented in table 2. For comparison reasons, this table also reports the results obtained by the *c*-modes [15] when it is used in the place of the fuzzy *c*-modes. From this table, we clearly see that except the categories "organic chemistry" and "mechanical engineering" the fuzzy *c*-modes outperformed the *c*-modes. The qualitative reason is that there are common words in all of the engineering categories. This fact directly implies the presence of uncertainty in the data set and therefore the use of the fuzzy logic-based clustering appears to be more efficient since it is able to quantitatively model this uncertainty.

5 Conclusions

We have shown how categorical data fuzzy clustering can be implemented to classify web documents. The basic idea of the approach is to extract a set of words and then transform them into categorical data vectors. The dissimilarity between two distinct words is measured using the well-known Hamming distance. Then, we apply a sequence of steps aiming towards generating a number of clusters, each of which is described by a single word. The classification of web documents is accomplished by using the minimum Hamming distance. The simulation results verified the efficiency of the proposed method.

Acknowledgements: This work is supported by the General Secretariat of Research and Technology (Project "Software Application in Interactive Kids TV-MPEG-21", project framework "Image, Sound, and Language Processing", project number : ΕΗΓ-16). The participants are the University of the Aegean, the Hellenic Public Radio and Television (ERT) and the Time Lapse Picture Hellas.

References

1. Smith, K.A, and Ng, A.: Web page clustering using a self-organizing map of user navigation patterns, Decision Support Systems 35 (2003) 245-256
2. Macskassy, S. A., Banerjee, A., Davison, B. D., and Hirsh, H.: Human performance on clustering web pages: a preliminary study, Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining (1998)
3. Anagnostopoulos, I., Anagnostopoulos, C., Loumos, V., and Kayafas, E.: Classifying web pages employing a probabilistic neural network, IEE Proceedings on Software 151(3) (2004) 139-150
4. Qi, D., and Sun, B.: A genetic k-means approach for automated web page classification, Proceedings of the 2004 IEEE International Conference on Information Reuse and Integration (2004) 241-246
5. Jain, A.K., Murty, M.N., and Flynn, P.J.: Data clustering: a review, ACM Computing Survey 31(3) (199) 264-323
6. Runkler, T.A., and Bezdek, J.C.: Web mining with relational clustering, International Journal of Approximate Reasoning 32 (2003) 217-236
7. Bezdek, J.C., and Pal, K.: Fuzzy models for pattern recognition: methods that search for structures in data, IEEE Press (1992), New York, NY
8. Manning, C.D., and Schutze, H.: Foundations of statistical natural language processing, MIT Press (1999), Cambridge, MA
9. Punin, J.R., Krishnamoorthy, M.S, and Zaki, M.J.: Web usage mining-languages and algorithms, Technical Report, Rensselaer Polytechnic Institute, NY (2001)
10. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions and reversals, Sov. Phys. Dokl. 6 (1966) 705-710
11. Z. Huang, and M. K. Ng, A fuzzy k-modes algorithm for clustering categorical data, *IEEE Transactions on Fuzzy Systems*, Vol. 7, no 4, 1999, pp. 446-452
12. Tsekouras, G. E., Papageorgiou, D., Kotsiantis, S., Kalloniatis, C., and Pintelas, P.: Fuzzy Clustering of Categorical Attributes and its Use in Analyzing Cultural Data, International Journal of Computational Intelligence 1(2) (2004) 147-151
13. Chen, J., Miculcic, A., and Kraft, D.H.: An integrated approach to information retrieval with fuzzy clustering and inferencing, in *Knowledge Management in Fuzzy DataBases*, Pons, O., Vila, M.A., and Kacprzyk, J. (Eds), Physic Verlag, Vol. 163 (2000)
14. Jones, K.S.: A statistical interpretation of tem specificity and its application in retrieval, J. Domument 28(1) (1972) 11-20
15. Huang, Z.: Extensions of the k-means algorithm for clustering large data sets with categorical values, Data Mining and Knowledge Discovery 2 (1998) 283-304