Using Symbolic Objects to Cluster Web Documents

Esteban Meneses
Costa Rica Institute of Technology
Computing Research Center
Cartago, Costa Rica
esteban.meneses@acm.org

Oldemar Rodríguez-Rojas
University of Costa Rica
School of Mathematics
San José, Costa Rica
oldemar.rodriguez@predisoft.com

ABSTRACT

Web Clustering is useful for several activities in the WWW, from automatically building web directories to improve retrieval performance. Nevertheless, due to the huge size of the web, a linear mechanism must be employed to cluster web documents. The dynamic cluster (*k-means*) is one classic algorithm used in this problem. We present a variant of the vector model to be used with the dynamic cluster algorithm. Our representation uses symbolic objects for clustering web documents. Some experiments were done with positive results and future work is optimistic.

Categories and Subject Descriptors

I.7.m [**Document and Text Processing**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*performance measures*

General Terms

Algorithms

Keywords

Symbolic Data Analysis, Web Clustering

1. INTRODUCTION

The aim of web clustering is to group web documents into very identifiable classes in such a way that common web activities can be improved. However, due to the huge quantity of web documents, techniques focused on web clustering must be efficient enough to manipulate even millions of elements. Because of that, linear models are generally preferred. One such linear model is the dynamic cluster (*k-means*) [4] algorithm, which can iteratively cluster a document collection into k classes.

The traditional dynamic cluster algorithm uses the vector model [3] for representing documents. In this model, given a web document collection $\mathcal{D}$ with n documents, every element $d \in \mathcal{D}$ is represented by a vector x_d in a m-dimensional space, typically $\mathcal{R}^m$. Each dimension stands for the frequency of term t_i in the document d, where $i = 1, 2, ..., m$. A dictionary with the m terms $\{t_1, t_2, ..., t_m\}$ can be build with the m more frequent terms in the collection $\mathcal{D}$ [2], for example.

The dynamic cluster algorithm works by finding k centers of gravity $\{g_1, g_2, ..., g_k\}$, each of which will attract some elements and form a cluster. The center g_i represents cluster C_i. Initially, these k centers are randomly selected. Then, iteratively the elements of $\mathcal{D}$ will be assigned to some cluster. In each phase every element is associated with the nearest center according to some *distance* measure λ. After that, centers are recalculated to minimize the criteria:

$$\sum_{i=1}^{k} \sum_{d_j \in C_i} \lambda(d_j, g_i)$$

There are many distance measures in the literature [6], but *Jaccard Extended Distance* has showed good clustering performance [5]. This function was used in this work.

This poster presents a variant for the representation of web documents using symbolic objects. The details of some experiments are first showed. Conclusions and future work are left for the final part.

2. SYMBOLIC WEB CLUSTERING

We extended the standard dynamic cluster algorithm using symbolic objects [1] instead of real-valued vectors for representing web documents. Symbolic objects are better at representing *concepts* rather than *individuals*. Its strength resides in its capacity for storing the *variablility* of concepts. In this case, the web page in considered as a concept that is formed by sections of the HTML code.

Symbolic objects are vectors where each entry can have any type: scalar, set, interval, histogram, graph, you name it. In the particular case of this poster, the histogram representation was explored.

Each document is represented with a symbolic vector with four entries represented by histograms. These four variables correspond to frequency of terms in four sections of the HTML code: text, bold, links and title. Each symbolic object is built after the web collection is analyzed and the most frequent terms are obtained. Previously, *stopwords* are eliminated and Porter's stemming algorithm is applied to every word in any of those four sections.

More formally, each document d in the collection $\mathcal{D}$ is represented by the symbolic object x_d in m histogram dimensions $\{x_{d1}, x_{d2}, ..., x_{dm}\}$. Each variable x_{di} is a normalized histogram $\{x_{di1}, x_{di2}, ..., x_{dip}\}$ with p *categories* or *modalities*.

The distance measure used is based on the *affinity index* [1]:

$$\lambda(x_d, x_{d'}) = 1 - \sum_{i=1}^{n} w_i \sum_{j=1}^{p} \sqrt{x_{dij} * x_{d'ij}}$$

WWW2006, May 22–26, 2006, Edinburgh, UK.
.

Model	Rand Index	Mutual Info.	Time
Vector	0.6982	0.1098	0.76 s
Histogram-10	**0.7064**	**0.1395**	**0.06 s**
Histogram-20	0.6842	0.1102	0.10 s
Histogram-30	0.6830	0.1025	0.12 s

Table 1: Results for the F-series

Model	Rand Index	Mutual Info.	Time
Vector	0.8863	0.2601	7.82 s
Histogram-10	0.8387	0.2119	13.3 s
Histogram-20	0.9003	0.3011	4.86 s
Histogram-30	**0.9087**	**0.3165**	**0.92 s**

Table 2: Results for the J-series

3. RESULTS

We used the F-series and the J-series web collections [2] for testing our approach. The F-series contains 98 documents from 4 classes, while the J-series is formed by 185 documents from 10 categories. The evaluation of the resulting clusters was made using the *rand index* and the *mutual information* measures [6]. The former computes how similar is the clustering obtained to the manual clustering (also present in the F and J series). The latter calculates the quality of the clusters according to how compact the cluster is. Experiments were repeated 50 times each over both databases.

The results for F-series are presented in Table 1. The vectorial representation of the F-series is formed by a 98 x 332 matrix. On the other hand, each symbolic model is accompanied by the number of categories in the histogram, i.e. *p*. The symbolic representation of histograms with 10 categories appears to be slightly better than the vector representation in both indexes. Adding more categories to the histograms doesn't improve the clustering.

In table 1 it can be seen the time taken by the different approaches to cluster F-series. The symbolic object shows how efficient such representation can be. Using 10 categories in histograms, symbolic objects are near 12 times faster than vector representation.

Table 2 shows the results for the J-series. In this case there is a sensible improvement when symbolic objects are used. Also, there is an interesting curve associated with the number of categories in the histogram. Using 30 categories provides the best results, as more categories are added no improvement is obtained. Few than 30 categories lowers the clustering efficiency and quality. In table 2 the time required to cluster the J-series using 10 or 20 categories is bigger than using 30 categories. This occurs because with few categories the dynamic cluster is prune to get a clustering where a cluster has no associated elements, making necessary a re-initialization. Nevertheless, using symbolic representation can make 7 times faster to cluster this web collection if compared with traditional representation.

Figure 1 shows two PCA or *principal component analysis* [1] over the F-series. The PCA is a dimension reduction technique. The left graphic was made using the classic representation, while the right graphic was made with symbolic representation. The *inertia* percentage (how much information is conserved after the transformation) of the classic PCA was `4,24%`, but using symbolic PCA the inertia percentage is `54.37%`.

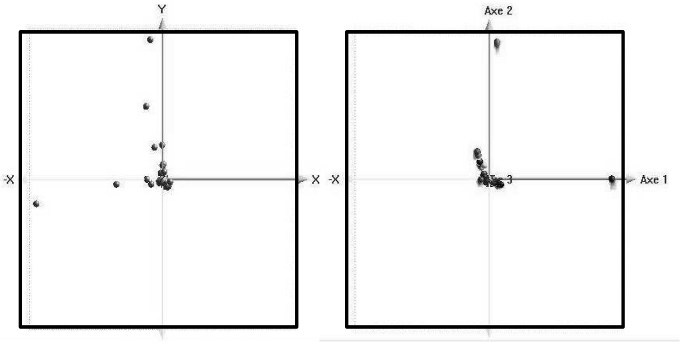

Figure 1: Principal component analysis for F-series

Besides this results, we also used the quality measures from [7]. The *quality of the partition* can be decomposed for each variable, to measure the importance of each variable to form the clustering. In one run of the J-series, using a symbolic object of histograms with 30 categories, the quality indexes for the different variables were: **text**=`0.0192`, **link**=`0.0044`, **bold**=`0.0067` and **title**=`0.3516`. Meaning this that the title in web documents helps a lot in building the clustering.

4. CONCLUSIONS AND FUTURE WORK

Symbolic objects can address the problem of web clustering with efficiency and semantic power. Given a symbolic object, it is more clear for the user what information is contained into it.

We are currently working on different distance measures between histograms and a variant of dynamic cluster algorithm to take into account what is called *strong forms* to better clustering a document collection.

In the future, we would like to extend the representation of the symbolic object used in this poster to include more information about the document.

5. REFERENCES

[1] H.-H. Bock and E. Diday. *Analysis of Symbolic Data.* Springer-Verlag, 2000.

[2] D. Boley, M. Gini, R. Gross, E.-H. Han, K. Hastings, G. Karypis, V. Kumar, B. Mobasher, and J. Moore.

[3] S. Chakrabarti. *Mining the Web.* Morgan Kaufmann Publishers, 2003.

[4] E. Diday and J. Simon. Cluster analysis. *Digital Pattern*, 1976.

[5] A. Schenker, M. Last, H. Bunke, and A. Kandel. A comparison of two novel algorithms for clustering web documents. *2nd IWWDA*, 2003.

[6] A. Strehl, J. Ghosh, and R. Mooney. Impact of similarity measures on web-page clustering. *AAAI-2000: Workshop of Artificial Intelligence for Web Search*, 2000.

[7] R. Verde, Y. Lechevallier, and M. Chavent. Symbolic clustering interpretation and visualization. *Journal of Symbolic Data Analysis*, 1(1), 2003.