
Combinatorial Clustering for Textual Data Representation in Machine Learning Models

Andrei Anghelescu and Ilya Muchnik

25 November 2002

1 Basic Ideas

In text stream analysis one of the main problems is finding an effective method to classify documents fast and correctly. This is the reason why dimensionality reduction and related methods of representation of significant information are critical to develop a good text classifier.

In this report we describe a novel purely combinatorial approach to obtain a meaningful representation of text data. There are two basic ideas that we realized in the current development of this approach. Namely, (1) Layered Clusters which induce over the entire data a stratification in a tower structure like a nesting doll (*Russian Matreshka*) [1][2], and, (2) parallel clustering of documents and their features (frequencies of words in our case). The clusters are sub-matrices of data which include each other according to the ordering given by the clustering model: the deepest cluster-matrix represents *the largest weighted quasi-clique* if the input data-matrix would be interpreted as a hyper-graph; its *effective weight* is also the largest possible; the second cluster includes the first one and represents the second level of a quasi-clique with less value of the effective weight in it, etc. The effective weight is used as the objective function whose optimization gives the above clustering structure for the data. Figure 1 shows usual changes of effective values of weights along the above mentioned stratification (for one of the data-matrices used in our analysis. In table 1 we show an example of a small matrix and its sub-matrices-clusters found by our method.

It is clear that this tower structure gives an ordinal scale for both documents and their features. The scale of documents points to which documents contain the most frequent words (of course, after having filtered stopwords), and, which include really rare words; similarly, a related feature scale shows which words are most frequent, and, what is their “location” in documents. In the present study we used for the purpose of learning, only the ordinal scale for document features. As an improvement, we plan to use a similar scale for documents in the near future.

So, if one gets a chain of nesting set of words (parts of our nesting clusters) presented in the considered data-matrix, one can follow the order of the chain

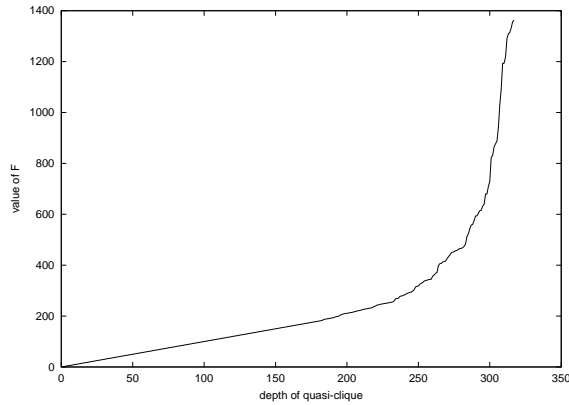


Figure 1: Effective values of weights

and interpret any position (subset of the corresponding words) as a particular subspace in which any classifier program can work. Of course, the most interesting case for us would be if we could construct a good classifier using a low dimension subspace. The fact that we can search candidates of those subspaces in the constructed chain gives a very efficient way to search the subspaces-candidates.

We realized this approach in three aspects: (1) when the subspace chain was generated based only on positive training data, (2) only on negative training data, and, (3) on the whole training data. It has to be emphasized that in every one of these cases of chosen subspaces a classifier design was based on using the whole set of documents, positive and negative. In the experiments we have applied an extremely simple procedure of searching an “efficient subspace” taken in the chain position which is in the middle point of the chain.

level 1 (input)					level 2					level 3 (deepest)				
1	2	3	4	5	x	2	3	4	5	x	x	x	x	x
2	6	6	6	4	x	6	6	6	4	x	6	6	6	x
0	6	6	6	3	x	6	6	6	3	x	6	6	6	x
0	6	6	6	2	x	6	6	6	2	x	6	6	6	x
0	6	6	6	0	x	6	6	6	0	x	6	6	6	x
2	6	6	6	0	x	6	6	6	0	x	6	6	6	x
1	2	0	1	2	x	x	x	x	x	x	x	x	x	x

Table 1: Example of embedded *quasi-cliques*

2 Formal description of developed clustering model

2.1 Theoretical Background: Quasi-Concave Set-Functions as Criteria for Combinatorial Clustering [2],[3]

Let us consider a finite set $W = \{1, 2, \dots, M\}$. For any pair composed of a subset $H (H \subseteq W)$ and its element $i (i \in H)$ we assign a numerical value $\pi(i, H)$, where the function $\pi(i, H)$ is a monotone increasing function of the second argument H . In other words, if $H_1 \subseteq H$ then $\pi(i, H) \geq \pi(i, H_1)$. Based on this function we build a set-character function which estimates “a cluster-ness” of H :

$$F(H) = \min_{i \in H} \pi(i, H). \quad (1)$$

Such defined function is quasi-concave [2]: for any H_1 and H_2

$$F(H_1 \cup H_2) \geq \min\{F(H_1), F(H_2)\}. \quad (2)$$

This property is the base for an universal polynomial procedure to find a set H which gives a maximum of the function. Usually the cluster-ness is interpreted as “a homogeneity” or “a matching of some particular property”, etc. [4]. The function $\pi(i, H)$ gives a lot of opportunities to model different types of a property that one might want to present in a clustering result. The presented work is based on the following informal hypothesis that *a large enough set of words with a high level of frequency which presents in a large subset of documents from one class, can be a base to build a classifier*. According to this hypothesis we have defined a particular $\pi(i, H)$.

Because we wanted to consider submatrices from a given matrix we use as the set W the set of indices which determine both columns and rows of the entire data-matrix $A = \|a_{ij}\|$. Below we use the following notations: $W = \{W_r, W_c\}$, where W_r is a set of indices for rows of the matrix A , W_c is a set of indices for columns of the same data-matrix ($|W_r| = N, |W_c| = n, N + n = M$). $H = \{H_r, H_c\}$ is a subset of W . We define, additionally, that $\pi(i, H) = 0$ if H doesn't include at least one element from both sets W_r and W_c :

$$\pi(i, H) = \begin{cases} \sum_{j \in H_c} a_{ij}, & \text{if } i \in H_r \\ \sum_{j \in H_r} a_{ij}, & \text{if } i \in H_c \\ 0, & \text{if } H_r = \emptyset \vee H_c = \emptyset \end{cases} \quad (3)$$

The sub-matrix-cluster (the above mentioned *quasi-clique*) that we are looking for is defined as H^* which gives the maximum value for the function $F(H) = \min_{i \in H} \pi(i, H)$. We have developed a method and software for finding this solution; the complexity of the developed procedure is a quadratic on the cardinality of W . The procedure has an additional property, very convenient from a practical point of view: in the process of finding a global maximum of $F(H)$ to build

a sequence of local maxima which are ordered according to their monotone increase values and which are subsets sequentially each other [[4], [5]].

We define H as a local maximum of $F(S)$, iff $F(H) > F(S)$ for all $S \supset H$; The above mentioned property is the sequence H_1, H_2, \dots, H_k of local maxima in which $H_k = H^*$ is a global maximum, $H_1 = W$, and, the sequence follows the order $F(H_1) < F(H_2) < \dots < F(H_k)$. Using the sequence one can easily test different extreme sub-matrices with different *level of word frequency* (in our experimental work we have used such levels which gave a subset of 100-300 words).

2.2 An outline of the procedure

The basic idea of the procedure is to introduce a particular chain of embedded subsets of W in order that the last subset of the chain will give the global maximum of $F(H)$. The procedure starts from the calculation of $F(W)$. According to definition of the function the value is related to a particular element $i_1 \in W$. After that the procedure calculates $F(W - \{i_1\})$ and finds the corresponding element $i_2 \in W - i_1$, and so on. Finally, it constructs the chain of subsets $\langle W, W - \{i_1\}, \dots, H_s, \dots, \{i_{N+n}\} \rangle$, where N is a cardinality of the set of documents, and n is the original set of words on which the set of documents are considered (the software implementation contains several improvements which greatly reduce the calculation time). Every element $H_s, s = 1, \dots, N + n$, of the chain is assigned the corresponding function value $F(H)$. The one with the largest value is associated to the global maximum of the function. Clearly, all local maxima are also on this chain.

3 Results

The experimental results are presented in [7]

References

- [1] B. Mirkin and I. Muchnik (2002) *Layered Clusters of Tightness Set Functions* Applied Mathematics Letters, 15, 147-151.
- [2] B. Mirkin and I. Muchnik (2002) *Induced Layered Clusters, Hereditary Mappings, and Convex Geometries*, Applied Mathematics Letters, 15, 293-298.
- [3] A. Genkin and I. Muchnik (1993) *Fixed Points Approach to Clustering*, Journal of Classification, No. 10:219-240
- [4] Y. Kempner and I. Muchnik (2002) *Quasi-concave set-function optimization on meet-semi-lattices* DIMACS Technical Reports, submitted

- [5] E.P. Xing, I. Muchnik, M. Zorn, and S. Spengler (2000) *Classification of Multi-Aligned Sequence Using Monotone Linkage Clustering and Alignment Segmentation* Bioinformatics Abstracts, DOE Human Genome Program, Contractor-Grantee Workshop VIII, Santa Fe, NM.
- [6] L.V. Shvartser, C. Kulikowski, I.B. Muchnik (2001) *Multiple sequence alignment using the quasi-concave function optimization based on the DIALIGN combinatorial structures* DIMACS Technical Reports, 2001-02
- [7] A. Anghelescu, I. Muchnik (2002) *An Experimental Evaluation of a Combinatorial Clustering Model for Textual Data Representation*
<http://mms-01.rutgers.edu/Documents/CombinatorialClustering/Experimental.v01.pdf>