Comparison of cluster algorithms for the analysis of text data using Kolmogorov complexity

Tina Geweniger, Frank-Michael Schleif
Alexander Hasenfuss,Barbara Hammer, Thomas Villmann
University Leipzig, Dept. of Medicine, 04103 Leipzig, Germany
{schleif,villmann}@informatik.uni-leipzig.de,{tg}@sonowin.de
Clausthal Univ. of Tech., Inst. of CS, 38678 Clausthal, Germany
{hasenfuss,hammer}@inf.tu-clausthal.de

No Institute Given

Abstract. In this paper we present a comparison of multiple cluster algorithms and their suitability for clustering text data. The clustering is based on similarities only, employing the Kolmogorov complexity as a similiarity measure. This motivates the set of considered clustering algorithms which take into account the similarity between objects exclusively. Compared cluster algorithms are Median kMeans, Median Neural Gas, Relational Neural Gas, Spectral Clustering and Affinity Propagation.
keywords: cluster algorithm, similarity data, neural gas, spectral clustering, message passing, kMeans, Kolmogorov complexity

1 Introduction

In the last years a variety of vector based clustering methods like self-organizing maps [**?**], neural gas (NG) [**?**] and affinity propagation (AP) [**?**] have been developed for a wide range of areas, e.g. bioinformatics, business, and robotics. Recent developments focus on algorithms which are purely based on the analysis of similarities between data. Beside AP respective algorithms are Median k-Means, Median and Relational NG [**?**,**?**] or spectral clustering [**?**] for example. These approaches have in common that they relax the condition that the objects of interest have to be embedded in a metric space. Instead, the only information used in these algorithms are the pairwise similarities. This ability provides greater flexibility of the algorithm if an embedding of data is impossible but a similarity description is available, for instance by external expert rating.

In this paper we compare the above mentioned algorithms in two experimental settings of text clustering. We consider two types of text sources: the first data are taken from the Multilingual Thesaurus of the European Union *Eurovoc*[1]. This database consists of laws and regulations in different categories whereas each text is available in multiple languages. It can be expected that the data should be clustered according to language specific features on the one hand side. On the other hand, contents specific features should also provide structural information for clustering. The second data set is a series of psychotherapy session transcripts of a complete psychodynamic psychotherapy with a respective clinical assessment of the session [**?**]. Here, it is known that narrative constructs provide information about the therapy state and, hence, should also be appearing in cluster structures. Both data sets are clustered by means of the priorly

[1] Obtained from: *http://europa.eu/eurovoc/sg/sga_doc/eurovoc_dif!SERVEUR/menu!prod!MENU?langue=EN*

given methods. The results are compared in terms of cluster agreement and the relation to the external description. As a measure for the cluster agreement we employ the Cohens-Kappa and variants.

The paper is organized as follows: first the encoding of the data is explained. After a short overview of the algorithms, the data sets are briefly described and finally the experimental results are presented and discussed.

2 Theoretical Background

Appropriate embedding of text data into metric spaces is a difficult task [**?**]. An approach is the analysis of textural data based on their Kolmogorov-Complexity [**?**]. It is based on the minimal description length (MDL) Z_x of a single document x and pairwise combined documents $Z_{x,y}$. The respective normalized information distance is given by:

$$NID_{x,y} = \frac{Z_{xy} - min(Z_x, Z_y)}{max(Z_x, Z_y)} \tag{1}$$

The normalized information distance is a similarity measure (distance metric) [**?**]. In particular, it is positive definite, i.e. $NID_{x,y} \geq 0$ with $NID_{x,x} = 0$, and symmetric $NID_{x,y} = NID_{y,x}$. Usually, the MDL is estimated by the compression length z according to a given standard compression scheme or algorithm. Then the NID is denoted as normalized compression distance (NCD) [**?**]. Due to technical reasons $NCD_{x,y}$ is non-vanishing in general but takes very small values [**?**]. Further, it violates usually the symmetry property in the sense that $NCD_{x,y} - NCD_{y,x} = \delta$ with $0 < \delta \ll 1$. Therefore, the symmetrized variant is frequently applied $NCD^s_{x,y} = \frac{(NCD_{x,y}+NCD_{y,x})}{2}$.

To estimate NCD, we used the Lempel-Ziv-Markow-Algorithm (LZMA) provided by the 7-Zip file archiver for the compression of the text data. z_x and z_y are the lengths of the compressed data sets T_x and T_y respectively. To obtain z_{xy} the two texts T_x and T_y are first concatenated to T_{xy} and subsequently compressed. In this way for all data pairs (x^i, x^j) the similarity (distance) $d_{ij} = NCD_{x^i,x^j}$ is calculated in both data sets (separately).

3 Algorithms

The role of clustering is to decompose a given data set $X = \{x_1, \ldots, x_n\}$ into clusters $C_1, \ldots, C_k \subset X$ such that the clusters are as homogeneous as possible. For crisp clustering as considered in the following, the sets C_i are mutually disjoint, and they cover the set X. We assume that data are characterized by pairwise dissimilarities d_{ij} specifying the dissimilarity of the data points x_i and x_j. For euclidean data, d_{ij} could be given by the squared euclidean metric, however, in general, every symmetric square matrix is appropriate. However, we assume here that only the distances are known but not the data objects itself. This restricts the set of applicable cluster algorithms. Following, we briefly describe recently developed approaches as well as classical ones which will later be compared.

3.1 Median k-Means

Median k-means (MKM) is a variant of classic *k-means* for discrete settings. The cost function for MKM is given by

$$E = \sum_{i=1}^{n} \sum_{j=1}^{p} X_{I(x^j)}(i) \cdot d\left(x^j, w^i\right) \tag{2}$$

where n is the cardinality of the set $W = \left\{w^k\right\}$ of the prototypes and p the number of data points. $X_{I(x^j)}(i)$ is the characteristic function of the winner index $I(x^j)$, which refers to the index of the prototype with minimum distance to x^j (winner).

E is optimized by iteration through the following two adaptation steps until convergence is reached.

1. determine the winner $I(x^j)$ for each data point x^j
2. Since for proximity data only the distance matrix is available the new prototype i has to be chosen from the set X of data points with $w^i = x^l$ where

$$l = \underset{l'}{\operatorname{argmin}} \sum_{j=1}^{p} X_{I(x^j)}(l) \cdot d(x^j, x^{l'}) \tag{3}$$

3.2 Median Neural Gas

A generalization of MKM incorporating neighborhood cooperativeness for faster convergence and better stability and performance is the Median Neural Gas (MNG). The respective cost function is

$$E_{MNG} = \sum_{i=1}^{n} \sum_{j=1}^{p} h_\lambda(k_i(x^j, W)) \cdot d(x^j, w^i) \tag{4}$$

with $h_\lambda(k_i(x^j, W))$ being the Gaussian shaped neighborhood function $h_\lambda(t) = exp(-t/\lambda)$ ($\lambda < 0$) and

$$k_i(x^j, W) = \#\left\{w^l | d(x^j, w^l) < d(x^j, w^i)\right\} \tag{5}$$

the winning rank. Then E_{MNG} can be optimized by iterating the following procedure:

1. $k_{ij} = k_i(x^j, W)$
2. and assuming fixed k_{ij} the prototype i is chosen as the data point with $w^i = x^l$ where

$$l = \underset{l'}{\operatorname{argmin}} \sum_{j=1}^{p} h_\lambda(k_{ij}) \cdot d(x^j, x^{l'})$$

3.3 Relational neural gas

Relational neural gas as proposed in [**?**] is based on a similar principle as MNG, whereby prototype locations can be chosen in a more general way than in MNG. Standard batch neural gas [**?**] has been defined in the euclidean setting, i.e. $x^i \in \mathbb{R}^m$ for some m. It optimizes the cost function $\frac{1}{2}\sum_{ij} \exp(-k_{ij}/\sigma^2)\|x^i - w^j\|^2$ with respect to the prototypes w_j where k_{ij} as above but using the Euclidean distance. $\sigma > 0$ denotes the neighborhood cooperation. For vanishing neighborhood $\sigma \to 0$, the standard quantization error is obtained. This cost function can be optimized in batch mode by subsequent optimization of prototype locations and assignments. Unlike k-means, neighborhood cooperation yields a very robust and initialization insensitive behavior of the algorithm.

The main observation of relational clustering is that optimum prototypes fulfill the relation $w^j = \sum_i \alpha_{ji} x^i$ with $\sum_i \alpha_{ji} = 1$. Therefore, the distance $\|x^i - w^j\|^2$ can be expressed solely in terms of the parameters α_{ji} and the pairwise distances $D = (d_{ij}^2)_{ij}$ of the data as

$$\|x^i - w^j\|^2 = (D \cdot \alpha_j)_i - 1/2 \cdot \alpha_j^t \cdot D \cdot \alpha_j. \tag{6}$$

Therefore, it is possible to find a formulation of batch NG which does not rely on the explicit embedding of data in a vector space:

init α_{ji} with $\sum_i \alpha_{ji} = 1$
repeat

- compute the distance $\|x^j - w^i\|^2$
- compute optimum assignments k_{ij} based on this distance matrix
- compute parameters $\tilde{\alpha}_{ij} = \exp(-k_{ij}/\sigma^2)$
- normalize $\alpha_{ij} = \tilde{\alpha}_{ij} / \sum_j \tilde{\alpha}_{ij}$

Obviously, this procedure can be applied to every symmetric dissimilarity matrix D, resulting in relational neural gas (RNG). The algorithm can be related to the dual cost function of NG:

$$\sum_i \frac{\sum_{ll'} \exp(-k_{il}/\sigma^2) \cdot \exp(-k_{il'}/\sigma^2) \cdot d_{ll'}}{4 \sum_l \exp(-k_{il}/\sigma^2))}$$

Since prototypes of RNG are represented virtually in terms of weighting factors α_{ij}, the algorithm yields a clustering rather than a compact description of the classes in terms of prototypes. However, it is possible to approximate the clusters by substituting the virtual prototypes w^j by its respective closest exemplar x_i in the data set X. We refer to this setting as 1-approximation of RNG.

3.4 Spectral clustering

Spectral clustering (SC) offers a popular clustering method which is based on a graph cut approach, see e.g. [**?**]. The idea is to decompose the vertices of the graph into clusters such that the resulting clusters are as close to connected components of the graph as possible. More precisely, assume vertices are enumerated by $1, \ldots, n$ corresponding to the data points x_i and undirected edges $i - j$ weighted with p_{ij} indicate the similarity of the vertices. We choose

$p_{ij} = -d_{ij} + \min_{ij} d_{ij}$, but alternative choices are possible, as described in [**?**]. Denote the resulting matrix by P, D denotes the diagonal matrix with vertex degrees $d_i = \sum_i p_{ij}$. Then normalized spectral clustering computes the smallest k eigenvectors of the normalized graph Laplacian $D^{-1} \cdot (D - P)$. The components of these eigenvectors constitute n data points in $\mathbb{R}^k$ which are clustered into k classes using a simple algorithm such as k-means. The index assignment gives the clusters C_i of X.

The method is exact if the graph decomposes into k connected components. As explained in [**?**], it constitutes a reasonable approximation to the normalized cut optimization problem $\frac{1}{2} \sum_i W(C_i, C_i^c)/\mathrm{vol}(C_i)$ for general graphs, where $W(A, A^c) = \sum_{i \in A, j \notin A} p_{ij}$ denotes the weights intersected by a cluster A and $\mathrm{vol}(A) = \sum_{j \in A} d_j$ the volume of a cluster A. Further, for normalized SC, some form of consistency of the method can be proven [**?**].

3.5 Affinity propagation

Affinity propagation (AP) constitutes an exemplar-based clustering approach as proposed in [**?**]. Given data points and pairwise dissimilarities d_{ij}, the goal is to find k exemplars x_i such that the following holds: if data points x^i are assigned to their respective closest exemplar by means of $I(i)$, the overall quantization error $\frac{1}{2} \sum_i d_{i,I(i)}$ should be minimum. This problem can be alternatively stated as finding an assignment function $I : \{1, \ldots, n\} \to \{1, \ldots, n\}$ such that the costs $-\frac{1}{2} \sum_i d_{i,I(i)} + \sum_i \delta_i(I)$ are maximum. Thereby, $\delta_i(I) = \begin{cases} -\infty & \text{if } I(i) \neq i, \exists j I(j) = i \\ 0 & \text{otherwise} \end{cases}$ punishes assignments which are invalid, because exemplar i is not available as a prototype but demanded as exemplar by some point j. Note that, this way, the number of clusters is not given priorly but it is automatically determined by the overall cost function due to the size of self-similarities $-d_{ii}$. These are often chosen identical for all i and as median value or half the average similarities. Large values $-d_{ii}$ lead to many clusters whereas small values lead to only a few or one cluster. By adjusting the diagonal d_{ii}, any number of clusters in $\{1, \ldots, n\}$ can be reached. AP optimizes this cost function by means of a popular and efficient heuristics. The cost function is interpreted as a factor graph with discrete variables $I(i)$ and function nodes $\delta_i(I)$ and $d_{i,I(i)}$. A solution is found by the max-sum-algorithm in this factor graph which can be implemented in linear time based on the number of dissimilarities d_{ij}. Note that $-d_{ij}$ can be related to the log probability of data point i to choose exemplar j, and $\delta_i(I)$ is the log probability of a valid assignment for i as being an exemplar or not. Thus, the max-sum algorithm can be interpreted as an approximation to compute assignments with maximum probability in the log-domain.

While spectral clustering yields decompositions of data points into clusters, affinity propagation also provides a representative exemplar for every cluster. This way, the clustering is restricted to specific types which are particularly intuitive. Further, unlike spectral clustering, the number of clusters is specified only implicitly by means of self-similarities.

4 Experiments and results

4.1 Measures for comparing results

To measure the agreement of two different cluster solutions we applied Cohen's Kappa κ_C [**?**]. Fleiss' Kappa κ_F as an extension of Cohen's Kappa is suitable for measuring the agreement of more than two classifiers [**?**]. For both measures yields the statement that if they are greater than zero the cluster agreements are not random but systematic. The maximum value of one is perfect agreement.

4.2 'Eurovoc' documents

The first data set consists of a selection of documents from the multilingual Thesaurus of the European Union "'Eurovoc"'. This thesaurus contains thousands of documents which are available in up to 21 languages each. For the experiments we selected a set of 600 transcripts in 6 different languages - 100 transcripts with the same contents in English, German, French, Spanish, Finnish and Dutch respectively. These transcripts can roughly be sorted into 6 different categories: International Affairs, Social Protection, Environment, Social Questions, Education and Communications, and Employment and Working Conditions.

First the distances between data are calculated according to $d_{ij} = NCD_{x^i,x^j}$ for the whole data set giving a large 600×600 matrix and, for each language set separately, yielding 6 small 100×100 matrices.

The first calculations in this section are based on the 600×600 matrix: As a first step the complete set containing all 600 documents was clustered into *six* groups using the above mentioned algorithms to investigate whether the language structure influences the clustering. It can be observed that all cluster solutions of the different methods are identical and exactly separating the data set according to the languages. In the second step we initialized a clustering into 12 clusters to examine the content based information for clustering. It can be observed that again a clean separation regarding to the languages was achieved. Yet, the segmentation within a single language was more or less irregular. For some languages there was no further break down at all, while others were separated into up to four different clusters. Hence, it seems that the language specifics dominate the clustering. Thereby, this behavior was shown more or less by all cluster algorithms. This fact is emphasized by the similarity of the cluster solutions judged in terms of Fleiss' Kappa (overall agreement) $\kappa_F = 0.6072$ referred as a substantial agreement. The the agreements of every two algorithms are estimated by Cohen's Kappa κ_C which also show a clear match:

	MNG	RNG	SC	AP
k-Means	0.56	0.50	0.59	0.73
MNG	—	0.58	0.54	0.73
RNG	—	—	0.57	0.63
SC	—	—	—	0.66

Noticeable is the clustering obtained by AP. Each language is separated into two clusters, which are almost identical with respect to the language sets. Measuring the similarity of the clusters between the different languages gives $\kappa_F = 0.8879$, a perfect agreement.

In the next step we examined each language separately using the 100×100 matrices. According to the given 6 text categories, we performed each clustering into 6 clusters. At first, we have to mention that the resulted cluster solutions

do not reflect the category system. This is observed for all languages and all algorithms. However, within each language the behavior of the different cluster approaches is more or less similar, i.e. comparing the cluster solutions gives high Kappa values above 0.4123 (moderate agreement). As an example, for English the solutions are depicted in Fig1a). with Fleiss' Kappa $\kappa_F = 0.5324$ (moderate agreement).

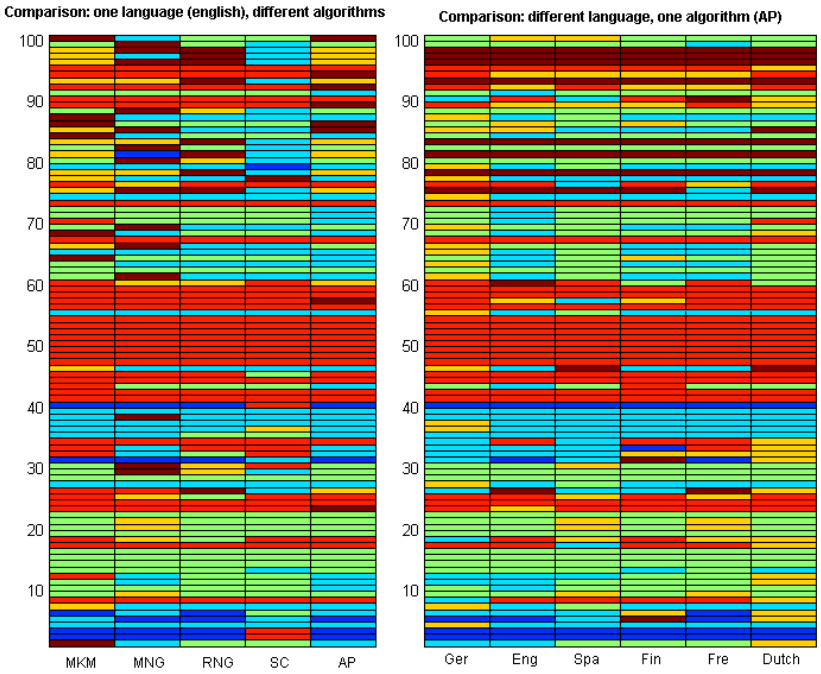

Fig. 1. (left) Comparison of the cluster solutions (six clusters) for the different algorithms. A moderate agreement can be observed.(right) Cluster solutions for the different languages obtained by AP-clustering. The similarity of the cluster results is high.

However, the averaged performance of the several algorithms according to the different languages varies. Despite AP and RNG, all algorithms show an instable behavior. AP and RNG offer similar cluster assignments independent from the language giving $\kappa_F = 0.5766$ and $\kappa_F = 0.3454$, respectively (see Fig.1b). This leads to the conclusion that the contents of the text can be clustered adequately in each language.

4.3 Psychotherapy transcripts

The second data set was a set of text transcripts of a series of 37 psychotherapy session dialogs of a psychodynamic therapy. Clustering these texts, using the NCD-distance as above, was again accomplished by applying all algorithms, here preferring a two-cluster solution according to the fact that the therapy was a two-phase process with the culminating point around session 17 [**?**]. The latter

fact is based on the evaluation of several clinical therapy measures [**?**]. Except SC, all algorithms cluster the data in a similar way such that the two process phases are assigned to separate clusters ($\kappa_F = 0.77$). This coincides with the hypothesis that narratives of the psychotherapy can be related to the therapeutic process.

4.4 Conclusions

In this paper we investigated the behavior of different cluster algorithms for text clustering. Thereby we restricted ourself to such algorithms, which only take the distances between data into account but not the objects to be clustered itself. As distance measure we used the information distance. It can be concluded that, if texts from different languages are available (here 'Eurovoc'-documents), this language structure dominates the clustering, independent from the cluster algorithm. An overall moderate agreement between the different approaches is observed. Content specific clustering (separated in each language) is more difficult. The overall agreement is good as well but with instable results for the different languages depending on the approaches. Here, AP and RNG (with curtailments) show the most reliable results. The content specific discrimination ability is also verified for a text data base of psychotherapy session transcripts, which can be related to different therapy phases. This phase structure is nicely verified by the text clustering by almost all cluster algorithms.[2]

References

[2] Acknowledgement: The authors would like to thank Matthias Ongyerth for his contributions in early experiments on AP and SC.