Günter, Kruse, Neumann (Eds.): Advances in Artificial Intelligence.
LNAI 2821, pp. 254-266, ISBN 3-540-20059-2, ©Springer 2003.

Automatic Document Categorization

Interpreting the Perfomance of Clustering Algorithms

Benno Stein and Sven Meyer zu Eissen
stein@upb.de, smze@upb.de

Paderborn University
Department of Computer Science
D-33095 Paderborn, Germany

Abstract Clustering a document collection is the current approach to automatically derive underlying document categories. The categorization performance of a document clustering algorithm can be captured by the F-Measure, which quantifies how close a human-defined categorization has been resembled.
However, a bad F-Measure value tells us nothing about the reason why a clustering algorithm performs poorly. Among several possible explanations the most interesting question is the following: Are the implicit assumptions of the clustering algorithm admissible with respect to a document categorization task?

Though the use of clustering algorithms for document categorization is widely accepted, no foundation or rationale has been stated for this admissibility question. The paper in hand is devoted to this gap. It presents considerations and a measure to quantify the sensibility of a clustering process with regard to geometric distortions of the data space. Along with the method of multidimensional scaling, this measure provides an instrument for accessing a clustering algorithm's adequacy.

Key words: Document Categorization, Clustering, F-Measure, Multidimensional Scaling, Information Visualization

1 Introduction

Clustering is a key concept in automatic document categorization and means grouping together texts with similar topics [5, 42]. It can serve several purposes:

1. Enhance the retrieval performance in terms of query relevance [13].
2. Enhance the retrieval performance in terms of response time [13].
3. Improve the user interface by facilitating navigation, inspection, and organization of document collections.
4. Automate text generation by providing the basis for a further processing like summarization.

Document clustering is a collective term for a complex data processing procedure that includes several model formation tasks: elimination of stop words, application of stemming algorithms, syntactic indexing based on term frequencies, semantic indexing based on term document correlations, or computation of similarity matrices [28].

For each of these tasks exist different approaches and several parameters, and different clustering algorithms behave differently sensitive to a concrete document representation model.

The various number of published experiments give an idea of what clustering algorithms can afford with respect to document categorization—but a justification, or a clear intuition why an algorithm performs well or poorly in a particular setting is hardly presented. This is in the nature of things: Aside from their computational complexity, clustering algorithms are primarily assessed by geometrical properties of the data space they are working on.

The categorization performance of a clustering algorithm can be quantified, for instance by the F-Measure. In a successful situation, say, for a high F-Measure value, one can argue that the chosen algorithm is adequate; with a bad F-Measure value, however, the following questions come up:

- Are the geometrical assumptions of the clustering algorithm admissible for the data at hand?
- Does noisy data disguise the underlying category structure?
- Is there an underlying structure at all?

These and similar questions can be answered easily by visual inspection.

1.1 Contributions of the Paper

The contributions of this paper are based on the hypothesis that a visual analysis of the data space is indispensable to understand the behavior of a clustering algorithm.

Note that visualizing a document collection is bound up with the question of dimensionality: The interesting data are documents which are abstracted towards feature vectors of several thousand dimensions. To become interpretable for a human beholder the objects must be embedded in a two or three-dimensional space. I. e., the document similarity, whose computation is based on all features in the original term space, has to be resembled by the geometrical object distance in the Euclidean embedding space. Although the singular value decomposition of large term document matrices shows that this reduction in degrees of freedom appears more drastic than it really is, the embedding implies a noticeable distortion of document similarities.

Stress measures are used to judge this distortion in the embedding space. Nevertheless, they tell us only little about the impact of the distortion with respect to a clustering algorithm. This is where the paper sets in. It contrasts an algorithm's clustering behavior in the high-dimensional term space and the low-dimensional Euclidean space and quantifies the degree of clustering coherence by a so-called "relative F-Measure". This measure can be regarded as a stress measure from the clustering algorithm perspective. In particular the paper shows

1. that a high embedding stress may or may not affect the performance of a clustering algorithm, and
2. how the relative F-Measure is used as a tool to interpret visualizations of the data space.

Moreover, the relative F-Measure can be used to compare clustering algorithms with respect to their robustness against geometric distortion.

2 Related Work and Background

Document clustering is a popular subject of research, and there are many publications on this topic dealing with performance experiments. E. g., Yang and Pedersen present a comparative study of feature selection methods [41], Salton presents a comparative study of term weighting models [33], and Aggarwal investigates the correlation between dimension reduction and Nearest Neighbor search [1]. The majority of the investigations employ a standard document representation method and analyze the categorization performance with respect to different clustering algorithms (strategies) [12, 42, 11, 37, 4].

Some of the recent results are quoted in the next subsection, while the subsections 2.2 and 2.3 are devoted to approaches for document space visualization.

2.1 Document Clustering

Let D be a set of objects each of which representing a document. An element $d \in D$ comprises a parsimonious but significant vector of term-number-pairs that characterize the document associated with d. Often, the terms in d are counted according to the approved $tf \cdot idf$-scheme, and the similarity computation between each two elements in D follows the cosine-measure. The construction of D from a document collection is called indexing and is not treated in this place; details can be found in [33].

An exclusive clustering of a set of documents D is a collection $\mathcal{C}$ of disjoint sets with $\bigcup_{C_i \in \mathcal{C}} C_i = D$. Most clustering algorithms can be assigned to one of the following classes.

- Iterative algorithms, which strive for a successive improvement of an existing clustering, such as k-Means, k-Medoid, Kohonen, or Fuzzy-k-Means [25, 19, 20, 40].
- Hierarchical algorithms, which create a tree of node subsets by successively merging or subdividing the objects, such as k-Nearest-neighbor, linkage, Ward, or Min-cut methods [10, 34, 16, 24, 39].
- Density-based algorithms, which separate a similarity graph into subgraphs of high connectivity values, such as DBSCAN, MAJORCLUST, or CHAMELEON [36, 8, 18].
- Meta-search algorithms, which treat clustering as a generic optimization task where a given goal criterion is to be minimized [3, 30, 31, 30].

The runtime of iterative algorithms is $\mathcal{O}(nkl)$, where n, k and l designate the number of documents, clusters, and necessary iterations to achieve convergence. Hierarchical algorithms construct a complete similarity graph, which results in $\mathcal{O}(n^2)$ runtime. When applied to non-geometrical data, the runtime of density-based algorithms is in the magnitude of hierarchical algorithms or higher.

The different clustering algorithms are differently sensitive with respect to noisy data, outliers, cluster dilations, non-convex cluster shapes, etc., and different statements can be found in the literature on this subject. Altogether, a trade-off can be observed between an algorithm's runtime complexity and its capability to detect clusters of complex shape. k-Means [25], for example, provides a simple mechanism for minimizing the sum of squared errors with k clusters. Moreover, aside from its efficiency, it provides a robust behavior, say, it rarely fails completely with respect to the quality of the

Table 1. Characterization of selected clustering algorithms with respect to geometrical and geometry-related properties [2, 18, 19].

Agglomeration characteristic	
dilative (cluster number over size)	Complete Link
contractive (cluster size over number)	Single Link
conservative (balanced behavior)	Group Average Link, k-Means, variance-based methods (Ward)
Type of detected clusters	(+) good in (o) to some extent (–) unqualified
spherical clusters	k-Means (+), k-Medoid (+), Group Average Link (o),
arbitrarily shaped	Single Link (+), k-Means (–), k-Medoid (–), DBSCAN (+)
small clusters	Group Average Link (o)
equally sized	Ward (+), Complete Link (+)
different density	CHAMELEON (+), MAJORCLUST (+)

clusters found. On the other hand, k-Means is not able to identify clusters that deviate much from a spherical shape. Table 1 lists typical geometrical properties of prominent clustering algorithms.

2.2 Document Space Visualization

Document space visualization is a collective term for approaches that prepare relations within a document collection D in order to make D amenable for human visual inspection. The result of such a preparation may be a thematic landscape, an information terrain, a rendered text surface, a hyperbolic graph layout, or an interaction graph [6, 7, 9, 17, 21, 26, 27, 29, 32, 35, 38].

Since documents are represented by high-dimensional term vectors, many visualization approaches employ a dimension reduction like multidimensional scaling (MDS) along with a cluster analysis as essential preprocessing steps when rendering D. To the interplay of these document abstraction steps, however, only less attention is payed.

A clustering in the reduced two- or three-dimensional embedding space may deviate significantly from a clustering in the original n-dimensional document space, $n \in [1000..5000]$. The developers of the WebRat visualization system "solve" this problem by performing the cluster analysis in the embedding space: "*Our algorithm returns labels for the clusters the user sees.*" [32]. This strategy obviously entails the risk of pretending clusters that may not exist in the original data; Figure 3 in Section 4 gives an example for such a situation (which is easily uncovered by computing the relative F-measure).

Navarro also reports on the sensitivity of the MDS with respect to the quality (= separability) of a document collection [27]: "*. . . MDS produces very good visualizations for higher quality data, but very poor visualizations of lower quality data.*"

Other visualization approaches rate cluster quality over layout quality. I.e., the cluster analysis is performed in the original space, while the visualization capabilities are

limited to an approximate, say locally reasonable cluster placement. The systems BiblioMapper, Blobby-Texts, and AISEARCH follow this paradigm [35, 29, 26].

Note that the impact of an MDS on the clustering performance in the embedding space can hardly be judged by a classical MDS stress value: The typical size of the visualized collection D, $|D| \in [100..1000]$, along with the drastic dimension reduction results in stress values that exceed the worst case of Kruskal's application-independent rules of thumb in the very most cases [22].

2.3 Multidimensional Scaling

Multidimensional scaling (MDS) is a class of techniques for the analysis of dissimilarity data. MDS is used to find representations in $\mathbf{R}^k$ of objects for which only pairwise dissimilarities are given. The dissimilarities need not to be correct in the sense of a metric—they may be biased or estimated by humans. MDS techniques aim to reflect the inter-object dissimilarities through the distances of their representatives in $\mathbf{R}^k$ as good as possible. For $k = 2$ or $k = 3$ the representation can serve to visualize the objects. In general, the k real values per object can be considered as abstract features that can be used for further analysis.

Stress functions are used to measure how good a set of representatives $x_1, \ldots, x_n \in \mathbf{R}^k$ approximates the dissimilarities $\delta_{i,j}$ of the embedded objects. A candidate stress function is the residual sum of squares

$$S(x_1, \ldots, x_n) = \left(\sum_{i<j} \left(\delta_{i,j} - d_{i,j}\right)^2 \right)^{1/2}$$

where $d_{i,j} = ||x_i - x_j||$ is the distance between x_i and x_j measured by a norm which is induced by an arbitrary metric on $\mathbf{R}^k$: High stress values are interpreted as a high degree of misfit of the representation.

Note that this stress function has several drawbacks. First, the stress value of an arbitrary representation is not normalized and hence is not comparable to stress values of other representations. Second, the relation between $\delta_{i,j}$ and $d_{i,j}$ is absolute; for the underlying model it might be useful or even necessary to relate the dissimilarities to the distances by a function. As a consequence, transformed dissimilarities, called disparities $\hat{\delta}_{i,j} = f(\delta_{i,j})$ are used, where the function f is derived from the underlying model. Note that f can be used to map ordinal data onto real values. In this case we speak of ordinal or non-metric MDS. To eliminate the mentioned drawbacks, stress measures called Stress-1 and Stress-2 are used [22], which are given by

$$S_1(x_1, \ldots, x_n) = \left(\frac{\sum_{i<j} \left(\hat{\delta}_{i,j} - d_{i,j}\right)^2}{\sum_{i<j} d_{i,j}^2} \right)^{1/2}$$

and

$$S_2(x_1, \ldots, x_n) = \left(\frac{\sum_{i<j} \left(\hat{\delta}_{i,j} - d_{i,j}\right)^2}{\sum_{i<j} (d_{i,j} - \bar{d})^2} \right)^{1/2}$$

where $\bar{d}$ is the average value of all distances $d_{i,j}$. Aside from measuring the goodness of fit, stress functions are employed as optimization criterion for MDS algorithms. A question which remains open is up to which stress value a representation can be judged valid with respect to the disparities. Kruskal provided rules of thumb for Stress-1 values, which are listed in Table 2.

Table 2. Kruskal's rules of thumb for the goodness of fit with respect to a stress value.

Stress	Goodness of fit
0	perfect
0.025	excellent
0.05	good
0.1	fair
0.2	poor

3 Interpreting Clustering Performance

The categorization performance of a clustering algorithm can be analyzed with external, internal, or relative measures [15]. External measures use statistical tests in order to quantify how well a clustering matches the underlying structure of the data. In absence of an external judgment, internal clustering quality measures must be used to quantify the validity of a clustering. Relative measures can be derived from internal measures by evaluating different clusterings and comparing their scores [19].

In our context, the underlying structure is the known categorization of a document collection as provided by a human editor, and external measures can be used.

3.1 The F-Measure

The F-Measure quantifies how well a clustering matches a reference partitioning of the same data; it hence is an external validity measure. The F-Measure combines the precision and recall ideas from information retrieval [23] and constitutes a well-accepted and commonly used quality measure for automatically generated document clusterings.

Let D represent the set of documents and let $\mathcal{C} = \{C_1, \ldots, C_k\}$ be a clustering of D. Moreover, let $\mathcal{C}^* = \{C_1^*, \ldots, C_l^*\}$ designate the reference partitioning. Then the recall of cluster j with respect to partition i, $rec(i,j)$, is defined as $|C_j \cap C_i^*|/|C_i^*|$. The precision of cluster j with respect to partition i, $prec(i,j)$, is defined as $|C_j \cap C_i^*|/|C_j|$. The F-Measure combines both values as follows:

$$F_{i,j} = \frac{2}{\frac{1}{prec(i,j)} + \frac{1}{rec(i,j)}}$$

Based on this formula, the overall F-Measure of a clustering $\mathcal{C}$ is:

$$F = \sum_{i=1}^{l} \frac{|C_i^*|}{|D|} \cdot \max_{j=1,\ldots,k} \{F_{i,j}\}$$

A perfect clustering matches the given partitioning exactly and leads to an F-Measure value of 1. In Figure 1 (left hand side) a cluster with a high precision and a low recall value is shown. Note that although the precision value is close to the maximum value of 1, the F-Measure value is rather low at 0.4. A high F-Measure value can only be achieved if both precision and recall are high, as exemplary shown in Figure 1 on the right hand side.

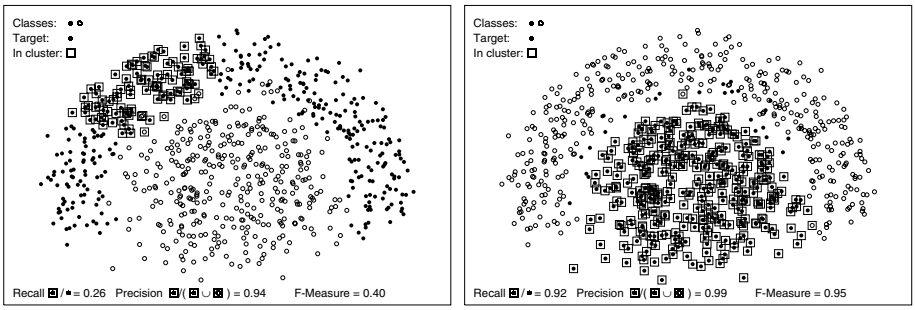

Figure 1. A cluster with high precision and low recall values (left), and a cluster with both high precision and high recall values (right).

3.2 The Relative F-Measure

Let $\mathcal{A}_1, \ldots, \mathcal{A}_{t_A}$ be the sequence of clusterings generated by a clustering algorithm in the original space, and let $\mathcal{B}_1, \ldots, \mathcal{B}_{t_B}$ be the sequence of clusterings generated by the same clustering algorithm in the embedding space. To measure the effect of the MDS distortion on the clustering process we compare the clusterings $\mathcal{A}_i$ to the corresponding clusterings $\mathcal{B}_j$, where $\mathcal{A}_1$ corresponds to $\mathcal{B}_1$, $\mathcal{A}_{t_A}$ corresponds to $\mathcal{B}_{t_B}$ and an equidistant mapping is applied in between.

For each pair of clusterings $\mathcal{A}_i$, $\mathcal{B}_j$, the clustering $\mathcal{A}_i$ defines the reference classification to which the clustering $\mathcal{B}_j$ is compared using the F-Measure. We denote the resulting value as the *relative* F-Measure value. If a clustering algorithm behaves identically in both the original space and the embedding space, the relative F-Measure value will always be 1. Typically, the comparison delivers a curve that starts with 1 and that oscillates between $1/k$ and 1, depending on the MDS distortion and the sensitivity of the clustering algorithm.[1] Note that if the relative F-Measure is high at the end of the clustering process, the algorithm found a clustering in the embedding space that is similar

[1] If k defines an upper bound for the number of clusters, $1/k$ defines a lower bound of possible F-Measure values.

to the clustering found in the original space. I. e., the distortion error of the embedding is without impact on the clustering algorithm, and we can use the MDS projection for visual inspection.

4 Illustration

This section illustrates the presented ideas with a collection D based on 800 objects. With respect to feature number (dimension), feature distribution, and object similarity each element in D resembles a document which falls into one of two classes. [2]

Figure 1, presented already in the previous section, shows the MDS projection of D in the two-dimensional space. This embedding has an error (stress value) of 31%, which is typical for embeddings of mid-sized document collections D, and which is not acceptable according to Kruskal's rules of thumb.

To get an idea of how the MDS distortion influences the performance of k-Means, MAJORCLUST, Single Link, and Group Average Link, we clustered the points in the original space as well as in the embedding space and computed the relative F-Measure in each clustering step. For convenience, the figures depict also the F-Measure curves for both the original and the embedded data in each clustering step.

4.1 k-Means Clustering

Figure 2 (top) shows the development of the relative F-Measure during the k-Means clustering process. The first part of the curve shows that cluster assignments differ noticeably for the original and the embedding space. This is a consequence of the MDS projection error—however, at the end of the clustering process the relative F-Measure value is high (0.9).

I. e., the MDS projected data in the embedding space appears to k-Means like the data in the original space, and, consequently, one can accept the found clusters: k-Means performs poorly here because the data contains entwined clusters.

4.2 Linkage-based Clustering

The behavior of Group Average Link (cf. Figure 3, top) differs substantially from k-Means. In the original space, a high F-Measure value is achieved (0.9), while the performance in the two-dimensional embedding space is poor. Consequently, the relative F-Measure at the end of the agglomeration process is also low (0.6).

Figure 3 (bottom left) shows the found cluster in the embedding space, but it cannot serve as a basis for an analysis of the performance of Group Average Link in the original space, since the relative F-Measure is low at the end of the clustering process, indicating that the algorithm gets misleaded by the MDS projection. To get an idea of the cluster quality in the original space, Figure 3 (bottom right) shows this cluster in the embedding space.

[2] Various experiments have been conducted with the new Reuters Corpus Volume 1, English Language `http://about.reuters.com/researchandstandards/corpus/`. This corpus contains about 34.000 single topic documents that fall into more than hundred classes. For illustration purposes only figures of the binary classification situation are shown in this section.

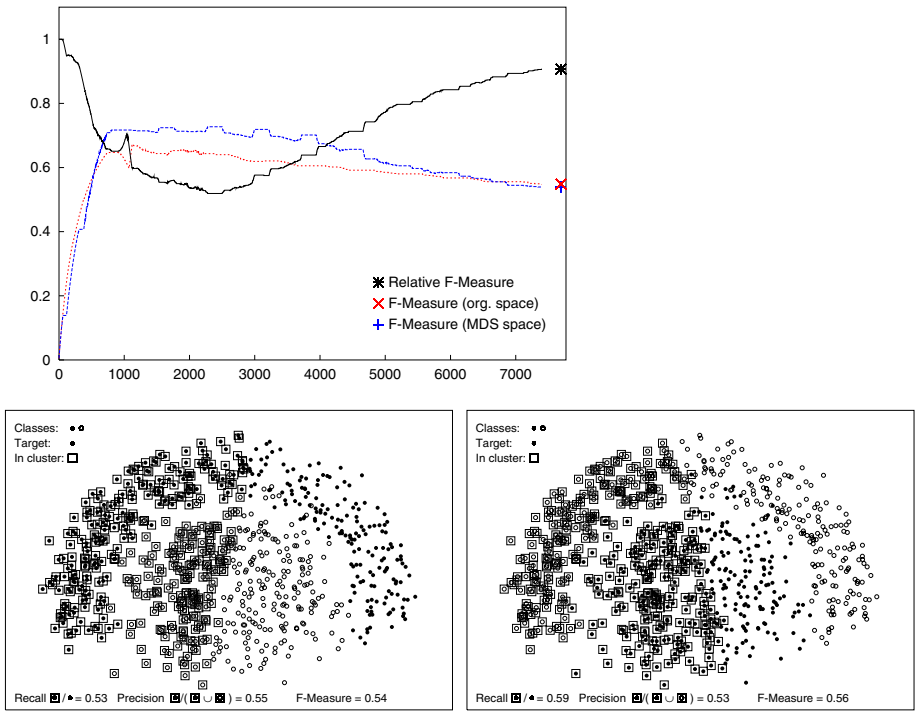

Figure 2. The development of the relative F-Measure during the k-Means clustering process (top), a cluster found by k-Means in the embedding space (bottom left), and the cluster found in the original space (bottom right). The x-axis and y-axis of the top figure displays the number iterations of k-Means and the F-Measure values respectively.

4.3 MajorClust Clustering

MAJORCLUST shows a good and robust F-Measure development during the clustering process (cf. Figure 4). Since the final relative F-Measure values are fairly high, the MDS projection can be used to analyze the performance of MAJORCLUST in the original space. The cluster which is found was already shown in Figure 1.

The reason why MAJORCLUST's F-Measure values in both of the spaces are not higher is that the algorithm assigns the remaining points to three clusters (cf. Figure 1), which lead to high precision but lower recall values and consequently to an overall F-Measure value of approximately 0.8.

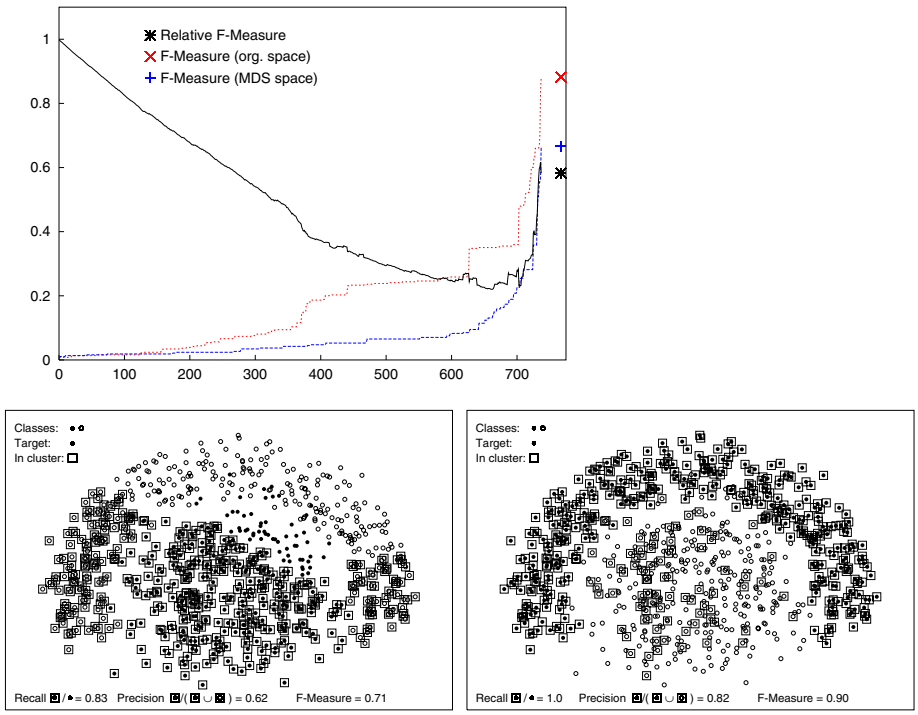

Figure 3. The development of the relative F-Measure during the Group Average Link clustering process (top), a cluster found by Group Average Link in the embedding space (bottom left), and the cluster found in the original space (bottom right). The x-axis and y-axis of the top figure displays the agglomeration level of Group Average Link and the F-Measure values respectively.

Discussion

As already pointed out by other authors, clustering algorithms often contain implicit assumptions about the clusters' shapes, sizes, or density distributions [14, cf. page 268]. While humans perform competitively with clustering algorithms in two dimensions, it is difficult to obtain an intuitive interpretation of data in high-dimensional spaces.

Documents are represented in the high-dimensional vector space model, and an embedding of the data for visual interpretation purposes is usually performed by multi-dimensional scaling. The stress values involved with an MDS are typically significantly above Kruskal's suggestions [22], which raises the question of interpretability of the resulting scatter plots.

The key idea of the paper in hand is the following: If a cluster algorithm behaves similar in both the original data space and the embedding space, then the latter is amenable to geometrical interpretation. For this purpose we have introduced the relative F-Measure which quantifies the coherence of two clusterings during the clustering process.

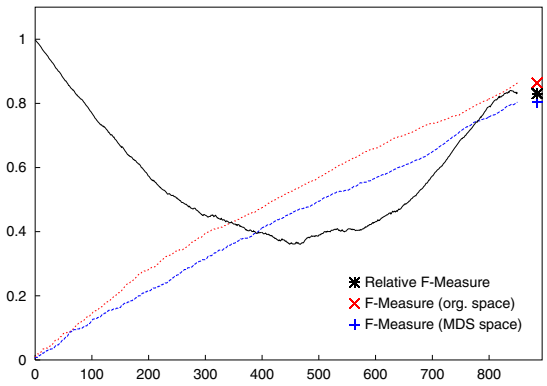

Figure 4. The F-Measure development during the MAJORCLUST clustering process. The x-axis and y-axis displays the number of iterations (node reassignments) of MAJORCLUST and the F-Measure values respectively.

We conducted several experiments in the field of automatic document categorization. It becomes clear that Kruskal's commonly accepted leveling rule for the interpretation of MDS stress values cannot be applied to detect an inadmissible distortion in the low-dimensional space. The relative F-Measure, however, provides a means to distinguish between admissible and inadmissible embeddings.

There is the question whether certain clustering algorithms behave more sensitive than others with respect to geometric distortion. Although our experiments included algorithms from each class mentioned in Subsection 2.1, they allow no final statement respecting distortion sensitivity or even a sensitivity-runtime tradeoff. This issue is subject of current research.

References

1. Charu C. Aggarwal. Hierarchical subspace sampling: a unified framework for high dimensional data reduction, selectivity estimation and nearest neighbor search. In *Proceedings of the ACM SIGMOD international conference on Management of data*, pages 452–463. ACM Press, 2002.
2. K. Backhaus, B. Erichson, W. Plinke, and R. Weiber. *Multivariate Anaylsemethoden*. Springer, 1996.
3. Thomas Bailey and John Cowles. Cluster Definition by the Optimization of Simple Measures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, September 1983.
4. Andrei Z. Broder, Steven C. Glassman, Mark S. Manasse, and Geoffrey Zweig. Syntactic Clustering of the Web. In *Selected papers from the sixth international conference on World Wide Web*, pages 1157–1166. Elsevier Science Publishers Ltd., 1997.
5. Heide Brücher, Gerhard Knolmayer, and Marc-Andre Mittermayer. Document classification methods for organizing explicit knowledge. In *Third European Conference on Organizational Knowledge, Learning, and Capabilities*, 2002.

6. A. Buja, D. F. Swayne, M. Littman, N. Dean, and H. Hofmann. XGvis: Interactive Data Visualization with Multidimensional Scaling. *Journal of Computational and Graphical Statistics*, 2001.
7. Matthew Chalmers. Using a landscape metaphor to represent a corpus of documents. In *Proc. European Conference on Spatial Information Theory*, volume 716 of *LNCS*, pages 377–390, 1993. URL `citeseer.nj.nec.com/chalmers93using.html`.
8. M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD96)*, 1996.
9. Sara Irina Fabrikant. Visualizing Region and Scale in Information Spaces. In *The 20th International Cartographic Conference*, pages 2522–2529, Beijing, China, August 2001.
10. K. Florek, J. Lukaszewiez, J. Perkal, H. Steinhaus, and S. Zubrzchi. Sur la liason et la division des points d'un ensemble fini. *Colloquium Methematicum*, 2, 1951.
11. Eui-Hong Han and George Karypis. Centroid-Based Document Classification: Analysis and Experimental Results. Technical Report 00-017, Univercity of Minnesota, Department of Computer Science / Army HPC Research Center, March 2000.
12. Taher H. Haveliwala, Aristides Gionis, Dan Klein, and Piotr Indyk. Evaluating strategies for similarity search on the web. In *Proceedings of the eleventh international conference on World Wide Web*, pages 432–442. ACM Press, 2002.
13. Makoto Iwayama and Takenobu Tokunaga. Cluster-based text categorization: a comparison of category search strategies. In Edward A. Fox, Peter Ingwersen, and Raya Fidel, editors, *Proceedings of SIGIR-95, 18th ACM International Conference on Research and Development in Information Retrieval*, pages 273–281, Seattle, USA, 1995. ACM Press, New York, US.
14. A. K. Jain, M. N. Murty, and P. J. Flynn. Data Clustering: a Review. *ACM Computing Surveys (CSUR)*, 31(3):264–323, 2000. ISSN 0360-0300.
15. Anil K. Jain and Richard C. Dubes. *Algorithm for Clustering in Data*. Prentice Hall, Englewood Cliffs, NJ, 1990. ISBN 0-13-022278-X.
16. S. C. Johnson. Hierarchical clustering schemes. *Psychometrika*, 32, 1967.
17. Eser Kandogan. Visualizing multi-dimensional clusters, trends, and outliers using star coordinates. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 107–116. ACM Press, 2001.
18. G. Karypis, E.-H. Han, and V. Kumar. Chameleon: A hierarchical clustering algorithm using dynamic modeling. Technical Report Paper No. 432, University of Minnesota, Minneapolis, 1999.
19. Leonard Kaufman and Peter J. Rousseuw. *Finding Groups in Data*. Wiley, 1990.
20. T. Kohonen. *Self Organization and Assoziative Memory*. Springer, 1990.
21. T. Kohonen, S. Kaski, K. Lagus, J. Salojrvi, J. Honkela, V. Paatero, and A. Saarela. Self organization of a massive document collection. In *IEEE Transactions on Neural Networks*, volume 11, may 2000. URL `citeseer.nj.nec.com/378852.html`.
22. J. B. Kruskal. Multidimensional Scaling by Optimizing Goodness of Fit to a Nonmetric Hypothesis. *Psychometrika, Vol. 29, No. 1,*, March 1964.
23. Bjornar Larsen and Chinatsu Aone. Fast and Effective Text Mining Using Linear-time Document Clustering. In *Proceedings of the KDD-99 Workshop San Diego USA*, San Diego, CA, USA, 1999.
24. Thomas Lengauer. *Combinatorical Algorithms for Integrated Circuit Layout*. Applicable Theory in Computer Science. Teubner-Wiley, 1990.
25. J. B. MacQueen. Some Methods for Classification and Analysis of Multivariate Observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, 1967.

26. Sven Meyer zu Eißen and Benno Stein. The AISEARCH Meta Search Engine Prototype. In Amit Basu and Soumitra Dutta, editors, *Proceedings of the 12th Workshop on Information Technology and Systems (WITS 02), Barcelona Spain*. Technical University of Barcelona, December 2002.
27. Daniel J. Navarro. Spatial Visualization of Document Similarity. Defence Human Factors Special Interest Group Meeting, 2001.
28. M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.
29. Randall M. Rohrer, David S. Ebert, and John L. Sibert. The Shape of Shakespeare: Visualizing Text using Implicit Surfaces. In *IEEE Symposium on Information Visualization*, pages 121–129, North Carolina, USA, October 1998.
30. Tom Roxborough and Arunabha. Graph Clustering using Multiway Ratio Cut. In Stephen North, editor, *Graph Drawing*, Lecture Notes in Computer Science, Springer, 1996.
31. Reinhard Sablowski and Arne Frick. Automatic Graph Clustering. In Stephan North, editor, *Graph Drawing*, Lecture Notes in Computer Science, Springer, 1996.
32. V. Sabol, W. Kienreich, M. Granitzer, J. Becker, K. Tochtermann, and K. Andrews. Applications of a Lightweight, Web-Based Retrieval, Clustering and Visualisation Framework. In *4th International Conference on Practical Aspects of Knowledge Management*, volume 2569 of *LNAI*, pages 359–368, 2002.
33. G. Salton. *Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by Computer*. Addison-Wesley, 1988.
34. P. H. A. Sneath. The application of computers to taxonomy. *J. Gen. Microbiol.*, 17, 1957.
35. Min Song. BiblioMapper: A Cluster-based Information Visualization Technique. In *IEEE Symposium on Information Visualization*, pages 130–136, North Carolina, USA, October 1998.
36. Benno Stein and Oliver Niggemann. *25. Workshop on Graph Theory*, chapter On the Nature of Structure and its Identification. Lecture Notes on Computer Science, LNCS. Springer, Ascona, Italy, July 1999.
37. Michael Steinbach, George Karypis, and Vipin Kumar. A comparison of document clustering techniques. Technical Report 00-034, Department of Computer Science and Egineering, University of Minnesota, 2000.
38. Edgar Weippl. Visualizing Content-based Relations in Texts. In *Proceedings of the 2nd Australasian conference on User interface*, pages 34–41. IEEE Computer Society Press, 2001. ISBN 0-7695-0969-X.
39. Zhenyu Wu and Richard Leahy. An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, November 1993.
40. J. T. Yan and P. Y. Hsiao. A fuzzy clustering algorithm for graph bisection. *Information Processing Letters*, 52, 1994.
41. Yiming Yang and Jan O. Pedersen. A comparative study on feature selection in text categorization. In Douglas H. Fisher, editor, *Proceedings of ICML-97, 14th International Conference on Machine Learning*, pages 412–420, Nashville, US, 1997. Morgan Kaufmann Publishers, San Francisco, US.
42. Ying Zaho and George Karypis. Criterion Functions for Document Clustering: Experiments and Analysis. Technical Report 01-40, Univercity of Minnesota, Department of Computer Science / Army HPC Research Center, Feb 2002.