Touch it, Mine it, View it, Shape it

Martin Hahmann, Dirk Habich, Wolfgang Lehner

TU Dresden; Database Technology Group; Dresden, Germany

Abstract: To benefit from the large amounts of data, gathered in more and more application domains, analysis techniques like clustering have become a necessity. As their application expands, a lot of unacquainted users come into contact with these techniques. Unfortunately, most clustering approaches are complex and/or scenario specific, which makes clustering a challenging domain to access. In this demonstration, we want to present a clustering process, that can be used in a hands-on way.

1 Introduction

Clustering is the partitioning of a set of objects into clusters [EKSX96, JMF99], so that similar objects are in the same cluster, while dissimilar objects are not. In order to create a clustering, an appropriate algorithm must be selected, parameterized and executed. The obtained result is evaluated and if necessary, algorithm and/or parameters are modified and the clustering is generated again. Each of these actions determines the course and outcome of the clustering process [JL05, JMF99]. Even so, user-support is lacking in practise, which made 'trial and error' a common approach to clustering, for users not familiar with the subject. Obviously, this often results in numerous iterations, unsatisfying results and eventually user frustration.

With *ensemble clustering*[GMT05, SG02], an alternative to single-algorithm clustering has been established. This approach creates multiple partitionings of a data set—the cluster ensemble—and aggregates them into one final clustering result. In doing so, quality and robustness are increased in comparison with single input clustering [GMT05, SG02]. Additionally, this procedure eases algorithm selection and parameterization. However, the overall resemblance to 'trial and error' remains, as unsatisfying aggregation results can only be adjusted by modifying the cluster-ensemble and repeating its creation and aggregation.

In our previous work, we have already adressed some of the described issues. In [HVR$^+$09] we proposed an extended aggregation algorithm, utilizing soft clustering input and allowing result adjustments by parameterization of the aggregation only. To enable user support, we introduced an interactive visualization to control our aggregation, assist with result interpretation and indicate appropriate result adjustments [HHL10b].

In our demonstration, we present a clustering process composed from this components and show how this easy-applicable process allows the step-by-step refinement of a clustering.

2 Process

In this section, we will outline the structure and components of our clustering process. This process incorporates an algorithmic platform, which covers selection and execution of algorithms and a visual-interactive interface, assisting the user during result evaluation and modification.

The already introduced ensemble-clustering concept, built the conceptual starting point in the development of our algorithmic platform. Besides positive effects on the clustering result, this method aids the user by reducing the emphasis on the identification of a single optimal algorithm/parameter combination. All existing aggregation approaches we examined, lacked controllability [HVR+09], thus result adjustments were only possible through modification of the input clusterings. Unfortunately this effectively nullifies the benefits regarding user support, since now a whole set of clusterings must be reconfigured. To overcome this issue, we proposed our enhanced *flexible clustering aggregation* concept [HVR+09], which extends the classic approach in three major areas. First, the aggregation input is enriched with additional information about object-cluster relations, by utilizing *soft clustering* algorithms [Bez81] to generate the cluster ensemble. To benefit from this gain in information, the core aggregation method was modified in a second expansion. Finally, these arrangements allowed the derivation of a scoring function and with it the implementation of a control mechanism for the clustering aggregation. With *flexible clustering aggregation* users can adjust results without touching the cluster-ensemble. The necessary parameters could be abstracted in a user-friendly way, so that clusterings are adjusted by *"merging"* and *"splitting"* clusters.

To support result interpretation and identification of appropriate adjustments, we developed a visualization concept that is tightly coupled to our algorithmic platform. Our approach *augur* [HHL10b] can be seen as a hybrid between the two major groups of data/clustering visualizations, which are: (i) data-driven and (ii) result-driven. The first group depicts all objects and dimensions of the data, resulting in incomprehensible presentations and information-overload, as datasets exceed a certain scale. In contrast, the second group is relatively scale-invariant since only analysis results are presented (e.g. a clustering can be depicted as bar chart showing relative cluster sizes). Unfortunately these visualizations often shows not enough information. The hybrid character of our approach is achieved by visualizing the result and its relations to data, which are already incorporated in the soft input of our aggregation.

In compliance with Shneiderman's mantra, *'overview first, zoom and filter, then details-on-demand'* [Shn96], our visualization features views for theses three levels of detail. Our overview acts as a visual entry point and shows basic characteristics of the clustering aggregate, like relative size and the distances between the prototypes (centroids) of all clusters. If the user identifies clusters of interest in the overview, e.g. two very close clusters, these can be selected individually to get more information regarding their composition and their relations to other clusters, thus performing *'zoom and filter'*. More detailed information concerning a cluster's internal similarity resp. composition are presented in the attribute view.

Figure 1: Touch it, Mine it, View it, Shape it

By combining the *flexible clustering aggregation* with our *augur* visualization we, devised our clustering process. The course of the process begins with the presentation of an initial clustering result to the user. Using *augur*, this result is interpreted and the parts that need adjustment are identified. Via an interactive component of the visualization, the users modifications are forwarded to the algorithmic platform. After they are applied to the clustering, the adjusted result is again presented using the augur visualization. With this procedure, users can refine clustering results in an iterative manner. A theoretic description of this process model and its components, e.g. the available user-feedback operations, are published in [HHL10a].

3 Demo Details

The demo at BTW comprises a detailed explanation of the necessary concepts and components of our process and its live demonstration. We are going to show how our visualization and interaction concepts can be used to conduct a visually-driven exploration of scientific data sets. Furthermore, we will prepare some application scenarios based on synthetic as well as real-world data-sets. Within these scenarios, we will illustrate the benefits of our iterative refinement approach with regard to its handling by users not familiar with the domain of clustering. Additionally, we want to use the BTW environment to discuss possible future developments for our employed aggregation algorithms and visualization concept with interested demo visitors.

4 Summary

In this paper, we introduced our hands-on clustering process, which offers inexperienced users an accessible way to generate a satisfying clustering. Execution and Parameterization are eased by the user-friendly character of our algorithmic platform. In tight coupling with this platform, our visual-interactive user-interface, supports the interpretation of clustering results by revealing characteristics of clusters as well as relations between them and the underlying data. This result- and relation-oriented approach offers assistance to the user during the identification of appropriate result modifications. In contrast to existing clustering procedures our approach allows the iterative refinement of a clustering.

References

[Bez81] James C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum, 1981.

[EKSX96] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *Proc. of KDD*, 1996.

[GMT05] Aristides Gionis, Heikki Mannila, and Panayiotis Tsaparas. Clustering Aggregation. In *Proc. of ICDE*, 2005.

[HHL10a] Martin Hahmann, Dirk Habich, and Wolfgang Lehner. Evolving Ensemble-Clustering to a Feedback-Driven Process. In *Proceedings of the IEEE ICDM Workshop on Visual Analytics and Knowledge Discovery (VAKD)*, 2010.

[HHL10b] Martin Hahmann, Dirk Habich, and Wolfgang Lehner. Visual Decision Support for Ensemble-Clustering. In *Proceedings of the 22nd International Conference on Scientific and Statistical Database Management (SSDBM)*, 2010. (to appear).

[HVR+09] Martin Hahmann, Peter Volk, Frank Rosenthal, Dirk Habich, and Wolfgang Lehner. How to Control Clustering Results? Flexible Clustering Aggregation. In *Advances in Intelligent Data Analysis VIII*, pages 59–70, 2009.

[JL05] Anil Jain and Martin Law. Data Clustering: A Users Dilemma. *Pattern Recognition and Machine Intelligence*, pages 1–10, 2005.

[JMF99] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Comput. Surv.*, 31(3), 1999.

[SG02] Alexander Strehl and Joydeep Ghosh. Cluster Ensembles — A Knowledge Reuse Framework for Combining Multiple Partitions. *Journal of Machine Learning Research*, 3, 2002.

[Shn96] Ben Shneiderman. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In *VL '96: Proceedings of the 1996 IEEE Symposium on Visual Languages*, page 336, Washington, DC, USA, 1996. IEEE Computer Society.