**MCExplorer: Interactive Exploration of Multiple (Subspace) Clustering Solutions**

Stephan Günnemann  Hardy Kremer  Ines Färber  Thomas Seidl

Data Management and Data Exploration Group
RWTH Aachen University, Germany
{guennemann, kremer, faerber, seidl}@cs.rwth-aachen.de

Abstract—Large amounts of data are ubiquitous today. Data mining methods like clustering were introduced to gain knowledge from these data. Recently, detection of multiple clusterings has become an active research area, where several alternative clustering solutions are generated for a single dataset. Each of the obtained clustering solutions is valid, of importance, and provides a different interpretation of the data. The key for knowledge extraction, however, is to learn how the different solutions are related to each other. This can be achieved by a comparison and analysis of the obtained clustering solutions.

We introduce our demo MCExplorer\(^1\), the first tool that allows for interactive exploration, browsing, and visualization of multiple clustering solutions on several granularities. MCExplorer is applicable to the output of both fullspace and subspace clustering approaches.

Keywords—multiple groupings, alternative clusterings, multi-view clustering, subspace clustering, interactive exploration

---

I. INTRODUCTION

In today's applications huge amounts of data are collected. Since a manual analysis of the data is virtually impossible, data mining methods like clustering try to extract interesting patterns such that meaningful interpretations can be drawn from these compact representations. In the past decades a multitude of clustering methods were developed to extract one ('the best') grouping out of the data. However, data is often multi-faceted: for a single dataset, multiple valid interpretations and thus different alternative groupings are possible. In a customer database, for example, customers can be grouped according to their buying patterns, e.g. their preferred articles, or according to their paying habits, e.g. if they pay their bills on time. Each of these valid views, also called concepts, provides different insights about the data. Accordingly, extracting these multiple or alternative clusterings is an active research area as recent publications in the field of fullspace clustering show [1], [2], [3], [4], [5].

The same holds for the research field of subspace clustering, where the observation of multiple valid groupings within the data is even more apparent: many subspace clustering algorithms implicitly detect multiple concepts by their very nature. Often, a single subspace clustering result corresponds to multiple views. An example is depicted in Fig. 1, where we can identify different groupings if only subsets of the dimensions are considered. The dimensions selected on the left allow a grouping of the persons based on the intuitive concept “health awareness” while another grouping based on the concept “enthusiasm for technology” is obtained with the dimensions on the right. While traditional subspace clustering techniques [6], [7] simply detect clusters in arbitrary subsets of the dimensions and concepts are only detected implicitly, some approaches [8], [9] explicitly utilize the idea of concepts to identify clusters in strongly differing subspaces. Recently, the tool CoDA [10] was introduced to determine concepts for the output of any subspace clustering method and to assign the detected subspace clusters to their corresponding concepts.

![Figure 1. Multiple (subspace) clustering solutions in a single database.](image-url)

Overall, the detection of multiple clustering solutions is an active research area for traditional clustering as well as for subspace clustering. However, all of the presented methods focus just on the extraction of different concepts and their clusters. So far, there is no possibility to compare and analyze these alternative solutions; this, however, is the key for meaningful knowledge extraction. Are the concepts similar to each other or do they provide novel and interesting patterns? Do structures of one concept occur also in another one? Is this redundant information? What are the similarities or differences between individual clusters of multiple concepts? With MCExplorer (Multiple Concepts Explorer) we present a tool to close this gap by interactive exploration, browsing, and visualization of alternative clustering solutions. Starting with an overview of the entire structure the user can progressively perform more
fine-grained comparisons of patterns to achieve an in-depth analysis. Focusing on the comparison of multiple concepts, and not the individual concepts, MCExplorer comprises a three-level process that covers the whole cycle of analyzing multiple clusterings:

- Exploration of concepts, to compare the multiple hidden groupings in the data.
- Exploration of clusters, to compare the identified clusters of different concepts.
- Exploration of elements, to compare the grouped objects of different clusters.

With MCExplorer the user can analyze alternative clustering solutions in an intuitive and interactive setting. Overall, MCExplorer supports the user in the knowledge extraction based on multiple valid groupings, completing the KDD process.

II. MCExplorer

In this section, we introduce MCExplorer that is integrated into the OpenSubspace [11], [12] and CoDA [10] framework, which add subspace clustering and multiple concept functionality to the well-known WEKA Data Mining Software. This framework provides MCExplorer with multiple groupings to be analyzed. Fig. 2 shows a screenshot of the framework with the MCExplorer integration.

The interactive exploration of MCExplorer is based on the Visual Exploration Paradigm [13]: Starting with an overview of all concepts, the user can navigate through the visualization of these patterns, and interesting concepts can be selected for a more detailed analysis. This detailed information can again be browsed and even more fine-grained information can be requested by the user. The comparison and analysis of multiple groupings, i.e. the coarsest level of the analysis, can be performed in the main window illustrated in Fig. 2, while the other two levels of MCExplorer are realized in child windows.

A. Exploring concepts

At the start of the analysis, the user gets an overview of all multiple concepts, as shown in the main window of MCExplorer in Fig. 2. In this overview, each concept is represented as a node. Formally, each concept $Concept_i$ is a set of (subspace) clusters $\{C_1, \ldots, C_m\} = Concept_i$ with $C_j = (O_j, S_j)$ describing the grouped objects $O_j$ and relevant dimensions $S_j$. Note that if a fullspace clustering is analyzed, the set $S_j$ is identical for each cluster and only the object grouping $O_j$ is important. MCExplorer enables the user to quickly assess the concept structure by visualizing the core properties of single concepts and the relationships between different concepts.

Core properties as the number of clustered objects and the average dimensionality are represented by the radius and color of a concept’s corresponding node. More informations can be retrieved, when the cursor is placed over single nodes.

Several aspects are visualized that reflect the relationships between the different concepts. Initially, the user has to select a concept to be the current one under consideration (centrally displayed in Fig. 2). The remaining concepts are circularly arranged around this concept based on their similarity: Concepts very similar to the selected one are located near to the center while dissimilar concepts are at the border of the plot. Simultaneously, MCExplorer arranges the surrounding concepts along the circular lines such that similar concepts are adjacent. Technically, this is achieved by maximizing the pairwise similarity between adjacent concepts. When the user selects another concept as the central one, an automatic rearrangement of the remaining concepts is performed. This visual approach enables the user to browse through the concept-structure and groups of (dis-)similar concepts can intuitively be captured. At each time, the user can select individual concepts to obtain a more detailed comparison between these; this detailed analysis of two concepts is described in Sec. II-B. Formally, the similarity between the concepts is determined by the CE measure [14], which is applicable for both subspace and fullspace clustering solutions.

Each node in the concept structure is enriched with additional information with respect to the central concept. This enables a different kind of comparison in contrast to the similarity described above. More concretely, the pie-charts in the surrounding nodes indicate the degree to which a node’s concept can be explained by using the central concept (the pie-chart of the central concept is explained later). If only
few objects of the concept are also grouped in the same way in the central concept, the highlighted sector is small. If the sector is large, a redundancy of the concept is indicated, because the clusters of a concept can also be detected by the central concept. Formally, the fraction of a concept $X$ that can be explained by another concept $Y$ is determined via

$$explain(X|Y) = \frac{\sum_{C_j \in X} \max_{C_k \in Y} \{ |O_j \cap O_k| \cdot |S_j \cap S_k| \}}{\sum_{C_j \in X} \{ |O_j| \cdot |S_j| \}}$$

and hence the size of the sector for a surrounding concept $Con$ given the central concept $Central$ is $explain(Con|Central)$. In contrast to the similarity between concepts, this information is non-symmetric. Therefore, $MCEXplorer$ visualizes the reverse property $explain(Central|Con)$ with the strength/weight of the edge connecting these two nodes, i.e., the edge weights indicate the degree to which the central concept can be explained by the other concepts. For the pie-chart of the central concept, we select the surrounding concept which explains most parts of the central one, indicating the highest degree of redundancy. Accordingly, the edge with the highest weight and thus the value $\max_{X \in Concepts} \{explain(Central|X)\}$ determines the size of the sector, where $Concepts$ is the set of all surrounding concepts.

The last feature on the level of exploring concepts allows the user to separately visualize the concepts just on the object information, just on the dimension information, or on both information together. Since relevant dimensions only occur in subspace clusterings, this visualization is constrained to inputs from subspace clustering algorithms. The obtainable knowledge is essential for the user, because two concepts can comprise identical object groupings but in completely different dimension sets. In $MCEXplorer$, the user is able to choose which information to use for determining the similarity between the concepts and for the pie-chart calculation (cf. Fig. 2 top right corner). Overall, three different plots for the same central concept are possible and can be compared. However, since the arrangement of nodes is dynamically adjusted based on the similarities, different plots can show different layouts, which hinders an easy interpretation. To enable a visual comparison of these different plots, $MCEXplorer$ integrates a synchronization of the layouts. Thus, in $MCEXplorer$ a single plot can be selected as the source, while the others are automatically synchronized, i.e., the central concept as well as the arrangement of adjacent nodes are taken over from the source plot.

### B. Exploring clusters

In the first level of $MCEXplorer$, the user can obtain an overview of all concepts, and by selecting any two concepts within the plots a more detailed analysis can be performed. This analysis is done in the second level of $MCEXplorer$, where the actual clusters of the selected concepts are compared as illustrated in Fig. 3. If in the previous step two concepts are identified as similar, the user is now able to identify the causative clusters for this effect. For example a cluster of one concept can be split up in smaller ones in another concept. To visually compare clusters, $MCEXplorer$ uses again an intuitive representation: clusters are represented by nodes where size and dimensionality are reflected by the node’s radius and color. A horizontal ordering of the clusters is performed, such that similar clusters are placed in the same regions of the plot. This ordering is based on all clusters simultaneously and hence the user can easily compare the whole clustering structure of the two concepts. Redundancy between the clusters is indicated by a very close grouping of many clusters. The similarity between the clusters is formally defined by their overlapping elements and the ordering of clusters is obtained by minimizing the total weighted crossings in weighted bipartite graphs [15].

To increase the expressiveness of the visualization, the nodes are again enriched by pie-charts. As default, the visualized sector for each cluster indicates the overlap to its most similar cluster in the other concept (light shaded sectors in Fig. 3). For example, a cluster completely contained in another cluster of the other concept is enriched by a sector covering the whole node. This representation is performed for each cluster of both concepts. With the help of this representation several aspects, for example a cluster split up, can easily be detected. Furthermore, by selecting an individual cluster its specific overlap with all clusters of the other concept can be analyzed (dark shaded sectors in Fig. 3). Analogously to the first level, the user can analyze objects and dimensions simultaneously or restrict the analysis to just one property.

Overall, the second level of $MCEXplorer$ enables the user to get an overview of the clustering structure of certain concepts and to understand the reasons for the similarity and dissimilarity of the detected clustering solutions.
C. Exploring elements

The finest analysis in the comparison of multiple concepts can be done in the last level of MCExplorer: the individual elements of clusters can be compared by the user. By selecting clusters in the previous level, the user is guided to this level as illustrated in Fig. 4. MCExplorer visualizes the data matrix and the embedded clusters. Each row corresponds to one object of the database and each column to one dimension. The elements contained in the selected clusters of the first concept are highlighted by rectangular regions within the matrix while the elements covered by clusters of the second concept are highlighted by circles. Overlapping and non-overlapping elements can thus visually be inspected by the user. To facilitate this impression, MCExplorer permutes the objects and dimensions, such that the individual highlighted regions are mostly connected. Thus, outstanding elements which occur in several concepts can be identified. For a further in-depth analysis of such objects, the user is able to select arbitrary sets of elements in Fig. 4 and their characteristics, as the value-distributions within the dimensions, can be illustrated.

Overall, MCExplorer provides the user with the opportunity to compare multiple groupings in an interactive setting. Based on a three level process, the user is able to browse from an overview to an in-depth analysis: whole concepts, their corresponding clusters, and their individual elements can be compared and their properties can be analyzed. Besides the raw patterns determined by the existing methods, MCExplorer enables the user to infer actual knowledge based on the given clustering solutions.

![Figure 4](image_url)

**Figure 4.** Comparison of concepts on the object level.

III. Demonstration Scenario

The demonstration setup for MCExplorer enables the conference attendees to analyze the multi-faceted nature of several real world datasets. We primary provide data available at the UCI archive [16]. Different clustering methods for determine multiple groupings, either subspace or full-space based, can be applied by the participants on these datasets. The attendees may explore these results by browsing through the identified concepts and they can test the different levels of MCExplorer to get an in-depth understanding of the detected patterns. The soundness of the clustering results for different algorithms can thus be verified by the participants. Overall, MCExplorer supports the user to compare and analyze the multiple groupings hidden in today’s data.

**Acknowledgments.** This work has been supported by the UMIC Research Centre, RWTH Aachen University, Germany.

**References**