An Evaluation Framework for Temporal Subspace Clustering Approaches

Hardy Kremer\textsuperscript{\textcopyright}  Stephan Günemann\textsuperscript{*}  Arne Held\textsuperscript{\circledast}  Thomas Seidl\textsuperscript{\textcopyright}
\textsuperscript{\textcopyright}RWTH Aachen University, Germany  \textsuperscript{*}Carnegie Mellon University, USA
\{lastname\}@cs.rwth-aachen.de  sguennem@cs.cmu.edu

Abstract—Mining multivariate time series data by clustering is an important research topic. Time series can be clustered by standard approaches like k-means, or by advanced methods such as subspace clustering and triclustering. A problem with these new methods is the lack of a general evaluation scheme that can be used by researchers to understand and compare the algorithms; publications on new algorithms mostly use different datasets and evaluation measures in their experiments, making comparisons with other algorithms rather unfair.

In this demonstration, we present our ongoing work on an experimental framework that offers the means for extensive visualization and evaluation of time series clustering algorithms. It includes a multitude of methods from different clustering paradigms such as fullspace clustering, subspace clustering, and triclustering. It provides a flexible data generator that can simulate different scenarios, especially for temporal subspace clustering. It offers external evaluation measures and visualization features that allow for effective analysis and better understanding of the obtained clusterings. Our demonstration system is available on our website.

Keywords—time series clustering, multivariate time series, clustering, subspace clustering, visualization, evaluation

I. INTRODUCTION

Clustering of temporal data is a major area of data mining research [1], [2], [3], [4]. Temporal data reflect the changing state of an observed system over time. Examples are financial ratios, patient monitoring, audio data, and climate simulation models.

Clustering approaches mine for unknown patterns in temporal databases by grouping time series based on their similarity. Multivariate (as well as univariate) time series belong to the class of high dimensional data. Mining high dimensional data is challenging, since not all of the dimensions of a data object are relevant for the mining task at hand. Accordingly, methods were developed that analyze substructures of the data. Triclustering approaches [5], [6] tackle the challenge of irrelevant dimensions for multivariate time series data. These approaches can be considered as extensions of subspace clustering [7] to the temporal domain. In [8] and [9], we introduced approaches that go beyond triclustering. They are based on the idea of individual sets of relevant intervals (i.e., subsequences in which clustered time series are similar) in each dimension; this idea differs to triclustering, where the relevant intervals hold for all objects and dimensions of a cluster. Moreover, our approach in [8] handles misaligned time series by adaptively shifting time series in the time domain, and it achieves robustness to measurement errors by allowing certain fractions of deviating values in each relevant point in time.

While the community has introduced multiple clustering approaches for time series subspace clustering, comparing these different methods is challenging. Currently, there is no common baseline for generating synthetic data nor for evaluating the results. For the mining tasks of stream clustering and subspace clustering, powerful toolkits [10], [11], [12], [13] have been proposed that offer capabilities for data generation, evaluation, and visualization of the results. None of the frameworks, however, provides tools for an effective analysis of temporal subspace clustering methods.

In this demonstration, we present our ongoing work on our evaluation framework for temporal subspace clustering methods. It originates from the development of our approaches published in [9], [8] to allow for an effective comparison with the competing approaches.

In the following section, we will give an overview over the capabilities of our system.

II. FEATURES OF OUR SYSTEM

Our system is divided into three components, each of which is represented by a separate tab in our framework (cf. Fig. 1): A powerful data generator for time series data, a clustering component in which different clustering algorithms can be applied to the generated and real-world data, and an experiments component for running batches of experiments under a multitude of settings. In the following, we will concentrate on the first two components.

A. Time Series Generator

The time series generator component is the foundation of our framework (Fig. 1 shows a screenshot). It generates a dataset of time series together with a grouping of these time series, termed the ground truth clustering; each cluster corresponds to a set of time series and metadata describing the relevant intervals in each dimension. In the screenshot, for example, the visualized ground truth cluster has a relevant interval in the 8th dimension. In the clustering tab (described in Sec. II-B) the clustering results of the analyzed algorithms are evaluated with respect to this ground truth clustering. The data generator is highly parametrizable with regard to a multitude of aspects. Besides general parameters...
as the number of clusters, the cluster size, and the number of dimensions, users can specify aspects as the noise in the data, the variance of the time series in a single cluster, the minimal and the maximal length of relevant intervals. It is possible to generate noise points inside of relevant intervals, and clusters can be generated in which the obtained time series are slightly shifted with respect to each other. Our generator is not restricted to non-overlapping clusterings: if desired, a time series can belong to several clusters, resulting in different relevant intervals in each of these clusters.

The visualization view of the time series generator is also shown in the screenshot; it allows for an in-depth look at each single cluster. The view supports zooming and also storing the current view as a screenshot. Fig. 2 shows the same view with a dataset in which we allowed a misalignment (shift) of a cluster’s time series to each other. Such a scenario could, for example, be caused by out-of-sync sensors.

For a better overview over the relevant dimensions of single clusters, we introduced a second kind of view, shown in Fig. 3. This visualization abstracts from the time series of a cluster and only illustrates the relevant intervals of all dimensions simultaneously.

B. Clustering Component

In the clustering component of our framework, different algorithms can be applied to the generated data. Fig. 4 shows a screenshot of the corresponding tab; algorithms and an optional custom dataset (e.g., real world data) can be selected in the upper left corner. We provide algorithms from three paradigms: fullspace clustering (k-Means with...
Figure 3. Data Generator Tab: Visualization of the relevant dimensions of a ground truth time series cluster over all dimensions. Each line corresponds to a different dimension (in this case, there are 10 dimensions).

statistical features [4], kMedoid with Longest Common Subsequences as distance measure [14]), subspace clustering (e.g., PROCLUS [15], MineClus [16]), and triclustering (MIC [6], TimeSC [9], RTSC [8]).

After clustering, external measures (CE, E4SC, F1, cf. [17]) and other metrics (correctly found time series, etc.) are used to evaluate the found clusterings with respect to the corresponding ground truth (Fig. 4, upper right: Clustering Results). Besides the aggregated view on the whole clustering, we also provide these metrics for each of the individual clusters (Fig. 4: Cluster Results). Combined, they give a good overview of the clustering quality obtainable by the analyzed algorithms in the chosen scenario. By using such a benchmark scenario (i.e., the same dataset and the same evaluation measures), we perform an effective and fair comparison of different algorithms. A better understanding of the algorithms is obtained by systematically varying parameters in the data generator and analyzing the effects on the evaluation measures.

Finally, the clusterings are visualized. Basically, we use the same principles as in the data generator tab, but we also provide additional information based on the comparison with the ground truth, allowing for a better understanding of the clusterings. Concretely, for each of the found clusters, we can toggle (bottom of Fig. 4) whether we want to see the correctly assigned time series, the incorrectly assigned ones, the missing ones, or a combination of these possibilities. In Fig. 5, for example, we show the same cluster as in Fig. 4, but instead of the incorrectly assigned time series (red) we display the missing ones (blue). The shown clusterings were obtained by kMeans with statistical features, i.e., a fullspace approach. Fig. 6 shows the result of a temporal subspace clustering algorithm; the relevant intervals are highlighted in dark green (as in the generator tab).

Figure 4. Clustering Tab. On the top left, one can select from a multitude of clustering algorithms. After performing a clustering, the results of the external evaluation measures (w.r.t. the ground truth) are shown on the right. On the bottom, the found clusters are visualized. Note that it is possible to deeply analyze a cluster, since there is a distinction between correctly (green) or incorrectly (red) assigned and missing time series (blue). In this illustration, we only selected the correctly and incorrectly assigned ones, and the clustering was obtained by a fullspace approach.
III. DEMONSTRATION SCENARIO

The demonstration setup for our evaluation framework enables the conference attendees to try different evaluation scenarios for time series clustering approaches. The visual feedback of the data generator and the clustering component allow the participants to verify the soundness of the clustering results for different algorithms, and it also fosters a better understanding of the concept of temporal subspace clustering in general. Our system (an executable Java .jar) can be downloaded from our website:

http://dme.rwth-aachen.de/TSCDemo

Overall, our goal is to build an experimental framework for temporal subspace clustering that will make it easy for researchers to run experimental benchmarks.

Acknowledgment. This work has been supported by a fellowship within the postdoc-program of the German Academic Exchange Service (DAAD).

REFERENCES