Information Systems ∎ (∎∎∎∎) ∎∎∎–∎∎∎

Contents lists available at ScienceDirect

Information Systems

journal homepage: www.elsevier.com/locate/infosys

Time-series clustering – A decade review

Saeed Aghabozorgi, Ali Seyed Shirkhorshidi*, Teh Ying Wah

Department of Information System, Faculty of Computer Science and Information Technology, University of Malaya (UM), 50603 Kuala Lumpur, Malaysia

ARTICLE INFO

Article history:
Received 13 October 2014
Accepted 27 April 2015

Keywords:
Clustering
Time-series
Distance measure
Evaluation measure
Representations

ABSTRACT

Clustering is a solution for classifying enormous data when there is not any early knowledge about classes. With emerging new concepts like cloud computing and big data and their vast applications in recent years, research works have been increased on unsupervised solutions like clustering algorithms to extract knowledge from this avalanche of data. Clustering time-series data has been used in diverse scientific areas to discover patterns which empower data analysts to extract valuable information from complex and massive datasets. In case of huge datasets, using supervised classification solutions is almost impossible, while clustering can solve this problem using un-supervised approaches. In this research work, the focus is on time-series data, which is one of the popular data types in clustering problems and is broadly used from gene expression data in biology to stock market analysis in finance. This review will expose four main components of time-series clustering and is aimed to represent an updated investigation on the trend of improvements in efficiency, quality and complexity of clustering time-series approaches during the last decade and enlighten new paths for future works.

1. Introduction

Clustering is a data mining technique where similar data are placed into related or homogeneous groups without advanced knowledge of the groups' definitions [1]. In detail, clusters are formed by grouping objects that have maximum similarity with other objects within the group, and minimum similarity with objects in other groups. It is a useful approach for exploratory data analysis as it identifies structure(s) in an unlabelled dataset by objectively organizing data into similar groups. Moreover, clustering is used for exploratory data analysis for summary generation and as a pre-processing step for other data mining tasks or as a part of a complex system.

With increasing power of data storages and processors, real-world applications have found the chance to store and keep data for a long time. Hence, data in many applications is being stored in the form of time-series data, for example sales data, stock prices, exchange rates in finance, weather data, biomedical measurements (e.g., blood pressure and electrocardiogram measurements), biometrics data (image data for facial recognition), particle tracking in physics, etc. Accordingly, different works are found in variety of domains such as Bioinformatics and Biology, Genetics, Multimedia [2–4] and Finance. This amount of time-series data has provided the opportunity of analysing time-series for many researchers in data mining communities in the last decade. Consequently, many researches and projects relevant to analysing time-series have been performed in various areas for different purposes such as: subsequence matching, anomaly detection, motif discovery [5], indexing, clustering,

* Corresponding author. Tel.: +60 196918918.
E-mail addresses: saeed@um.edu.my (S. Aghabozorgi), shirkhorshidi_ali@siswa.um.edu.my, Shirkhorshidi_ali@yahoo.co.uk (A. Seyed Shirkhorshidi), tehyw@um.edu.my (T. Ying Wah).

http://dx.doi.org/10.1016/j.is.2015.04.007
0306-4379/© 2015 Published by Elsevier Ltd.

classification [6], visualization [7], segmentation [8], identifying patterns, trend analysis, summarization [9], and forecasting. Moreover, there are many on-going research projects aimed to improve the existing techniques [10,11].

In the recent decade, there has been a considerable amount of changes and developments in time-series clustering area that are caused by emerging concepts such as big data and cloud computing which increased size of datasets exponentially. For example, one hour of ECG (electrocardiogram) data occupies 1 gigabyte, a typical weblog requires 5 gigabytes per week, the space shuttle database has 200 gigabytes and updating it requires 2 gigabytes per day [12]. Consequently, clustering craved for improvements in recent years to cope with this incremental avalanche of data to keep its reputation as a helpful data-mining tool for extracting useful patterns and knowledge from big datasets. This review is opportune, because despite the considerable changes in the area, there is not a comprehensive review on anatomy and structure of time-series clustering. There are some surveys and reviews that focus on comparative aspects of time-series clustering experiments [6,13–17] but none of them tend to be as comprehensive as we are in this review. This research work is aimed to represent an updated investigation on the trend of improvements in efficiency, quality and complexity of clustering time-series approaches during the last decade and enlighten new paths for future works.

1.1. Time-series clustering

A special type of clustering is time-series clustering. A sequence composed of a series of nominal symbols from a particular alphabet is usually called a temporal sequence, and a sequence of continuous, real-valued elements, is known as a time-series [15]. A time-series is essentially classified as dynamic data because its feature values change as a function of time, which means that the value(s) of each point of a time-series is/are one or more observations that are made chronologically. Time-series data is a type of temporal data which is naturally high dimensional and large in data size [6,17,18]. Time-series data are of interest due to their ubiquity in various areas ranging from science, engineering, business, finance, economics, healthcare, to government [16]. While each time-series is consisting of a large number of data points it can also be seen as a single object [19]. Clustering such complex objects is particularly advantageous because it leads to discovery of interesting patterns in time-series datasets. As these patterns can be either frequent or rare patterns, several research challenges have arisen such as: developing methods to recognize dynamic changes in time-series, anomaly and intrusion detection, process control, and character recognition [20–22]. More applications of time-series data are discussed in Section 1.2. To highlight the importance and the need for clustering time-series datasets, potentially overlapping objectives for clustering of time-series data are given as follows:

1. Time-series databases contain valuable information that can be obtained through pattern discovery. Clustering is a common solution performed to uncover these patterns on time-series datasets.
2. Time-series databases are very large and cannot be handled well by human inspectors. Hence, many users prefer to deal with structured datasets rather than very large datasets. As a result, time-series data are represented as a set of groups of similar time-series by aggregation of data in non-overlapping clusters or by a taxonomy as a hierarchy of abstract concepts.
3. Time-series clustering is the most-used approach as an exploratory technique, and also as a subroutine in more complex data mining algorithms, such as rule discovery, indexing, classification, and anomaly detection [22].
4. Representing time-series cluster structures as visual images (visualization of time-series data) can help users quickly understand the structure of data, clusters, anomalies, and other regularities in datasets.

The problem of clustering of time-series data is formally defined as follows:

Definition 1:. Time-series clustering, given a dataset of n time-series data $D=\{F_1, F_2, .., F_n\}$, the process of unsupervised partitioning of D into $C=\{C_1, C_2, .., C_k\}$, in such a way that homogenous time-series are grouped together based on a certain similarity measure, is called time-series clustering. Then, C_i is called a cluster, where $D=\cup_{i=1}^{k} C_i$ and $C_i \cap C_j = \varnothing$ for $i \neq j$.

Time-series clustering is a challenging issue because first of all, time-series data are often far larger than memory size and consequently they are stored on disks. This leads to an exponential decrease in speed of the clustering process. Second challenge is that time-series data are often high dimensional [23,24] which makes handling these data difficult for many clustering algorithms [25] and also slows down the process of clustering [26]. Finally, the third challenge addresses the similarity measures that are used to make the clusters. To do so, similar time-series should be found which needs time-series similarity matching that is the process of calculating the similarity among the whole time-series using a similarity measure. This process is also known as “whole sequence matching” where whole lengths of time-series are considered during distance calculation. However, the process is complicated, because time-series data are naturally noisy and include outliers and shifts [18], at the other hand the length of time-series varies and the distance among them needs to be calculated. These common issues have made the similarity measure a major challenge for data miners.

1.2. Applications of time-series clustering

Clustering of time-series data is mostly utilized for discovery of interesting patterns in time-series datasets [27,28]. This task itself, fall into two categories: The first group is the one which is used to find patterns that frequently appears in the dataset [29,30]. The second group are methods to discover patterns which happened in datasets surprisingly [31–34]. Briefly, finding the clusters of time-series can be advantageous in different domains to answer following real world problems:

Anomaly, novelty or discord detection: Anomaly detection are methods to discover unusual and unexpected patterns which happen in datasets surprisingly [31–34]. For example,

Please cite this article as: S. Aghabozorgi, et al., Time-series clustering – A decade review, Information Systems (2015), http://dx.doi.org/10.1016/j.is.2015.04.007

Table 1
Samples of objectives of time-series clustering in different domains.

Category	Clustering application	Research works
Aviation/ Astronomy	Astronomical data (star light curves) – pre-processing for outlier detection	[41]
0.5		
Biology	Multiple gene expression profile alignment for microarray time-series data clustering	[42]
	Functional clustering of time series gene expression data	[43]
	Identification of functionally related genes	[44–46]
0.5		
Climate	Discovery of climate indices	[47,48]
	Analysing PM_{10} and $PM_{2.5}$ concentrations at a coastal location of New Zealand	[49]
0.5		
Energy	Discovering energy consumption pattern	[50,51]
0.5		
Environment and urban	Analysis of the regional variability of sea-level extremes	[52]
	Earthquake - Analysing potential violations of a Comprehensive Test Ban Treaty (CTBT) – Pattern discovery and forecasting	[53,54]
	Analysis of the change of population distribution during a day in Salt Lake County, Utah, USA	[55]
	Investigating the relationship between the climatic indices with the clusters/trends detected based on clustering method.	[56]
0.5		
Finance	Finding seasonality patterns (retail pattern)	[57]
	Personal income pattern	[58]
	Creating efficient portfolio (a group of stocks owned by a particular person or company)	[59]
	Discovery patterns from stock time-series	[60]
	Risk reduced portfolios by analyzing the companies and the volatility of their returns	[61]
	Discovery patterns from stock time-series	[29,62]
	Investigate the correlation between hedging horizon and performance in financial time-series.	[63]
Medicine	Detecting brain activity	[64,65]
	Exploring, identifying, and discriminating pathological cases from MS clinical samples	[66]
0.5		
Psychology	Analysis of human behaviour in psychological domain	[67]
Robotics	Forming prototypical representations of the robot's experiences	[68,69]
Speech/voice recognition	Speaker verification	[70]
	Biometric voice classification using hierarchical clustering	[71]
User analysis	Analysing multivariate emotional behaviour of users in social network with the goal to cluster the users from a fully new perspective-emotions	[72]

in sensor databases, clustering of time-series which are produced by sensor readings of a mobile robot in order to discover the events [35].

1- **Recognizing dynamic changes in time-series:** detection of correlation between time-series [36]. For example, in financial databases, it can be used to find the companies with similar stock price move.
2- **Prediction and recommendation:** a hybrid technique combining clustering and function approximation per cluster can help user to predict and recommend [37–40]. For example, in scientific databases, it can address problems such as finding the patterns of solar magnetic wind to predict today's pattern.
3- **Pattern discovery:** to discover the interesting patterns in databases. For example, in marketing database, different daily patterns of sales of a specific product in a store can be discovered.

Table 1 depicts some applications of time-series data in different domains.

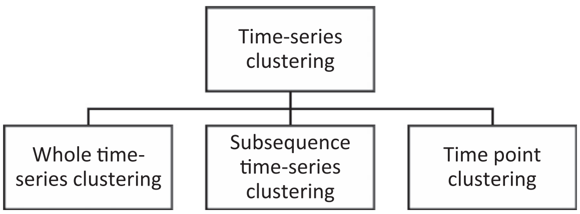

Fig. 1. Time-series clustering taxonomy.

1.3. Taxonomy of time-series clustering

Reviewing the literature, one can conclude that most of clustering time-series related works are classified into three categories: "whole time-series clustering", "subsequence clustering" and "time point clustering" as depicted in Fig. 1. The first two categories are mentioned by Keogh and Lin (2005). Q2

- **Whole time-series clustering** is considered as clustering of a set of individual time-series with respect to their similarity. Here, clustering means applying conventional (usually) clustering on discrete objects, where objects are time-series.

- **Subsequence clustering** means clustering on a set of subsequences of a time-series that are extracted via a sliding window, that is, clustering of segments from a single long time-series.
- **Time point clustering** is another category of clustering which is seen in some papers [74–76]. It is clustering of time points based on a combination of their temporal proximity of time points and the similarity of the corresponding values. This approach is similar to time-series segmentation. However, it is different from segmentation as all points do not need to be assigned to clusters, i.e., some of them are considered as noise.

Essentially, sub-sequence clustering is performed on a single time-series, and Keogh and Lin (2005) represented that this type of clustering is meaningless. Time-point clustering also is applied on a single time-series, and is similar to time-series segmentation as the objective of time-point clustering is finding the clusters of time-point instead of clusters of time-series data. The focus of this study is on the "whole time-series clustering". A complete review on whole time-series clustering is performed and shown in Table 4. Reviewing the literature, it is noticeable that various techniques have been recommended for the clustering of whole time-series data. However, most of them take one of the following approaches to cluster time-series data:

1. Customizing the existing conventional clustering algorithms (which work with static data) such that they become compatible with the nature of time-series data. In this approach, usually their distance measure (in conventional algorithms) is modified to be compatible with the raw time-series data [16].
2. Converting time-series data into simple objects (static data) as input of conventional clustering algorithms [16].
3. Using multi resolutions of time-series as input of a multi-step approach. This approach is discussed further in Section 5.6.

Beside this common characteristic, there are generally three different ways to cluster time-series, namely shape-based, feature-based and model-based.

Fig. 2 shows a brief of these approaches. In the **shape-based** approach, shapes of two time-series are matched as well as possible, by a non-linear stretching and contracting of the time axes. This approach has also been labelled as a raw-data-based approach because it typically works directly with the raw time-series data. Shape-based algorithms usually employ conventional clustering methods, which are compatible with static data while their distance/similarity measure has been modified with an appropriate one for time-series. In the **feature-based** approach, the raw time-series are converted into a feature vector of lower dimension. Later, a conventional clustering algorithm is applied to the extracted feature vectors. Usually in this approach, an equal length feature vector is calculated from each time-series followed by the Euclidean distance measurement [77]. In **model-based** methods, a raw time-series is transformed into model parameters (a parametric model

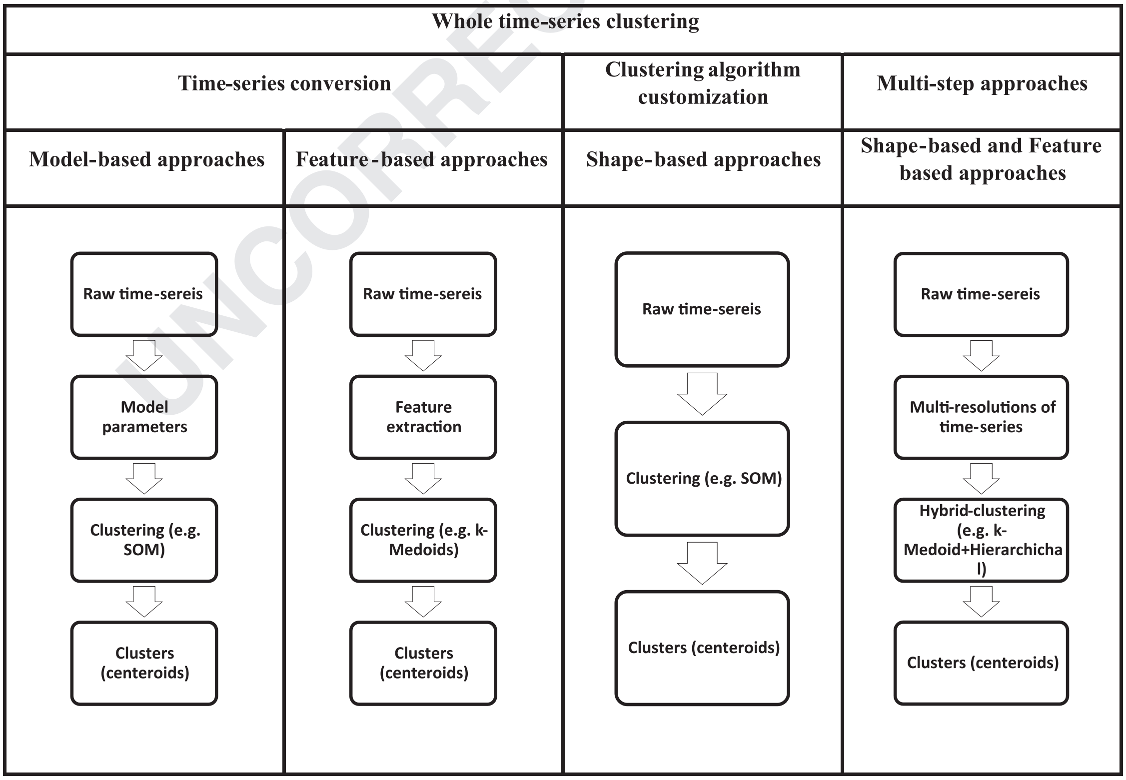

Fig. 2. The time-series clustering approaches.

Please cite this article as: S. Aghabozorgi, et al., Time-series clustering – A decade review, Information Systems (2015), http://dx.doi.org/10.1016/j.is.2015.04.007

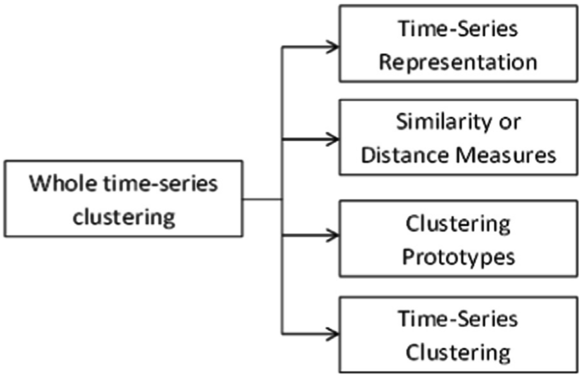

Fig. 3. An overview of four components of whole time-series clustering.

for each time-series,) and then a suitable model distance and a clustering algorithm (usually conventional clustering algorithms) is chosen and applied to the extracted model parameters [16]. However, it is shown that usually model-based approaches has scalability problems [78], and its performance reduces when the clusters are close to each other [79].

Reviewing existing works in the literature, it is implied that essentially time-series clustering has four components: dimensionality reduction or representation method, distance measurement, clustering algorithm, prototype definition, and evaluation. Fig. 3 shows an overview of these components.

The general process in the time-series clustering uses some or all of these components depending on the problem. Usually, data is approximated using a representation method in such a way that can fit in memory. Afterwards, a clustering algorithm is applied on data by using a distance measure. In the clustering process, usually a prototype is required for summarization of the time-series. At last, the clusters are evaluated using criteria. In the following subsections, each component is discussed, and several related works and methods are reviewed.

1.4. Organization of the review

In the rest of this paper, we will provide a state-of-the-art review on main components available in time-series clustering plus the evaluation methods and measures available for validating time-series clustering. In Section 2, time-series representation is discussed. Similarity and dissimilarity measures are represented in Section 3. Sections 4 and 5 are dedicated to clustering prototypes and clustering algorithms respectively. In section 6 evaluation measures is discussed and finally the paper is concluded in Section 7.

2. Representation methods for time series clustering

The first component of time-series clustering explained here is dimension reduction which is a common solution for most whole time-series clustering approaches proposed in the literature [9,80–82]. This section reviews methods of time-series dimension reduction which is known as time-series representation as well. Dimensionality reduction represents the raw time-series in another space by transforming time-series to a lower dimensional space or by feature extraction. The reason that dimensionality reduction is greatly important in clustering of time-series is firstly because it reduces memory requirements as all raw time-series cannot fit in the main memory [9,24]. Secondly, distance calculation among raw data is computationally expensive, and dimensionality reduction significantly speeds up clustering [9,24]. Finally, when measuring the distance between two raw time-series, highly unintuitive results may be garnered, because some distance measures are highly sensitive to some "distortions" in the data [3,83], and consequently, by using raw time-series, one may cluster time-series which are similar in noise instead of clustering them based on similarity in shape. The potential to obtain a different type of cluster is the reason why choosing the appropriate approach for dimension reduction (feature extraction) and its ratio is a challenging task [26]. In fact, it is a trade-off between speed and quality and all efforts must be made to obtain a proper balance point between quality and execution time.

Definition 2:. Time-series representation, given a time-series data $F_i = \{f_1, ..., f_t, ..., f_T\}$, representation is transforming the time-series to another dimensionality reduced vector $F'_i = \{f'_1, ..., f'_x\}$ where $x < T$ and if two series are similar in the original space, then their representations should be similar in the transformation space too.

According to [83], choosing an appropriate data representation method can be considered as the key component which effects the efficiency and accuracy of the solution. High dimensionality and noise are characteristics of most time-series data [6], consequently, dimensionality reduction methods are usually used in whole time-series clustering in order to address these issues and promote the performance. Time-series dimensionality reduction techniques have progressed a long way and are widely used with large scale time-series dataset and each has its own features and drawbacks. Accordingly, many researches had been carried out focusing on representation and dimensionality reduction [84–90]. It is worth here to mention about the one of the recent comparisons on representation methods. H. Ding et al. [91] have performed a comprehensive comparison of 8 representation methods on 38 datasets. Although, they had investigated the indexing effectiveness of representation methods, the results are advantageous for clustering purpose as well. They use tightness of lower bounds to compare representation methods. They show that there is very little difference between recent representation methods. In taxonomy of representations, there are generally four representation types [9,83,92,93]: data adaptive, non-data adaptive, model-based and data dictated representation approaches as are depicted in Fig. 4.

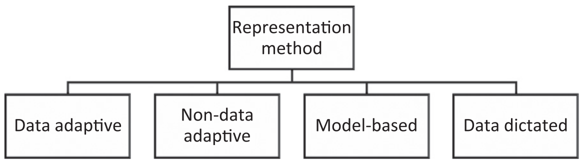

Fig. 4. Hierarchy of different time-series representation approaches.

Please cite this article as: S. Aghabozorgi, et al., Time-series clustering – A decade review, Information Systems (2015), http://dx.doi.org/10.1016/j.is.2015.04.007

Table 2
Representation methods for time-series data.

Representation method	Complexity	Type	Comments	Introduced by
Discrete Fourier Transform (DFT)	$O(n(\log(n))$	Non data adaptive, Spectral	**Usage:** Natural Signals **Pros:** No false dismissals. **Cons:** Not support time warped queries	[20,108]
Discrete Wavelet Transform (DWT)	O(n)	Non data adaptive, Wavelet	**Usage:** stationary signals **Pros:** Better results than DFT **Cons:** Not stable results, Signals must have a length $n=2^{some\_integer}$	[85,108,109]
Singular Value Decomposition (SVD)	very expensive $O(Mn^2)$	Data adaptive	**Usage:** text processing community **Pros:** underlying structure of the data	[20,97]
Discrete Cosine Transformation (DCT)	[a]	Non data adaptive, Spectral	-	[97]
Piecewise Linear Approximation (PLA)	$O(n \log n)$ complexity for “bottom up” algorithm	Data adaptive	**Usage:** natural signals, biomedical **Cons:** Not (currently) indexable, very expensive $O(n^2N)$	[86]
Piecewise Aggregate Approximation (PAA)	Extremely Fast $O(n)$	Non data adaptive	-	[24,90]
Adaptive Piecewise Constant Approximation (APCA)	$O(n)$	Data adaptive	**Pros:** Very efficient **Cons:** complex implementation	[87]
Perceptually important point (PIP)	[a]	Non data adaptive	**Usage:** Finance	[110]
Chebyshev Polynomials (CHEB)	[a]	Non data adaptive, Wavelet, Orthonormal	–	[99]
Symbolic Approximation (SAX)	$O(n)$	Data adaptive	**Usage:** string processing and bioinformatics **Pros:** Allows Lower bounding and Numerosity Reduction **Cons:** Discretize and alphabet size	[111]
Clipped Data	[a]	Data dictated	**Usage:** Hardware **Cons:** Ultra compact representation	[83]
Indexable Piecewise Linear Approximation (IPLA)	[a]	Non data adaptive	-	[101]

[a] Not indicated by authors.

2.1. Data adaptive

Data adaptive representation methods are performed on all time-series in datasets and try to minimize the global reconstruction error [94] using arbitrary length (non-equal) segments. This technique has been applied in different approaches such as Piecewise Polynomials Interpolation (PPI) [95], Piecewise Polynomials Regression (PPR) [96],

Table 3
Similarity measure approaches in the literature.

Distance measure	Characteristics	Method	Defined by
Dynamic Time Warping (DTW)	Elastic Measure (one-to-many/one-to-none) Very well in deal with temporal drift. Better accuracy than Euclidean distance [129,114,120,90]. Lowe efficiency than Euclidean distance and triangle similarity.	Shape-based	[118,119]
Pearson's correlation coefficient and related distances	Invariant to scale and location of the data points	Compression based dissimilarity	[124]
Euclidean distance (ED)	Lock-step Measure (one-to-one) using in indexing, clustering and classification, Sensitive to scaling.	Shape-based	[20]
KL distance	–	Compression based dissimilarity	[130]
Piecewise probabilistic	–	Compression based dissimilarity	[131]
Hidden Markov models (HMM)	Able to capture not only the dependencies between variables, but also the serial correlation in the measurements	Model based	[116]
Cross-correlation based distances	Noise reduction, able to summarize the temporal structure	Shape-based	[132]
Cosine wavelets	–	Compression based dissimilarity	[126]
Autocorrelation	–	Compression based dissimilarity	[133]
Piecewise normalization	It involves time intervals, or "windows," of varying size. But it is not clear how to determine these "windows."	Compression based dissimilarity	[125]
LCSS	Noise robustness	Shape-based	[120,121]
Cepstrum	A spectral measure which is the inverse Fourier transform of the short-time logarithmic amplitude spectrum	Compression based dissimilarity	[107]
Probability-based distance	Able to cluster seasonality patterns	Compression based dissimilarity	[57]
ARMA	–	Model based	[107,117]
Short time-series distance (STS)	Sensitive to scaling. Can capture temporal information, regardless of the absolute values	Feature-based	[44]
J divergence	–	Shape-based	[53]
Edit Distance with Real Penalty (ERP)	Robust to noise, shifts and scaling of data, a constant reference point is used	Shape-based	[134]
Minimal Variance Matching (MVM)	Automatically skips outliers	Shape-based	[122]
Edit Distance on Real sequence (EDR)	Elastic measure (one-to-many/one-to-none), uses a threshold pattern	Shape-based	[135]
Histogram-based	Using multi-scale time-series histograms	Shape-based	[136]
Threshold Queries (TQuEST)	Threshold-based Measure, considers intervals, during which the time-series exceeds a certain threshold for comparing time-series rather than using the exact time-series values.	Model based	[137]
DISSIM	Proper for different sampling rates	Shape-based	[138]
Sequence Weighted Alignment model (Swale)	Similarity score based on both match rewards and mismatch penalties.	Shape-based	[139]
Spatial Assembling Distance (SpADe)	Pattern-based Measure	Model based	[140]
Compression-based dissimilarity measure (CDM)	In [123] Keogh et al. a parameter-light distance measure method based on Kolmogorov complexity theory is suggested. Compression-based dissimilarity measure (CDM) is adopted in this paper.	Compression based dissimilarity	[123]
Triangle similarity measure	Can deal with noise, amplitude scaling very well and deal with offset translation, linear drift well in some situations [141].	Shape-based	[141]
Dictionary-based compression	Lang et al. [142] develop a dictionary compression score for similarity measure. A dictionary-based compression technique is suggested to compute long time-series similarity	Compression based dissimilarity	[142]

Piecewise Linear Approximation (PLA), Piecewise Constant Approximation (PCA), Adaptive Piecewise Constant Approximation (APCA) [87], Singular Value Decomposition (SVD) [20,97], Natural Language, Symbolic Natural Language (NLG) [98], Symbolic Aggregate ApproXimation (SAX) and iSAX [84]. Data adaptive representations can better approximate each series, but the comparison of several time-series is more difficult.

Please cite this article as: S. Aghabozorgi, et al., Time-series clustering – A decade review, Information Systems (2015), http://dx.doi.org/10.1016/j.is.2015.04.007

2.2. Non-data adaptive

Non-data adaptive approaches are representations which are suitable for time-series with fix size (equal-length) segmentation, and the comparison of representations of several time-series is straightforward. The methods in this group are Wavelets [85]: HAAR, DAUBECHIES, Coeiflets, Symlets, Discrete Wavelet Transform(DWT), spectral Chebyshev Polynomials [99], spectral DFT [20], Random Mappings [100], Piecewise Aggregate Approximation (PAA) [24] and Indexable Piecewise Linear Approximation (IPLA) [101].

2.3. Model based

Model based approaches represent a time-series in a stochastic way such as Markov Models and Hidden Markov Model (HMM) [102–104], Statistical Models, Time-series Bitmaps [105], and Auto-Regressive Moving Average (ARMA) [106,107]. In the data adaptive, non-data adaptive, and model based approaches user can define the compression-ratio based on the application in hand.

2.4. Data dictated

In contrast, in data dictated approaches, the compression-ratio is defined automatically based on raw time-series such as Clipped [83,92]. In the following table (Table 2) the most famous representation methods in the literature are shown.

2.5. Discussion on time series representation methods

Different approaches for representation of time-series data are proposed in previous studies. Most of these approaches are focused to speed up the process and reduce the execution time and mostly they emphasis on indexing process for achieving to this goal. At the other hand some other approaches consider the quality of representation, as an instance in [83], the authors focus on the accuracy of representation method and suggest a bit level approximation of time-series. Each time-series is represented by a bit string, and each bit value specifies whether the data point's value is above the mean value of the time-series. This representation can be used to compute an approximate clustering of the time-series. This kind of representation which also referred to as clipped representation has capability of being compared with raw time-series, but in the other representations, all time-series in dataset must be transformed into the same representation in terms of dimensionality reduction. However, clipped series are theoretically and experimentally sufficient for clustering based on similarity in change (model based dissimilarity measurement) not clustering based on shape. Reviewing the literature shows that limited works are available for discrete valued time-series and also it is noticeable that most of research works are based on evenly sampled data while limited works addressed unevenly sampled data. Additionally data error is not taken into account in most of research works. Finally among all of the papers reviewed in this article, none of them addressed handling multivariate time series data with different length for each variable.

3. Similarity/dissimilarity measures in time-series clustering

This section is a review on distance measurement approaches for time-series. The theoretical issue of time-series similarity/dissimilarity search is proposed by Agrawal et al. [108] and subsequently it became a basic theoretical issue in data mining community. Time-series clustering relies on distance measure to a high extent. There are different measures which can be applied to measure the distance among time-series. Some of similarity measures are proposed based on a specific time-series representations, for example, MINDIST which is compatible with SAX [84], and some of them work regardless of representation methods, or are compatible with raw time-series. In traditional clustering, distance between static objects is exactly match based, but in time-series clustering, distance is calculated approximately. In particular, in order to compare time-series with irregular sampling intervals and length, it is of great significance to adequately determine the similarity of time-series. There is different distance measures designed for specifying similarity between time-series. The Hausdorff distance, modified Hausdorff (MODH), HMM-based distance, Dynamic Time Warping (DTW), Euclidean distance, Euclidean distance in a PCA subspace, and Longest Common Sub-Sequence (LCSS) are the most popular distance measurement methods that are used for time-series data. References on distance measurement methods are given in Table 3. One of the simplest ways for calculating distance between two time-series is considering them as univariate time-series, and then calculating the distance measurement across all time points.

Definition 3:. Univariate time-series, a univariate time-series is the simplest form of temporal data and is a sequence of real numbers collected regularly in time, where each number represents a value [25].

Definition 4:. Time-series distance, let $F_i = \{f_{i1}, .., f_{it}, .., f_{iT}\}$ be a time-series of length T. If the distance between two time-series is defined across all time points, then $dist(F_i, F_j)$ is the sum of the distance between individual points

$$dist(F_i, F_j) = \sum_{t=1}^{T} dist(f_{it}, f_{jt}) \tag{3.1}$$

Researches done on shape-based distance measuring of time-series usually have to challenge with problems such as noise, amplitude scaling, offset translation, longitudinal scaling, linear drift, discontinuities and temporal drift which are the common properties of time-series data, these problems are broadly investigated in the literature [86]. The choice of a proper distance approach depends on the characteristic of time-series, length of time-series, representation method, and of course on the objective of clustering time-series to a high extent. This is depicted in Fig. 5.

Typically, there are three objectives which respectively require different approaches [112].

3.1. Finding similar time-series in time

Because this similarity is on each time step, correlation based distances or Euclidean distance measure are proper for this objective. However, because this kind of distance

Please cite this article as: S. Aghabozorgi, et al., Time-series clustering – A decade review, Information Systems (2015), http://dx.doi.org/10.1016/j.is.2015.04.007

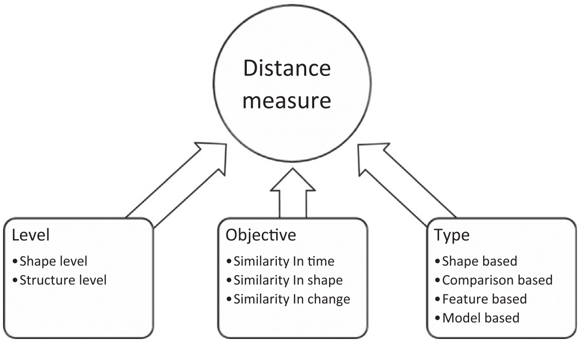

Fig. 5. Distance measure approaches in the literature.

measuring is costly on raw time-series, the calculation is performed on transformed time-series, such as Fourier transforms, wavelets or Piecewise Aggregate Approximation (PAA). Keogh and Kasetty [6], have done an comprehensive review on this matter. Clustering of time-series that are correlated, (e.g., to cluster time-series of share price related to many companies to find which shares change together and how they are correlated) is categorized as clustering based on similarity in time [83,112].

3.2. Finding similar time-series in shape

The time of occurrence of patterns is not important to find similar time-series in shape. As a result, elastic methods [108,113] such as Dynamic time Warping (DTW) [114] is used for dissimilarity calculation. Using this definition, clusters of time-series with similar patterns of change are constructed regardless of time points, for example, to cluster share price related to different companies which have a common pattern in their stock independent on its occurrence in time-series [112]. Similarity in time is an especial case of similarity in shape. A research has revealed that similarity in shape is superior to metrics based on similarity in time [115].

3.3. Finding similar time-series in change (structural similarity)

In this approach, usually modelling methods such as Hidden Markov Models (HMM) [116] or an ARMA process [107,117] are utilized, and then similarity is measured on the parameters of fitted model to time-series. That is, clustering time-series with similar autocorrelation structure, e.g., clustering of shares which have a tendency to increase after a fall in share price in the next day [112]. This approach is proper for long time-series, not for modest or short time-series [21].

Clustering approaches could be classified into two categories based on the length of time-series: "shape level" and "structure level". The "**shape level**" is usually utilized to measure similarity in short-length time-series clustering such as expression profiles or individual heartbeats by comparing their local patterns, whereas "**structure level**" measures similarity which is based on global and high level structure, and it is used for long-length time-series data such as an hour's worth of ECGs or yearly meteorological data [21]. Focusing on shape-based clustering of short length time-series, in this study, shape level similarity is used. Depending on the objective and length of time-series, the proper type of distance measures is determined. Essentially, there are four types of distance measure in the literature. Please refer to Table 3 for references on the types of distance measure. *Shape-based similarity* measure is to find the similar time-series in time and shape, such as Euclidean, DTW [118,119], LCSS [120,121], MVM [122]. It is a group of methods which are proper for short time-series. *Compression based similarity* is suitable for short and long time-series, such as CDM [123], Autocorrelation, Short time-series distance [44], Pearson's correlation coefficient and related distances [124], Cepstrum [107], Piecewise normalization [125] and Cosine wavelets [126]. *Feature based similarity* measure are proper for long time-series, such as Statistics, Coefficients, *Model based similarity* is proper for long time-series, such as HMM [116] and ARMA [107,117].

A survey on various methods for efficient retrieval of similar time-series were given by Last and Kandel [127]. Furthermore, in [16], authors have presented the formulas of various measures. Then, Zhang et al. [128] have performed a complete survey on the aforementioned distance measurements and compared them in different applications. In Table 3, different measures are compared in terms of complexity and their characteristics.

3.4. Discussion on distance measures

Choosing an adequately accurate distance measure is controversial in time-series clustering domain. There are many distance measure proposed by researchers which were compared and discussed in Section 3. However, the following conclusion can be drawn from literature.

1) Investigating the mentioned approaches as similarity/dissimilarity measure, it is implied that the most effective and accurate approaches are the ones which are based on dynamic programming (DP) which are very expensive in time execution (the cost of comparing two time-series is quadratic in the length of the time-series) [143]. Although, usually some constraints are taken for these distance/similarity measurements to mitigate the complexity [119,144], it needs careful tuning of parameters to be efficient and effective. As a result, again, a trade-off between speed and accuracy should be found in usage of this metrics. In another view, it is worthwhile to understand the extent that distance measure is effective in large scale datasets of time-series. This matter is not obtained from literature because most of the considered works are based on rather small datasets.
2) In the similarity measure researches, varieties of challenges are considered pertaining to distance measurement. A big challenge is the issue of incompatibility of distance metric with the representation method. For example, one of the common approaches that is applied to time-series analysis is based upon frequency-domain [85,109], while using this space, it is difficult to find the similarity among sequences and produce value-based differences to be used in clustering.

Please cite this article as: S. Aghabozorgi, et al., Time-series clustering – A decade review, Information Systems (2015), http://dx.doi.org/10.1016/j.is.2015.04.007

3) Euclidean distance and DTW are the most common methods for similarity measure in time-series clustering. A research has shown that, in terms of time-series classification accuracy, the Euclidean distance is surprisingly competitive [145], however, DTW also has its strength in similarity measurements which cannot be declined

4. Time-series cluster prototypes

Finding the cluster prototype or cluster representative is an essential subroutine in time-series clustering approaches [3,86,112,114,146,147]. One of the approaches to address the low quality problem in time-series clustering is remedying the issue of inaccurate prototypes of clusters, especially in partitioning clustering algorithms such as k-Means, k-Medoids, Fuzzy C-Means (FCM), or even Ascendant Hierarchical Clustering which requires a prototype. In these algorithms, the quality of clusters is highly dependent on quality of prototypes. Given time-series in a cluster, it is clear that the cluster's prototype R_j minimizes the distance between all time-series in the cluster and its prototype. Time-series R_j that minimizes $E(C_i, R_j)$ is called a Steiner sequence [148].

$$E(C_i, R_j) = \frac{1}{n}\sum_{x=1}^{n} dist(F_x, R_j),\ C_i = \{F_1, F_2, .., F_n\} \tag{4.1}$$

There are a few methods for calculating prototypes published in the literature of time-series, however most of these publications have not proved the correctness of their methods [149]. But, generally three approaches can be seen for defining the prototypes:

1. The medoid sequence of the set.
2. The average sequence of the set.
3. The local search prototype.

In following these three approaches are explained and discussed.

4.1. Using medoid as prototype

In time-series clustering, the most common way to approach optimal Steiner sequence is to use cluster medoid as the prototype [150]. In this approach, the centre of a cluster is defined as a sequence which minimizes the sum of squared distances to other objects within the cluster. Given time-series in a cluster, the distance of all time-series pairs within the cluster is calculated using a distance measure such as Euclidean or DTW. Then, one of the time-series in the cluster, which has lower sum of square error is defined as medoid of the cluster [151]. Moreover, if the distance is a non-elastic approach such as Euclidean, or if the centroid of the cluster can be calculated, it can be said that medoid is the nearest time-series to centroid. Cluster medoid is very common among works related to time-series clustering and has been used in many papers such as: [77,150,152,153].

4.2. Using averaging prototype

If the time-series are from equal length, and distance metric is a none-elastic distance metric (e.g., Euclidean distance) in clustering process, then the averaging method is a simple averaging technique which is equal to mean of the time-series at each point. However, in the case that there are time-series with different length [149] or in the case which the similarity between time-series is based on "similarity in shape", its one-to-one mapping nature, makes it unable to capture the actual average shape. For example, in the cases that Dynamic Time Warping (DTW) or Longest Common Sub-Sequence (LCSS) are very appropriate [154], averaging prototype is evaded, because it is not a trivial task. For more evidence, one can see many works in the literature [86,112,114,146,155,156], which avoid using elastic approaches (e.g., DTW and LCSS) where there is a need to use a prototype without providing adequate reasons (whether the clustering is based on similarity in time or shape). Two averaging methods using DTW and LCSS are briefly explained following in this section.

Shape averaging using Dynamic Time Warping (DTW): in this approach, one method to define the prototype of a cluster is by combination of pairs of time-series hierarchically or sequentially. For example, shape averaging using Dynamic Time Warping, until only one time-series is left [154]. The drawback of this method is its dependency on the ordering of choosing pairs which results in different final prototypes [2]. Another method is the approach mentioned by Abdulla and Chow [157], where authors proposed a cross-words reference template (CWRT), where at first, the medoid is find as the initial guess, then all sequences are aligned by DTW to the medoid, and then the average time-series is computed. The resulting time-series has the same length as the medoid, but the method is invariant to the order of processing sequences [77]. In another study, the authors present a global averaging method for defining the prototypes [158]. They use an averaging approach where the distance method for clustering or classification is DTW. However, its accuracy is dependent on the length of the initial average sequence and value of its coordinates.

Shape averaging using Longest Common Sub-Sequence (LCSS): the longest common subsequence [159] generally permits to make a summary of a set of sequences. This approach supports the elastic distances and unequal size time-series. Aghabozorgi et al. [160] and Aghabozorgi, Wah, Amini, and Saybani [161] propose a fuzzy clustering approach for time-series clustering, and utilize the averaging method by LCSS as prototype.

4.3. Using local search prototype

In this approach, at first the medoid of cluster is computed, then using averaging method (Section 4.2), averaged prototype is calculated based on warping paths. Afterward, new warping paths are calculated to the averaged prototype. Hautamaki et al. [77] propose a prototype obtained by local search, instead of medoid to overcome the poor quality in time-series clustering in Euclidean space. They apply medoid, average and local search on k-Medoids, Random Swap (RS) and Agglomerative Hierarchical clustering (where k-means is used to fine-tune the output) to evaluate their work. They figured out that local search provides the best clustering accuracy and also more improvement to k-Medoids. However, it is not clear how much improvement it has in comparison with other works such as medoid averaging methods which are another frequently used prototype.

4.4. Discussion

One of the problems which lead to low accuracy of clusters is poor definition or updating method of prototypes in time-series clustering process, especially in partitioning approaches. Many clustering algorithms suffer from low accuracy of representation methods [77,149]. Moreover, the inaccurate prototype can affect convergence of clustering algorithms which results in low quality of obtained clusters [149]. Different approaches of defining prototypes were discussed in Section 4. In this study, the averaging approach is used in order to find the prototypes of the sub-clusters because the used distance metric is a none-elastic distance metric (ED). Although for the merging purpose, an arbitrary method can be used if it is compatible with elastic methods such as [158], however for different schemes the simple “medoid” is used as prototype to be compatible with the elasticity of distance metric DTW, with k-Medoids algorithm, and also to provide fair condition for evaluation of the proposed model with existing approaches.

5. Time-series clustering algorithms

In this section, the existing works related to clustering of time-series data are concentrated and discussed. Some of them are using raw time-series and some try to use reduction methods before clustering of time-series data. As it is demonstrated in Fig. 6, generally clustering can be broadly classified into six groups: Partitioning, Hierarchical, Grid-based, Model-based, Density-based clustering and Multi-step clustering algorithms. In the following, the application of each group in time-series clustering is discussed in detail.

5.1. Hierarchical clustering of time-series

Hierarchal clustering [150] is an approach of cluster analysis which makes a hierarchy of clusters using agglomerative or divisive algorithms. Agglomerative algorithm considers each item as a cluster, and then gradually merges the clusters (bottom-up). In contrast, divisive algorithm starts with all objects as a single cluster and then splits the cluster to reach the clusters with one object (top-down). In general, hierarchical algorithms are weak in terms of quality because they cannot adjust the clusters after splitting a cluster in divisive method, or after merging in agglomerative method. As a result, usually hierarchical clustering algorithms are combined with another algorithm as a hybrid clustering approach to remedy this issue. Moreover, some extended works are done to improve the performance of hierarchical clustering such as Chameleon [162], CURE [163] and BIRCH [164] where the merge approach is enhanced or constructed clusters are refined.

Similarly in hierarchical clustering of time-series, nested hierarchy of similar groups is generated based on a pair-wise distance matrix of time-series [165]. Hierarchical clustering has a great visualization power in time-series clustering [86,166] which makes it an approach to be used for time-series clustering to a great extent. For example, Oates, Schmill, and Cohen [167] use agglomerative clustering to produce the clusters of the experiences of an autonomous agent. They use Dynamic Time Warping (DTW) as a dissimilarity measure with a dataset containing 150 trials of real Pioneer data in a variety of experiences. In another study by Hirano and Tsumoto [168], the authors use average linkage agglomerative clustering which is a type of hierarchical approach for time-series clustering. Moreover, in many researches, hierarchical is used to evaluate dimensionality reduction or distance metric due to its power in visualization. For example, in a study [9], the authors presented Symbolic Aggregate Approximation (SAX) representation and they used hierarchical clustering to evaluate their work. They show that using SAX, hierarchical clustering has a result similar with Euclidean distance.

Additionally, in contrast to most algorithms, hierarchy clustering does not require the number of clusters as an initial parameter which is a well-known and outstanding feature of this algorithm. It is also a strength point in time-series clustering, because usually it is hard to define the number of clusters in real world problems. Moreover, despite many algorithms, hierarchical clustering has the ability to cluster time-series with unequal length. It is possible to cluster unequal time-series using this algorithm if an appropriate elastic distance measure such as Dynamic Time Warping (DTW) [118,119] or Longest Common Subsequence (LCSS) [120,121] is used to compute the dissimilarity/similarity of time-series. In fact the reality that prototypes are not necessary in its process has made this algorithm capable to accept unequal time-series. However, hierarchical clustering is essentially not capable to deal effectively with large time-series [21] due to its quadratic computational complexity and accordingly, it leads to be restricted to small datasets because of its poor scalability.

5.2. Partitioning clustering

A partitioning clustering method makes k groups from n unlabelled objects in the way that each group contains at least one object. One of the most used algorithms of partitioning clustering is **k-Means** [169] where each cluster has a prototype which is the mean value of its objects. The main idea behind k-Means clustering is the minimization of the total distance (typically Euclidian distance) between all objects in a cluster from their cluster center (prototype).

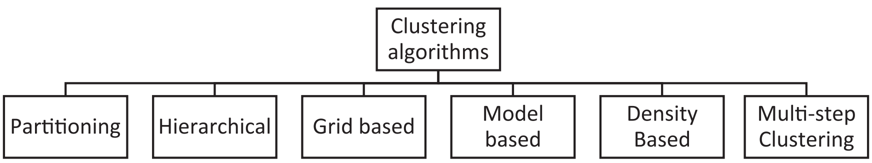

Fig. 6. Clustering approaches.

Please cite this article as: S. Aghabozorgi, et al., Time-series clustering – A decade review, Information Systems (2015), http://dx.doi.org/10.1016/j.is.2015.04.007

Table 4
Whole time-series clustering algorithms.

Article	Representation method	Distance measurement	Clustering algorithm	Comments (P:Positive, N:Negative)
Košmelj and Batagelj [50]	Raw time-series	Euclidean	Modified relocation clustering	P: Multiple variable support
Golay et al. [132]	Raw time-series	Euclidean and two cross correlation-based	FCM	P: Noise Robustness
Kakizawa, Shumway, and Taniguchi [192]	Raw time-series	J divergence	Agglomerative hierarchical	P: Multiple variable support
Van Wijk and Van Selow [166]	Raw time-series	Root mean square	Agglomerative hierarchical	N: Single variable, using raw time-series
Policker and Geva [193]	Raw time-series	Euclidean	Fuzzy clustering	N: Single, using raw time-series
Qian, Dolled-Filhart, Lin, Yu, and Gerstein [194]	Raw time-series	Ad hoc distance	Single-linkage	N: using raw time-series Sensitive to noise
Kumar and Patel [57]	Raw time-series	Gaussian models of data errors	Agglomerative hierarchical	–
Liao et al. [152]	Raw time-series	DTW and Kullback–Liebler distance	*k*-Medoids-based genetic clustering	P: Support unequal time-series N: Single variable support Sensitive to noise
Wismüller et al. [64]	Raw time-series	***	Neural network clustering	N: Single variable support, using raw time-series
Möller-Levet, Klawonn, Cho, and Wolkenhauer [44]	piecewise linear function	STS	Modified FCM	–
Vlachos, Lin, and Keogh [165]	DWT (Discrete Wavelet Transform) Haar wavelet	Euclidean	k-means,	P: Incremental N: Sensitive to noise
Shumway [53]	Raw time-series	Kullback–Leibler discrimination information Measures	Agglomerative hierarchical	P: Multiple variable support
Lin, Vlachos, Keogh, and Gunopulos [18]	Wavelets.	Euclidean Distance	partitioning clustering, k-Means and EM	P: Incremental N: Sensitive to noise
Z.J. Wang and Willett [195]	Raw time-series	GLR (generalized likelihood ratio)	two stages approach	N: Subsequence Segmentation. Sensitive to noise
[111]	SAX	compression-based distance	Hierarchy	N: Sensitive to noise
X. Wang, Smith, and Hyndman [196]	global characteristics	Euclidean	SOM	N: Only focus on dimensionality reduction method Sensitive to noise
Ratanamahatana, Keogh, Bagnall, and Lonardi [83]	BLA (clipped time-series representation)	LB_clipped	k-means	N: Sensitive to noise
Focardi and others [197]	Raw time-series	3 types of distances	–	N: Using Raw time-series Sensitive to noise
Abonyi, Feil, Nemeth, and Arva [198]	PCA	SpCA Factor	Hierarchical	P: Anomaly detection N: Sensitive to noise
Tseng and Kao [199]	gene expression	Euclidean distance, Pearson's correlation	Modified CAST	P: Focus on clustering N: Sensitive to noise
Bagnall and Janacek [112]	Clipped	Euclidean	k-Means, k-Medoids	–
Liao [200]	SAX	Euclidean and symmetric version of Kullback–Liebler	*k*-Means and fuzzy *c*-Means	P: Multiple variable support Support unequal time-series
Ratanamahatana and Niennattrakul [4]	Raw time-series	Dynamic Time Warping	k-Means, k-Medoids	P: Noise Robustness N: using raw time-series
Bao [201] Bao and Yang [202]	a critical point model (CPM)	–	turning points	P: Using important points
Lin, Keogh, Wei, and Lonardi [84]	ESAX	Min-Distance	Partitioning Hierarchal	N: Only focus on distance measurement Sensitive to noise
Hautamaki et al. [77]	Raw time-series	DTW	K-mean, Hierarchical, RS	P: Only was compared with medoid Support unequal time-series
Guo, Jia, and Zhang [60]	feature-based using ICA	–	modified k-means	N: Sensitive to noise
Liu and Shao [203]	SAX	trend statistics distance	Hierarchical	P: Using symbolized TS
Fu, Chung, Luk, and Ng [204]	PIP (perceptually important points)	Vertical distance	k-Means	P: incremental Support unequal time-series N: Only indexing Sensitive to noise
Lai, Chung, and Tseng [205]	SAX, Raw time-series	Min-Dist, Eucleadian distance	Two-level clustering: CAST,CAST	P: Support unequal time-series N: Based on subsequence,CAST is poor in front of huge data Sensitive to noise

Please cite this article as: S. Aghabozorgi, et al., Time-series clustering – A decade review, Information Systems (2015), http://dx.doi.org/10.1016/j.is.2015.04.007

Table 4 (*continued*)

Article	Representation method	Distance measurement	Clustering algorithm	Comments (P:Positive, N:Negative)
Gullo, Ponti, Tagarelli, Tradigo, and Veltri [66]	DSA	DTW	k-Means	–
Zhang [206]	Raw time-series	triangle distance	Hierarchical	–
Aghabozorgi [161]	Discrete Wavelet Transform (DWT)	Longest Common Sub-Sequence (LCSS)	Fuzzy c-Means Clustering (FCM)	P: Flexibility and accuracy
Zakaria [207]	Shapelets	length-normalized Euclidean distance	k-Means	P: Cluster time-series of different lengths
Darkins [208]	Gaussian process data model	Dirichlet Process Model (DPM)	Bayesian Hierarchical Clustering (BHC)	–
Ji [48]	Raw time-series	Euclidean Distance (ED)	Fuzzy c-Means Clustering (FCM)	P: Dynamic nature of algorithm
Seref [209]	Raw time-series	Arbitrary pairwise distance matrices	DKM-S (Modified Discrete k-Median Clustering)	–
Ghassempour [210]	Hidden Markov Models (HMMs)	KL-Distance	PAM (Partitioning Around Medoids)	P: Support categorical and continues values
Aghabozorgi [211]	Piecewise Aggregate Approximation (PAA)	Euclidean distance and Dynamic Time Warping	Hybrid, k-Medoids+Hierarchical	P: Better accuracy over traditional clustering algorithms

Prototype in k-Means process is defined as mean vector of objects in a cluster. However, when it comes to time-series clustering, it is a challenging issue and is not trivial [149]. Another member of partitioning family is **k-Medoids (PAM)** algorithm [150], where the prototype of each cluster is one of the nearest objects to the centre of the cluster. Moreover, CLARA and CLARANS [170] are improved version of k-Medoid algorithm for mining in spatial databases. In both k-Means and k-Medoids clustering algorithms, number of clusters, k, has to be pre-assigned, which is not available or feasible to be determined for many applications, so it is impractical in obtaining natural clustering results and is known as one of their drawbacks in static objects [21] and also time-series data [15]. It is even worse in time-series because the datasets are very large and diagnostic checks for determining the number of clusters is not easy. Accordingly, authors in [171] investigate the role of choosing correct initial clusters in quality and time-execution of k-Means in time-series clustering. However, k-Means and k-Medoids are very fast compared to hierarchical clustering [169,172] and it has made them very suitable for time-series clustering and has been used in many works [18,60,77,112,173].

k-Means and k-Medoids algorithms make clusters which are constructed in ‘hard’ or ‘crispy’ manner and it means that an object is either a member of a cluster or not. On the other hand, **FCM** (**Fuzzy c-Means**) algorithm [174,175] and Fuzzy c-Medoids algorithm [176] build ‘soft’ clusters. In fuzzy clustering, an object has a degree of membership in each cluster [177]. Fuzzy partitioning algorithms have been used for time-series clustering in some areas. For example, in [70], authors use FCM (Fuzzy c-Means) to cluster time-series for speaker verification. In another work [178], the authors use fuzzy variant to cluster similar object motions that were observed in a video collection. They adopt an EM-based algorithm and a mixture of HMMs to cluster time-series data. Then, each time-series is assigned to each cluster to a certain degree. Moreover, using FCM, authors in [132] cluster MRI time-series of brain activities. They use raw univariate time-series of equal length. As distance metric, they use Euclidian distance and cross-correlation. They evaluate their work with different numbers of clusters (k) and recommend using a large number of clusters as initial clusters. However, it is not defined how they achieve the optimal number of clusters in this work.

Generally, partitioning approaches, whether crispy or fuzzy, need defining prototypes and their accuracy are directly depends on the definition of these prototypes and their updating method. Hence, they are more compatible with finding clusters of similar time-series in time and preferably with equal length time-series because defining the prototype for elastic distance measures which handle the similarity in shape is not very straight forward which is discussed in Section 4.

5.3. Model-based clustering

Model-based clustering attempts to recover the original model from a set of data. This approach assumes a model for each cluster, and finds the best fit of data to that model. In detail, it presumes that there are some centroids chosen at random, and then some noise is added to them with a normal distribution. The model that is recovered from the generated data defines clusters [179]. Typically, model-based methods use either statistical approaches, e.g., COBWEB [180], or Neural Network approaches, e.g., ART [181] or Self-Organization Map [182]. In some of works in time-series clustering area, authors use Self-Organizing Maps (SOM) for clustering of time-series data. As mentioned, SOM is a model-based clustering based on neural networks, which is similar to processing that happens in the brain. For example, in [25], authors use SOM to cluster time-series features. However, because SOM needs to define the dimension of weight vector, it cannot work well with time-series of unequal length [16]. Additionally, there are a few articles which use model based clustering of time-series data which are composed of polynomial models [112], Gaussian mixed models [183], ARIMA [106], Markov chain [68] and Hidden Markov models [184,185]. In general, model based clustering has two drawbacks: first, it needs to set parameters

Please cite this article as: S. Aghabozorgi, et al., Time-series clustering – A decade review, Information Systems (2015), http://dx.doi.org/10.1016/j.is.2015.04.007

and it is based on user assumptions which may be false and consequently the result clusters would be inaccurate. Second, it has a slow processing time (especially neural networks) on large datasets [186].

5.4. Density-based clustering

In density based clustering, clusters are subspaces of dense objects which are separated by subspaces in which objects have low density. One of the famous algorithms which works by density-based concept is DBSCAN [187] where a cluster is expanded if its neighbours are dense. OPTICS [188] is another density-based algorithm which addresses the issue of detecting meaningful clusters in data of varying density. The model proposed by Chandrakala and Chandra [189] is one of the rare cases, where the authors propose a density based clustering method in kernel feature space for clustering multivariate time-series data of varying length. Additionally they present a heuristic method of fnding the initial values of the parameters used in their proposed algorithm. However, reviewing the literature it is noticeable that density-based clustering has not been used broadly for time-series data clustering because of its rather high complexity.

5.5. Grid-based clustering

The grid-based methods quantize the space into a finite number of the cells that form a grid, and then perform clustering on the grid's cells. STING [190] and Wave Cluster [191] are two typical examples of clustering algorithms which are based on grid-based concept. To the best of our knowledge, there is no work in the literature applying grid-based approaches for clustering of time-series. In Table 4 a summary of related works are mentioned based on the adopted representation method, distance measure, clustering algorithm and if it is applicable, definition of prototype.

Considering many works, it was understood that in most of models, the authors use time-series data as raw data or dimensionality reduced data, with standard traditional clustering algorithms. It is obvious that this type of analyzing time-series which use a brute-force approach without any optimization is a proper solution for scientific theories, but not for real world problems, because they are naturally very slow or inaccurate in large data bases. As a result, in many studies the attention of the researchers has drawn to using more customized algorithms for time-series data clustering as the ultimate solution.

In the following section, specific approaches are discussed and emphasize is on the solutions which are addressing the low quality of time-series clustering problems due to the mentioned issues in process of clustering.

5.6. Multi-step clustering

Although there are many studies to improve the quality of representation approaches, distance measurement, and prototypes, a few articles emphasis on enhancing algorithms and present a new model (usually as a hybrid method) for clustering of time-series data. In the following the most related works are presented and discussed:

1. Cheng-Ping Lai et al. [205] describe the problem of overlooking of information using dimension reduction. They claim that overlooked information could provide different meaning in time-series clustering results. To solve this issue, they adopt a two-level clustering method, where both the whole time-series and the subsequence of time-series are taken into account in the first and second level respectively. They used SAX transformation as dimension reduction method and CAST as clustering algorithm in the first level in order to group first-level data. In the second level, to measure distances between time-series, Dynamic Time Warping (DTW) has been used for varying length data, and Euclidean distance for equal length data. Finally, second-level data, of all the time-series, are then grouped by a clustering algorithm.
   In this study, the distance measure method used in order to find the first level result, is not clear while it is of great importance, because, for example, if the length of time-series are different (which is a possible case), it will effect on choosing dimension reduction and distance measurement methods. Another issue is that the authors have used CAST algorithm in their proposed approach for two times, once for making initial clusters, then for splitting each cluster into sub-clusters (although they used it 3 times in pseudo code). However, using CAST algorithm needs determining the threshold of affiliation which is a very sensitive parameter in this algorithm [212]. Additionally in this work, more granulated time-series are clustered which is actually based on the sub-sequence clustering. However, the work done by Keogh and Lin [73] indicates that subsequence clustering is meaningless. The authors in that work define "meaningless" as when the clustering output is independent of the input. And finally, their experimental result is not based on the published datasets in the literature. Therefore, there is not a way to compare their method with existing approaches for time-series clustering.
2. The authors in [206] also propose a new multi-level approach for shape based time-series clustering. In the first step, some candidate time-series are chosen from a made one-nearest neighbour network. In order to make the network of time-series, authors propose triangle distance for calculating similarity between time-series data. Then, hierarchical clustering is performed on chosen candidate time-series. To handle the shifts in time-series, Dynamic Time Warping (DTW) is utilized in the second step of clustering. Using this approach the size of data is reduced by approximately ten per cent.
   One of the issues in this algorithm is that it needs a nearest-neighbour network in the first level while complexity of making the nearest-neighbour network is O (n^2) which is very high. As a result, they try to reduce the search area by using k-Means as pre-clustering of data and limit the search only in each cluster to reduce the cost of network creation. However, because raw time-series is used in the process of pre-clustering to reduce the size of data, making the network itself is still very costly. As a result, the complexity of whole clustering is high which is not applicable on large datasets.
   Another problem is that pre-clusters developed in this model may not be accurate because the pre-clusters are

constructed by a non-elastic distance measure on raw time-series and it may be affected by outliers.
Although the experimental results are based on two syntactic datasets, however, the results should be tested on more datasets [6] because characteristics of time-series varies in different data-sets from different domains.
Finally, the error rate of choosing the candidates is computed but the quality of the final clusters has not measured using any standard and common metrics to be comparable with other methods.

3. In a group of works, an incremental clustering approach is adopted which exploit the multi-resolution characteristic of time-series data to cluster them in multi-step, for instance, Vlachos et al. [165] developed a method based on standard k-Means and Discrete Wavelet Transform (DWT) decomposition to cluster time-series data. They extended the k-Means algorithm to perform clustering of time-series incrementally at different resolutions of DWT decomposition. At first, they use Haar wavelet transformations to decompose all the time-series. After that, they apply the k-Means clustering on various regulations from a chaos to a finer level. At the end of each level, the extracted centers are reused as the initial centers for the next level of resolution. They doubled the center coordinates of each level because the length of a time-series is doubled in next level. In this algorithm, more and more detail are used during the clustering process. In order to compute the clustering error, they computed clustering error at the end of each level by summing up the number of objects clustered incorrectly divided by the cardinality of the dataset. In another similar work, Lin et al. [18] generalized this work and presented an anytime version of the partitioned clustering algorithm (k-mean and EM) for time-series. In this method also, authors use the multi-resolution property of wavelets in their algorithm. Following these works, Lin et al. in [213] present a multi-resolution clustering approach based on multi-resolution PAA (MPAA) for the incremental clustering algorithm of time-series.
Considering speed of clustering these approaches are quite good, however, in all these models, it is not clear that to what level it should be continued (the termination point). Additionally, in each iteration, all the time-series which are in the same resolution are re-clustered again. Therefore, the noise in some of them can affect the whole process. Moreover, this model is applicable only for partitioning clustering, which implies that it is not working for other types of algorithms such as arbitrary shape algorithms or hierarchical algorithms in the case where user needs the structure of data (the hierarchy of clusters). Another problem which these models should resolve is working with distance measures such as DTW which at first, are very costly and cannot be applied on whole dataset, and secondly, defining the prototypes using them is not a trivial task.
4. A new approach presented recently by Aghabozorgi and Wah [62] on co-movement of the stock market by using a three-phase method: (1) pre-clustering of time series; (2) purifying and summarization; and (3) merging. This new 3-PhaseTime series Clustering model (3PTC), can construct the clusters based on similarity in shape. This model facilitates the accurate clustering of time series data sets and is designed specifically for very large time series data sets. In the first phase of the model, data are pre-processed, transformed into a low dimensional space, and grouped approximately. Then, the pre-clustered time series are refined in the second phase using an accurate clustering method, and are represented by some prototypes. Finally, in the third phase, the prototypes are merged to construct the ultimate clusters. To evaluate the accuracy of the proposed model, the 3PTC is tested extensively using published time series data sets from diverse domains. results show the advantage of the proposed method wherein the analysis allows better prediction and understanding of the co-movement of companies even with local shifts.
5. In another work [211], a hybrid clustering algorithm called Two-step Time series Clustering (TTC) algorithm is proposed based on the similarity in shape of time series data. In this method, time series data are first grouped as subclusters based on similarity in time.The subclusters are then merged using the k-Medoids algorithm based on similarity in shape.This model has two contributions, first it is more accurate than other conventional and hybrid approaches and second, it determines the similarity in shape among time series data with a low complexity. To evaluate the accuracy of the proposed model, the model is tested extensively using syntactic and real-world time series datasets. The resutls in the experiments with various datasets and with different evaluation methods, show that TTC outperforms other conventional and hybrid clustering.

6. Time-series clustering evaluation measures

In this section evaluation method for clustering algorims are discussed. Keogh and Kasetty [6] have made an interesting research on different articles in time-series mining and conclude that the evaluation of time-series mining should follow some disciplines which are recommended as:

- The validation of algorithms should be performed on various ranges of datasets (unless the algorithm is created only for a specific set). The used dataset should be published and freely available
- Implementation bias must be avoided by careful design of the experiments
- If possible, data and algorithms should be freely provided
- New methods of similarity measures should be compared with simple and stable metrics such as Euclidean distance.

In general, evaluating of extracted clusters is not easy in the absence of data labels [26] and it is still an open problem. The definition of clusters depends on the user, the domain, and it is subjective. For example, the number of clusters, the size of clusters, definition for outliers, and definition of the similarity among the time-series in a problem are all the concepts which depend on the task at hand and should be declared subjectively. These have made the time-series clustering a big challenge in the data mining domain. However, owing to the classified data labelled by

Please cite this article as: S. Aghabozorgi, et al., Time-series clustering – A decade review, Information Systems (2015), http://dx.doi.org/10.1016/j.is.2015.04.007

human judge or by their generator (in synthetic datasets), the result can be evaluated by using some measures. The label of human judge is not perfect in terms of clustering raw data, but in practice it captures the strengths and shortcomings of the algorithms as ground truth. To evaluate MTC, the datasets are used from different domains which their labels are known. Fig. 7 shows the process for evaluation of a new model in time-series clustering.

Rand Index, Adjusted Rand Index, Entropy, Purity, Jacard, *F*-measure, FM, CSM, and MNI are used for the evaluation of MTC. All of these clustering evaluation criteria have values ranging from 0 to 1, where 1 corresponds to the case when ground truth and finding clusters are identical (except Entropy which is conversed and called cEntropy). Thus, here, bigger criteria values are preferred. Each of the mentioned evaluation criterion has its own benefit and there is no consensus of which criterion is better than other in the data mining community. Regarding to the time-series clustering algorithms, the evaluation measures employed in the different approaches are discussed in this section. Visualization and scalar measurements are the major technique for evaluation of clustering quality which also is known as clustering validity in some articles [214]. The techniques to evaluate any newly proposed model are explained in the following sections as is depicted in Fig. 8.

In scalar accuracy measurements, a single real number is generated to represent the accuracy of different clustering methods. Numerical measures that are applied to judge various aspects of cluster validity are classified into two types:

External Index: this index is used to measure the similarity of formed clusters to the externally supplied class labels or ground truth, and is the most popular clustering evaluation method [215]. In the literature, this index is known also as external criterion, external validation, extrinsic methods, and supervised methods because the ground truth is available.

Internal Index: this index is used to measure the goodness of a clustering structure without respect to external information. In the literature, this index is known also as internal criterion, internal validation, intrinsic and unsupervised methods.

These evaluation techniques are discussed in the rest of this section.

6.1. External index

External validity indices are the measures of the agreement between two partitions, one of which is usually a known/golden partition which is also known as ground truth (e.g., true class labels), and another is from the clustering procedure. Ground truth is the ideal clustering that is often built using human experts. In this type of evaluation, ground truth is available, and the index evaluates how well the clustering matches the ground truth [216]. Complete reviews and comparisons of some popular techniques exist in the literature [217–220]. However, there is not a compromise and universally accepted technique to evaluate clustering approaches, though there are many candidates which can be discounted for a variety of reasons. For external indices, usually match corresponding clusters and information theoretic are used as approach. Based on these approaches, many indices are presented in different articles [217,221].

Cluster purity: one of the ways to measure the quality of a clustering solution is cluster purity [222]. *Purity* is a simple and transparent evaluation measure. Considering $G=\{G_1, G_2, \ldots, G_M\}$ as ground truth clusters, and $C=\{C_1, C_2, \ldots, C_M\}$ as the clusters made by a clustering algorithm under evaluations, in order to compute the purity of cluster C with respect to G, each cluster is assigned to the class which is most frequent in the cluster, and then the accuracy of this assignment is measured by counting the number of correctly assigned objects and dividing by

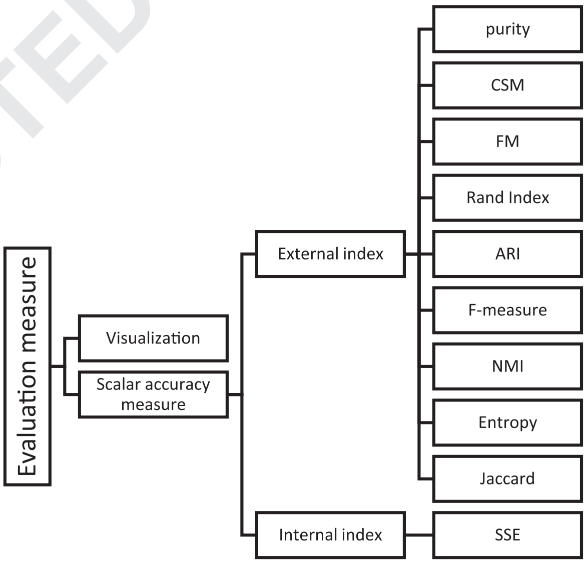

Fig. 8. Evaluation measure hierarchy used in the literature.

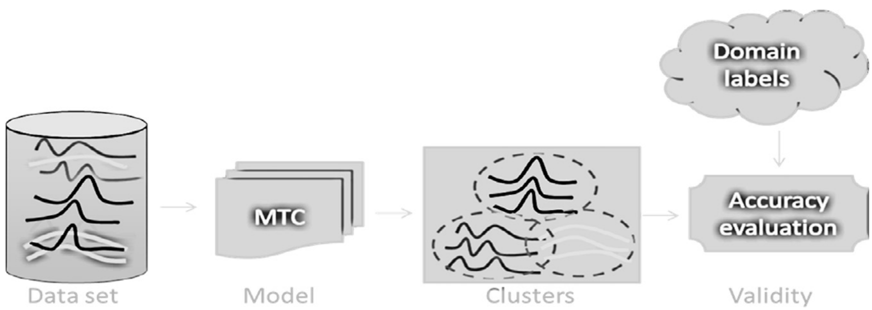

Fig. 7. Experimental evaluation of MTC.

Please cite this article as: S. Aghabozorgi, et al., Time-series clustering – A decade review, Information Systems (2015), http://dx.doi.org/10.1016/j.is.2015.04.007

number of objects in the cluster. A bad clustering has purity value close to 0, and a perfect clustering has a purity of 1. However, high purity is easy to achieve when the number of clusters is large, in particular, purity is 1 if each objects gets its own cluster. Thus, one cannot only rely on purity as the quality measure. Purity was used for evaluation of time-series clustering in different studies [4,21].

- **Cluster Similarity Measure (CSM):** CSM [16] is a simple metric used for validity of clusters in time-series domain [18,26,107,223].
- **Folkes and Mallow index (FM):** This metric is the index for computing the accuracy of time-series clustering in multimedia domain [26,83].
- **Jaccard Score:** Jaccard [224] is one of the metrics that has been used in various studies as external index [22,26,83].
- **Rand index (RI):** A popular quality measure [22,26,83] for evaluation of time-series clusters is the Rand index [225,226], which measures the agreement between two partitions and shows how much clustering results are close to the ground truth.
- **Adjusted Rand Index (ARI):** RI does not take a constant value (such as zero) two random clustering. Hence, in [227], authors suggest a corrected-for-chance version of the RI which works better than RI and many other indices [228,229]. This approach was used in gene expression domain successfully [230,231].
- ***F*-measure:** *F*-measure [232] is a well-established measure for assessing the quality of any given clustering solution with respect to ground truth. *F*-measure compares how closely each cluster matches a set of categories of ground truth. *F*-measure has been used in clustering of time-series data [22,66,233,234] and in natural language processing for evaluating clustering [235].
- **Normalized Mutual Information (NMI):** as mentioned, high purity in the large number of clusters is a drawback of purity measure. In order to make trade-off between the quality of the clustering against the number of clusters, NMI [236] is utilized as quality measure.in various studies [26,237,238]. Moreover, NMI can be used to compare clustering approaches with different numbers of clusters, because this measure is normalized [216].
- **Entropy:** entropy [239,240] of a cluster shows how dispersed classes are with a cluster (this should be low). Entropy is a function of the distribution of classes in the resulting clusters.

In short, one of the most popular approaches for quality evaluation of clusters is external indices to find how good the finding cluster results are [215] which also is used for evaluation of the proposed models in this study. However, it is not directly applicable in real-life unsupervised tasks, because the ground truth is not available for all datasets. Therefore, in the case that ground truth is not available, internal index is used which is discussed in following section.

6.2. Internal index

Typical objective functions in clustering, formalize the goal of attaining high intra-cluster similarity (objects within a cluster are similar) and low inter-cluster similarity (objects from different clusters are dissimilar). Internal validation compares solutions based on the goodness of fit between each clustering and the data. Internal validity indices evaluate clustering results by using only features and information inherent in a dataset. They are usually used in the case that true solutions (ground truth) are unknown. However, this index can only make comparisons between different clustering approaches that are generated using the same model/metric. Otherwise, it makes assumptions about cluster structure.

There are many internal indices such as Sum of Squared Error, Silhouette index, Davies-Bouldin, Calinski-Harabasz, Dunn index, R-squared index, Hubert-Levin (C-index), Krzanowski-Lai index, Hartigan index, Root-Mean-Square Standard Deviation (RMSSTD) index, Semi-Partial R-squared (SPR) index, Distance between two clusters (CD) index, Weighted inter-intra index, Homogeneity index, and Separation index. Sum of Squared Error (SSE) is an objective function that describes the coherence of a given cluster, "better" clusters are expected to give lower SSE values [241]. For evaluation of clusters in terms of accuracy, the Sum of Squared Error (SSE) can be used as the most common measure in different works [18,165]. For each time-series, the error is the distance to the nearest cluster.

7. Conclusion

Although different researches have been conducted on time-series clustering, the unique characteristics of time-series data are barriers that fail most of conventional clustering algorithms to work well for time-series. In particular, the high dimensionality, very high feature correlation, and typically large amount of noise that characterize time-series data have been viewed as an interesting research challenge in time-series clustering. Accordingly, most of the studies in the literature have concentrated on two subroutines of clustering:

1. A vast number of researches have focused on high dimensional characteristic of time-series data and tried to present a way of representing time-series in a lower dimension compatible with conventional clustering algorithms.
2. Different efforts have been taken on presenting a distance measurement based on raw time-series or the represented data.

The common characteristic in both above approaches is clustering of the transferred, extracted or raw time-series using conventional clustering algorithms such as k-Means, k-Medoid or hierarchical clustering. However, most of them suffer from neglecting the data which is caused by dimensionality reduction, inaccurate similarity calculation due to high complexity of accurate measures, and lack of quality in clustering algorithms because of their nature which is suitable for static data.

Highlighting the four representation methods discussed in this article it can be concluded that the main goal of data adaptive methods is to minimize the global reconstruction error using arbitrary length segments. They are better in approximating each series but when there is several time-

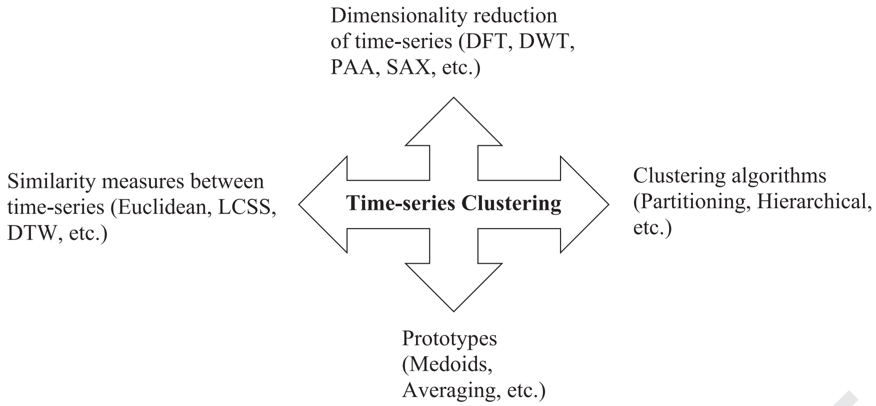

Fig. 9. four aspect of studying time-series clustering.

series they face difficulty. At the other hand, non-data adaptive methods are suitable for the fixed size time-series and model based approaches represent the time series in stochastic ways. In these three approaches user can define the compression-ratio based on the application in hand while in data dictated approaches, the compression-ratio is defined automatically based on raw time-series.

At the other hand, one of the important challenges in choosing representation methods is to have a compatible and appropriate similarity measure. Reviewing and comparing available similarity measures in this study revealed that the most effective and accurate approaches are those which are based on dynamic programming (DP) which are expensive in computation and their complexity needs to be tuned and handled before application. After all, literature shows that the most popular similarity measures in time-series clustering are Euclidean distance and DTW.

Another challenging issue which can affect the accuracy of clustering is choosing the appropriate prototype. The most commonly used prototype is medoid while using Averaging method is scarce, because it is limited to be used for time-series with equal length and with using non-elastic distance measures. After all, results show that the best clustering accuracy among other prototypes mentioned in this study belong to the local search prototype.

Finally reviewing time-series clustering algorithms reveals that comparing to other algorithms; partitioning algorithms are widely used because of their fast response. However, as the number of clusters needs to be pre-assigned, these algorithms are not applicable in most real world applications. In addition, because of their dependency to prototypes, they are more suitable for clustering equal length time-series. Hierarchical clustering at the other hand doesn't need the number of clusters to be pre-defined and also it has a great visualization power in time-series clustering and is a prefect tool for evaluation of dimensionality reduction or distance metrics and also the ability to cluster time-series with unequal length is its other superiority in comparison to partitioning algorithms as well. But hierarchical clustering is restricted to the small datasets because of its quadratic computational complexity. Model based and density based algorithms usage is scarce for the same problem of slow process and high complexity. In addition model based algorithms are suffering from their dependence on user assumptions for parameters. Recently few studies are focusing on improving and enhancing algorithms by representing new models which are mostly based on combination of different algorithms as hybrid or multistep clustering algorithms.

Further research works on time-series representation can address unattended or barely attended areas such as multivariate time series data with different length, unevenly sampled data and discrete valued time-series. In terms of similarity measures, many of proposed similarity measures do not show any improvements to Euclidean distance and as experiments in [6] shows their error rates are even worse. Consequently still the need for more precise similarity measure is not fulfilled. The same story goes to cluster prototypes, although a lot of studies are conducted, still none of them could beat the medoid and averaging prototypes which are the most used approaches.

Actually, by assuming that time-series clustering can be improved by advancements in four different aspects as is represented in Fig. 9, considering the literature, it can be concluded that most of the studies are focusing on improving representation methods, distance measurement methods, and prototypes while the portion of enhancing clustering approaches is approximately less than 10% in comparison with other parts:

Among a few approaches and algorithms which have been proposed for time-series clustering, there are some studies which have taken explicit or implicit strategies for increasing the quality (considering the scalability). However, as clustering approaches are either accurate which are constructed expensively, or inaccurate but made inexpensively, one still can see the problem of low quality or lack of meaningfulness in the clusters. In brief, although there are opportunities for improvement in all four aspect of time-series clustering, it can be concluded that the main opportunity for future works in this filed could be working on new hybrid algorithms with using existing or new clustering approaches in order to balance the quality and the expenses of clustering time-series.

References

[1] P. Rai, S. Singh, A survey of clustering techniques, Int. J. Comput. Appl. 7 (12) (2010) 1–5.
[2] V. Niennattrakul, C. Ratanamahatana, On clustering multimedia time series data using k-means and dynamic time warping, in: Proceedings

of the International Conference on Multimedia and Ubiquitous Engineering, 2007, MUE '07, 2007, pp. 733–738.

[3] C. Ratanamahatana, Multimedia retrieval using time series representation and relevance feedback, in: Proceedings of 8th International Conference on Asian Digital Libraries (ICADL2005), 2005, pp. 400–405.

[4] C. Ratanamahatana, V. Niennattrakul, Clustering multimedia data using time series, in: Proceedings of the International Conference on Hybrid Information Technology, 2006, ICHIT '06, 2006, pp. 372–379.

[5] J. Lin, E. Keogh, S. Lonardi, J. Lankford, D. Nystrom, Visually mining and monitoring massive time series, in: Proceedings of 2004 ACM SIGKDD International Conference on Knowledge Discovery and data Mining – KDD '04, 2004, p. 460.

[6] E. Keogh, S. Kasetty, On the need for time series data mining benchmarks: a survey and empirical demonstration, Data Min. Knowl. Discov. 7 (4) (2003) 349–371.

[7] K. Haigh, W. Foslien, and V. Guralnik, Visual query language: finding patterns in and relationships among time series data, Seventh Workshop on Mining Scientific And Engineering Datasets, 2004, pp. 324–332.

[8] E. Keogh, S. Chu, D. Hart, Segmenting time series: a survey and novel approach, Data Min. Time Ser. Databases 57 (1) (2004) 1–21.

[9] J. Lin, E. Keogh, S. Lonardi, and B. Chiu, A symbolic representation of time series, with implications for streaming algorithms, in: Proceedings of 8th ACM SIGMOD Workshop on Research Issues Data Mining and Knowledge Discovery – DMKD '03, 2003, p. 2.

[10] J. Zakaria, S. Rotschafer, A. Mueen, K. Razak, E. Keogh, Mining massive archives of mice sounds with symbolized representations, in: SIGKDD, 2012, pp. 1–10.

[11] T. Rakthanmanon, A.B. Campana, G. Batista, J. Zakaria, E. Keogh, Searching and mining trillions of time series subsequences under dynamic time warping, in: proceedings of the Conference on Knowledge Discovery and Data Mining, 2012, pp. 262–270.

[12] E. Keogh, A decade of progress in indexing and mining large time series databases, in: Proceedings of the International Conference on Very Large Data Bases (VLDB), 2006, pp. 1268–1268.

[13] S. Laxman, P.S. Sastry, A survey of temporal data mining, Sadhana 31 (2) (2006) 173–198.

[14] V. Kavitha, M. Punithavalli, Clustering time series data stream—a literature survey, Int. J. Comput. Sci. Inf. Secur. 8 (1) (2010) arxiv:1005.4270.

[15] C. Antunes, A.L. Oliveira, Temporal data mining: an overview, in: KDD Workshop on Temporal Data Mining, 2001, pp. 1–13.

[16] T. Warrenliao, Clustering of time series data—a survey, Pattern Recognit. 38 (11) (2005) 1857–1874.

[17] S. Rani, G. Sikka, Recent techniques of clustering of time series data: a survey, Int. J. Comput. Appl 52 (15) (2012) 1–9.

[18] J. Lin, M. Vlachos, E. Keogh, D. Gunopulos, Iterative incremental clustering of time series, Adv. Database Technol 2004 (2004) 521–522.

[19] R. Kumar, P. Nagabhushan, Time series as a point—a novel approach for time series cluster visualization in: Proceedings of the Conference on Data Mining, 2006, pp. 24–29.

[20] C. Faloutsos, M. Ranganathan, Y. Manolopoulos, Fast subsequence matching in time-series databases, ACM SIGMOD Rec. 23 (2) (1994) 419–429.

[21] X. Wang, K. Smith, R. Hyndman, Characteristic-based clustering for time series data, Data Min. Knowl. Discov. 13 (3) (2006) 335–364.

[22] M. Chiş, S. Banerjee, A.E. Hassanien, Clustering time series data: an evolutionary approach, Found. Comput. Intell. 6 (1) (2009) 193–207.

[23] J. Lin, E. Keogh, W. Truppel, Clustering of streaming time series is meaningless, in: Proceedings of 8th ACM SIGMOD Workshop on Research Issues Data Mining and Knowlegde Discovery DMKD 03, 2003, p. 56.

[24] E. Keogh, M. Pazzani, K. Chakrabarti, S. Mehrotra, A simple dimensionality reduction technique for fast similarity search in large time series databases, Knowl. Inf. Syst. 1805 (1) (2000) 122–133.

[25] X. Wang, K.A. Smith, R. Hyndman, D. Alahakoon, A Scalable Method for Time Series Clustering, 2004.

[26] H. Zhang, T.B. Ho, Y. Zhang, M.S. Lin, Unsupervised feature extraction for time series clustering using orthogonal wavelet transform, Informatica 30 (3) (2006) 305–319.

[27] H. Wang, W. Wang, J. Yang, P.P.S. Yu, Clustering by pattern similarity in large data sets, in: Proceedings of 2002 ACM SIGMOD International Conference Management data – SIGMOD '02, vol. 2, 2002, p. 394.

[28] G. Das, K.I. Lin, H. Mannila, G. Renganathan, P. Smyth, Rule discovery from time series,, Knowl. Discov. Data Min (1998) 16–22.

[29] T.C. Fu, F.L. Chung, V. Ng, R. Luk, Pattern discovery from stock time series using self-organizing maps, in: Workshop Notes of KDD2001 Workshop on Temporal Data Mining, 2001, pp. 26–29.

[30] B. Chiu, E. Keogh, S. Lonardi, Probabilistic discovery of time series motifs, in: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2003, pp. 493–498.

[31] E. Keogh, S. Lonardi, B.Y. Chiu, Finding surprising patterns in a time series database in linear time and space, in: Proceedings of the Eighth ACM SIGKDD, 2002, pp. 550–556.

[32] P.K. Chan, M.V. Mahoney, Modeling multiple time series for anomaly detection, in: Proceedings of Fifth IEEE International Conference on Data Mining, 2005, pp. 90–97.

[33] L. Wei, N. Kumar, V. Lolla, E. Keogh, Assumption-free anomaly detection in time series, in: Proceedings of the 17th International Conference on Scientific and Statistical Database Management, 2005, pp. 237–240.

[34] M. Leng, X. Lai, G. Tan, X. Xu, Time series representation for anomaly detection, in: Proceedings of 2nd IEEE International Conference on Computer Science and Information Technology, 2009, ICCSIT 2009, 2009, pp. 628–632.

[35] P.M. Polz, E. Hortnagl, E. Prem, Processing and Clustering Time Series of Mobile Robot Sensory Data. Technical Report, Österreichisches Forschungsinstitut für Artificial Intelligence, Wien, TR-2003-10, 2003, 2003.

[36] W. He, G. Feng, Q. Wu, T. He, S. Wan, J. Chou, A new method for abrupt dynamic change detection of correlated time series,, Int. J. Climatol. 32 (10) (2011) 1604–1614.

[37] A. Sfetsos, C. Siriopoulos, Time series forecasting with a hybrid clustering scheme and pattern recognition, IEEE Trans. Syst. Man Cybern 34 (3) (2004) 399–405.

[38] N. Pavlidis, V.P. Plagianakos, D.K. Tasoulis, M.N. Vrahatis, Financial forecasting through unsupervised clustering and neural networks, Oper. Res. 6 (2) (2006) 103–127.

[39] F. Ito, T. Hiroyasu, M. Miki, H. Yokouchi, Detection of Preference Shift Timing using Time-Series Clustering, 2009, pp. 1585–1590.

[40] W. Pedrycz, D. Graves, Proximity fuzzy clustering and its application to time series clustering and prediction in: Proceedings of the 2010 10th International Conference on Intelligent Systems Design and Applications ISDA10, 2010, pp. 49–54.

[41] U. Rebbapragada, P. Protopapas, C.E. Brodley, C. Alcock, Finding anomalous periodic time series, Mach. Learn. 74 (3) (2009) 281–313.

[42] N. Subhani, L. Rueda, A. Ngom, C.J. Burden, Multiple gene expression profile alignment for microarray time-series data clustering, Bioinformatics 26 (18) (2010) 2281–2288.

[43] A. Fujita, P. Severino, K. Kojima, J.R. Sato, A.G. Patriota, S. Miyano, Functional clustering of time series gene expression data by Granger causality, BMC Syst. Biol. 6 (1) (2012) 137.

[44] C. Möller-Levet, F. Klawonn, K.H. Cho, O. Wolkenhauer, Fuzzy clustering of short time-series and unevenly distributed sampling points, Adv. Intell. Data Anal. (2003) 330–340.

[45] J. Ernst, G.J. Nau, Z. Bar-Joseph, Clustering short time series gene expression data, Bioinforma. 21 (Suppl. 1) (2005) i159–i168. 21.

[46] Mikhail Pyatnitskiy, I. Mazo, M. Shkrob, E. Schwartz, E. Kotelnikova, Clustering gene expression regulators: new approach to disease subtyping, PLoS One 9 (1) (2014) e84955.

[47] M. Steinbach, P.N. Tan, V. Kumar, S. Klooster, and C. Potter, Discovery of climate indices using clustering, in: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery And data Mining, 2003, pp. 446–455.

[48] M. Ji, F. Xie, Y. Ping, A dynamic fuzzy cluster algorithm for time series, Abstr. Appl. Anal. 2013 (2013) 1–7.

[49] M.A. Elangasinghe, N. Singhal, K.N. Dirks, J.A. Salmond, S. Samarasinghe, Complex time series analysis of PM10 and PM2.5 for a coastal site using artificial neural network modelling and k-means clustering, Atmos. Environ. 94 (2014) 106–116.

[50] K. Košmelj and V. Batagelj, Cross-sectional approach for clustering time varying data J. Classif., 1990.

[51] F. Iglesias, W. Kastner, Analysis of similarity measures in times series clustering for the discovery of building energy patterns, Energies 6 (2) (2013) 579–597.

[52] M.G. Scotto, A.M. Alonso, S.M. Barbosa, Clustering time series of sea levels: extreme value approach, J. Waterw. Port, Coastal, Ocean Eng. 136 (4) (2010) 215–225.

[53] R.H.R. Shumway, Time-frequency clustering and discriminant analysis, Stat. Probab. Lett 63 (3) (2003) 307–314.

[54] Shen Liu, E.A. Maharaj, B. Inder, Polarization of forecast densities: a new approach to time series classification, Comput. Stat. Data Anal. 70 (2014) 345–361.
[55] Y. Sadahiro, T. Kobayashi, Exploratory analysis of time series data: detection of partial similarities, clustering, and visualization, Comput. Environ. Urban Syst. 45 (2014) 24–33.
[56] M. Gorji Sefidmazgi, M. Sayemuzzaman, A. Homaifar, M.K. Jha, S. Liess, Trend analysis using non-stationary time series clustering based on the finite element method, Nonlinear Process. Geophys. 21 (3) (2014) 605–615.
[57] M. Kumar, N.R. Patel, Clustering seasonality patterns in the presence of errors, in: Proceedings of Eighth ACM SIGKDD, 2002, pp. 557–563.
[58] A.J. Bagnall, G. Janacek, B. De la Iglesia, M. Zhang, Clustering time series from mixture polynomial models with discretised data in: Proceedings of the Second Australasian Data Mining Workshop, 2003, pp. 105–120.
[59] H. Guan, Q. Jiang, Cluster financial time series for portfolio, in: Proceedings of the International Conference on Wavelet Analysis and Pattern Recognition, 2007, pp. 851–856.
[60] C. Guo, H. Jia, N. Zhang, Time series clustering based on ICA for stock data analysis, in: Proceedings of 4th International Conference on Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08, 2008, pp. 1–4.
[61] A. Stetco, X. Zeng, J. Keane, Fuzzy cluster analysis of financial time series and their volatility assessment, in: Proceedings of 2013 IEEE International Conference on Systems, Man, and Cybernetics, 2013, pp. 91–96.
[62] S. Aghabozorgi, T. Ying Wah, Stock market co-movement assessment using a three-phase clustering method, Expert Syst. Appl. 41 (4) (2014) 1301–1314.
[63] Y.-C. Hsu, A.-P. Chen, A clustering time series model for the optimal hedge ratio decision making, Neurocomputing 138 (2014) 358–370.
[64] A. Wismüller, O. Lange, D.R. Dersch, G.L. Leinsinger, K. Hahn, B. Pütz, D. Auer, Cluster analysis of biomedical image time-series, Int. J. Comput. Vis 46 (2) (2002) 103–128.
[65] M. van den Heuvel, R. Mandl, Pol, H. Hulshoff, Normalized cut group clustering of resting-state fMRI data, PLoS One 3 (4) (2008) e2001.
Q8 [66] F. Gullo, G. Ponti, A. Tagarelli, G. Tradigo, P. Veltri, A time series approach for clustering mass spectrometry data, J. Comput. Sci. (2011). 2010.
[67] V. Kurbalija, J. Nachtwei, C. Von Bernstorff, C. von Bernstorff, H.-D. Burkhard, M. Ivanović, L. Fodor, Time-series mining in a psychological domain, in: Proceedings of the Fifth Balkan Conference in Informatics, 2012, pp. 58–63.
[68] M. Ramoni, P. Sebastiani, P. Cohen, Multivariate clustering by dynamics, in :Proceedings of the national Conference on Artificial Intelligence, 2000, pp. 633–638.
[69] Gopalapillai, Radhakrishnan, D. Gupta, T.S.B. Sudarshan, Experimentation and analysis of time series data for rescue robotics, Recent Adv Intell Inf (2014) 443–453.
[70] D. Tran, M. Wagner, Fuzzy c-means clustering-based speaker verification, Adv. Soft Comput. 2002 2275 (2002) 363–369.
Q10 Q9 [71] Fong, Simon, Using hierarchical time series clustering algorithm and wavelet classifier for biometric voice classification, J. Biomed. Biotechnol. (2012) 215019.
[72] J. Zhu, B. Wang, B. Wu, Social network users clustering based on multivariate time series of emotional behavior, J. China Univ. Posts Telecommun 21 (2) (2014) 21–31.
[73] E. Keogh, J. Lin, Clustering of time-series subsequences is meaningless: implications for previous and future research, Knowl. Inf. Syst. 8 (2) (2005) 154–177.
[74] A. Gionis, H. Mannila, Finding recurrent sources in sequences, in: Proceedings of the Seventh Annual International Conference on RESEARCH in Computational Molecular Biology, 2003, pp. 123–130.
[75] A. Ultsch, F. Mörchen, ESOM-Maps: Tools for Clustering, Visualization, and Classification with Emergent SOM, 2005.
[76] F. Morchen, A. Ultsch, F. Mörchen, O. Hoos, Extracting interpretable muscle activation patterns with time series knowledge mining, J. Knowl. BASED 9 (3) (2005) 197–208.
[77] V. Hautamaki, P. Nykanen, P. Franti, Time-series clustering by approximate prototypes, in: Proceedings of 19th International Conference on Pattern Recognition, 2008, ICPR 2008, 2008, no. D, pp. 1–4.
[78] M. Vlachos, D. Gunopulos, G. Das, "Indexing time-series under conditions of noise,", in: M. Last, A. Kandel, H. Bunke (Eds.), Data Mining in Time Series Databases, World Scientific, Singapore, 2004, p. 67.
Q11 [79] T. Mitsa, Temporal Data Mining, vol. 33, Chapman & Hall/CRC, 2009.
[80] E. Keogh, Hot sax: efficiently finding the most unusual time series subsequence, in: Proceedings of Fifth IEEE International Conference on Data Mining ICDM05, 2005, pp. 226–233.
[81] E. Ghysels, P. Santa-Clara, R. Valkanov, Predicting volatility: getting the most out of return data sampled at different frequencies, J. Econom 131 (1–2) (2006) 59–95.
[82] G. Duan, Y. Suzuki, K. Kawagoe, Grid representation of time series data for similarity search, in: The institute of Electronic, Information, and Communication Engineer, 2006.
[83] C. Ratanamahatana, E. Keogh, A.J. Bagnall, S. Lonardi, A novel bit level time series representation with implications for similarity search and clustering, in: Proceedings of 9th Pacific-Asian International Conference on Knowledge Discovery and Data Mining (PAKDD'05), 2005, pp. 771–777.
[84] J. Lin, E. Keogh, L. Wei, S. Lonardi, Experiencing SAX: a novel symbolic representation of time series", Data Min. Knowl. Discov. 15 (2) (2007) 107–144.
[85] K. Chan, A.W. Fu, Efficient time series matching by wavelets, in: Proceedings of 1999 15th International Conference on Data Engineering, vol. 15, no. 3, 1999, pp. 126–133.
[86] E. Keogh, M. Pazzani, An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback, in: Proceedings of the 4th International Conference of Knowledge Discovery and Data Mining, 1998, pp. 239–241.
[87] E. Keogh, K. Chakrabarti, M. Pazzani, S. Mehrotra, Locally adaptive dimensionality reduction for indexing large time series databases,, ACM SIGMOD Rec 27 (2) (2001) 151–162.
[88] I. Popivanov, R.J. Miller, Similarity search over time-series data using wavelets, in: ICDE '02: Proceedings of the 18th International Conference on Data Engineering, 2002, pp. 212–224.
[89] Y.L. Wu, D. Agrawal, A. El Abbadi, " comparison of DFT and DWT based similarity search in time-series databases, in: Proceedings of the Ninth International Conference on Information and Knowledge Management, 2000, pp. 488–495.
[90] B.K. Yi, C. Faloutsos, Fast time sequence indexing for arbitrary Lp norms, in: Proceedings of the 26th International Conference on Very Large Data Bases, 2000, pp. 385–394.
[91] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, E. Keogh, Querying and mining of time series data: experimental comparison of representations and distance measures,, Proc. VLDB Endow 1 (2) (2008) 1542–1552.
[92] A.A.J. Bagnall, C. "Ann" Ratanamahatana, E. Keogh, S. Lonardi, G. Janacek, A bit level representation for time series data mining with shape based similarity, Data Min. Knowl. Discov. 13 (1) (2006) 11–40.
[93] J. Shieh, E. Keogh, iSAX: disk-aware mining and indexing of massive time series datasets, Data Min. Knowl. Discov. 19 (1) (2009) 24–57.
[94] X. Wang, A. Mueen, H. Ding, G. Trajcevski, P. Scheuermann, E. Keogh, Experimental comparison of representation methods and distance measures for time series data, Data Min. Knowl. Discov. (2012) p. Springer Netherlands,.
[95] Y. Morinaka, M. Yoshikawa, T. Amagasa, S. Uemura, The L-index: an indexing structure for efficient subsequence matching in time sequence databases, in: Proceedings of 5th PacificAisa Conference on Knowledge Discovery and Data Mining, 2001, pp. 51–60.
[96] H. Shatkay S.B. Zdonik, Approximate queries and representations for large data sequences, in: Proceedings of the Twelfth International Conference on Data Engineering, 1996, pp. 536–545.
[97] F. Korn, H.V. Jagadish, C. Faloutsos, Efficiently supporting ad hoc queries in large datasets of time sequences, ACM SIGMOD Record 26 (1997) 289–300.
[98] F. Portet, E. Reiter, A. Gatt, J. Hunter, S. Sripada, Y. Freer, C. Sykes, Automatic generation of textual summaries from neonatal intensive care data, Artif. Intell. 173 (7) (2009) 789–816.
[99] Y. Cai and R. Ng, Indexing spatio-temporal trajectories with Chebyshev polynomials, in: Procedings of 2004 ACM SIGMOD International, 2004, p. 599.
[100] E. Bingham, Random projection in dimensionality reduction: applications to image and text data, in: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2001, pp. 245–250.
[101] Q. Chen, L. Chen, X. Lian, Y. Liu, Indexable PLA for efficient similarity search, in: Proceedings of the 33rd International Conference on Very large Data Bases, 2007, pp. 435–446.
[102] D. Minnen, T. Starner, M. Essa, C. Isbell, Discovering characteristic actions from on-body sensor data, in: Proceedings of 10th IEEE International Symposium on Wearable Computers, 2006, pp. 11–18.

Please cite this article as: S. Aghabozorgi, et al., Time-series clustering – A decade review, Information Systems (2015), http://dx.doi.org/10.1016/j.is.2015.04.007

[103] D. Minnen, C.L. Isbell, I. Essa, T. Starner, Discovering multivariate motifs using subsequence density estimation and greedy mixture learning, Proc. Natl. Conf. Artif. Intell. 22 (1) (2007) 615.
[104] A. Panuccio, M. Bicego, and V. Murino, A Hidden Markov Model-based approach to sequential data clustering, in Structural, Syntactic, and Statistical Pattern Recognition, T. Caelli, A. Amin, R. Duin, R. De, and M. Kamel, Eds. 2002.
[105] N. Kumar, N. Lolla, E. Keogh, S. Lonardi, Time-series bitmaps: a practical visualization tool for working with large time series databases, SIAM 2005 Data Min (2005) 531–535.
[106] M. Corduas, D. Piccolo, Time series clustering and classification by the autoregressive metric, Comput. Stat. Data Anal. 52 (4) (2008) 1860–1872.
[107] K. Kalpakis, D. Gada, V. Puttagunta, Distance measures for effective clustering of ARIMA time-series, in: Proceedings 2001 IEEE International Conference on Data Mining, 2001, pp. 273–280.
[108] R. Agrawal, C. Faloutsos, A. Swami, Efficient similarity search in sequence databases, Found. Data Organ. Algorithms 46 (1993) 69–84.
[109] K. Kawagoe, T. Ueda, A similarity search method of time series data with combination of Fourier and wavelet transforms, in: Proceedings Ninth International Symposium on Temporal Representation and Reasoning, 2002, 86–92.
[110] F.L. Chung, T.C. Fu, R. Luk, Flexible time series pattern matching based on perceptually important points, in: Jt. Conference on Artificial Intelligence Workshop, 2001, pp. 1–7.
[111] E. Keogh, S. Lonardi, C. Ratanamahatana, Towards parameter-free data mining, in: Proceedings of Tenth ACM SIGKDD International Conference on Knowledge Discovery Data Mining, vol. 22, no. 25, 2004, pp. 206–215.
[112] A.J. Bagnall, G. Janacek, Clustering time series with clipped data, Mach. Learn. 58 (2) (2005) 151–178.
[113] W.G. Aref, M.G. Elfeky, A.K. Elmagarmid, Incremental, online, and merge mining of partial periodic patterns in time-series databases, Trans. Knowl. Data Eng 16 (3) (2004) 332–342.
[114] S. Chu, E. Keogh, D. Hart, M. Pazzani, et al., Iterative deepening dynamic time warping for time series, in: Proceedings of the Second SIAM International Conference on Data Mining, 2002, pp. 195–212.
[115] C. Ratanamahatana, E. Keogh, Three myths about dynamic time warping data mining, in: Proceedings of the International Conference on Data Mining (SDM'05), 2005, pp. 506–510.
[116] P. Smyth, Clustering sequences with hidden Markov models,, Adv. Neural Inf. Process. Syst 9 (1997) 648–654.
[117] Y. Xiong, D.Y. Yeung, Mixtures of ARMA models for model-based time series clustering, Data Min, 2002. ICDM 2003 (2002) 717–720.
[118] H. Sakoe, S. Chiba, A dynamic programming approach to continuous speech recognition,, Proceedings of the Seventh International Congress on Acoustics vol. 3 (1971) 65–69.
[119] H. Sakoe, S. Chiba, Dynamic programming algorithm optimization for spoken word recognition,, IEEE Trans. Acoust. Speech Signal Process 26 (1) (1978) 43–49.
[120] M. Vlachos, G. Kollios, D. Gunopulos, Discovering similar multidimensional trajectories, in: Proceedingsof 18th International Conference on Data Engineering, 2002, pp. 673–684.
[121] A. Banerjee, J. Ghosh, Clickstream clustering using weighted longest common subsequences, in: Proceedings of the Workshop on Web Mining, SIAM Conference on Data Mining, 2001, pp. 33–40.
[122] L.J. Latecki, V. Megalooikonomou, Q. Wang, R. Lakaemper, C. Ratanamahatana, E. Keogh, Elastic partial matching of time series,, Knowl. Discov. Databases PKDD 2005 (2005) 577–584.
[123] E. Keogh, S. Lonardi, C. Ratanamahatana, L. Wei, S.H. Lee, J. Handley, Compression-based data mining of sequential data,, Data Min. Knowl. Discov. 14 (1) (2007) 99–129.
[124] J.L. Rodgers, W.A. Nicewander, Thirteen ways to look at the correlation coefficient,, Am. Stat 42 (1) (1988) 59–66.
[125] P. Indyk, N. Koudas, Identifying representative trends in massive time series data sets using sketches, in: Proceedings of 26th International Conference on Very Large Data Bases, 2000, pp. 363–372.
[126] Y. Huhtala, J. Karkkainen, Mining for similarities in aligned time series using wavelets, Data Mining and Knowl. Discov. Theory Tools Technol. (1999) 105–393.
[127] M. Last, A. Kandel, Data mining in time series databases, World Sci. (2004).
[128] Z. Zhang, K. Huang, T. Tan, Comparison of similarity measures for trajectory clustering in outdoor surveillance scenes, in: Proceedings of 18th International Conference on Pattern Recognition, ICPR 2006, vol. 3, pp. 1135–1138, 2006.
[129] J. Aach, G.M. Church, Aligning gene expression time series with time warping algorithms, Bioinformatics 17 (6) (2001) 495.
[130] R. Dahlhaus, On the Kullback–Leibler information divergence of locally stationary processes, Stoch. Process. Appl 62 (1) (1996) 139–168.
[131] E. Keogh, A probabilistic approach to fast pattern matching in time series databases, in: Proceedings of the 3rd International Conference of Knowledge Discovery and Data Mining, 1997, pp. 52–57.
[132] X. Golay, S. Kollias, G. Stoll, D. Meier, A. Valavanis, P. Boesiger, A new correlation-based fuzzy logic clustering algorithm for FMRI,, Magn. Reson. Med. 40 (2) (1998) 249–260.
[133] C. Wang, X. Sean Wang, Supporting content-based searches on time series via approximation,, Sci Stat Database (2000) 69–81.
[134] L. Chen R. Ng, On the marriage of lp-norms and edit distance, in: Proceedings of the Thirtieth International Conference on Very Large Data Bases-Volume 30, 2004, pp. 792–803.
[135] L. Chen, M.T. Özsu, V. Oria, Robust and fast similarity search for moving object trajectories, in: Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data, 2005, pp. 491–502.
[136] L. Chen, M.T. Özsu, Using multi-scale histograms to answer pattern existence and shape match queries,, Time 2 (1) (2005) 217–226.
[137] J. Aßfalg, H.P. Kriegel, P. Kröger, P. Kunath, A. Pryakhin, M. Renz, Similarity search on time series based on threshold queries, Adv. Database Technol. 2006 (2006) 276–294.
[138] E. Frentzos, K. Gratsias, Y. Theodoridis, Index-based most similar trajectory search, in: Proceedings of 23rd International Conference on Data Engineering, 2007, ICDE 2007. IEEE, 2007, pp. 816–825.
[139] M.D. Morse, J.M. Patel, An efficient and accurate method for evaluating time series similarity, in: Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data SIGMOD 07, 2007, p. 569.
[140] Y. Chen, M.A. Nascimento, B.C. Ooi,A.K.H. Tung, Spade: on shape-based pattern detection in streaming time series, in: Proceedings of IEEE 23rd International Conference on Data Engineering, 2007. ICDE 2007. , 2007, pp. 786–795.
[141] X. Zhang, J. Wu, X. Yang, H. Ou, T. Lv, A notime series classificationvel pattern extraction method for, Optim. Eng. 10 (2) (2009) 253–271.
[142] W. Lang, M. Morse, J.M. Patel, Dictionary-based compression for long time-series similarity, Knowl. Data Eng. IEEE Trans 22 (11) (2010) 1609–1622.
[143] S. Salvador, P. Chan, Toward accurate dynamic time warping in linear time and space, Intell. Data Anal. 11 (5) (2007) 561–580.
[144] F. Itakura, Minimum prediction residual principle applied to speech recognition. Minimum prediction residual principle applied to speech recognition, IEEE Trans. Acoust. Speech Signal Process 23 (1) (1975) 67–72.
[145] B. Lkhagva, Y.u. Suzuki, K. Kawagoe, New time series data representation ESAX for financial applications, in: Proceedings of 22nd International Conference on Data Engineering Workshops, 2006, pp. 17–22.
[146] A. Corradini, Dynamic time warping for off-line recognition of a small gesture vocabulary, in: IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, 2001, pp. 82–89.
[147] L. Rabiner, S. Levinson, Speaker-independent recognition of isolated words using clustering techniques,, IEEE Trans. Acoust. Speech Signal Process 27 (4) (1979) 336–349.
[148] D. Gusfield, Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology, Cambridge University Press, 1997.
[149] V. Niennattrakul, C. Ratanamahatana, Inaccuracies of shape averaging method using dynamic time warping for time series data, Comput. Sci. 2007 (2007) 513–520.
[150] L. Kaufman, P.J. Rousseeuw, E. Corporation, Finding groups in data: an introduction to cluster analysis, vol. 39, Wiley Online Library, 1990.
[151] V. Vuori, J. Laaksonen, A comparison of techniques for automatic clustering of handwritten characters, Pattern Recognit., 2002 3 (2002) 30168.
[152] T.W. Liao, B. Bolt, J. Forester, E. Hailman, C. Hansen, R. Kaste, J. O'May, Understanding and projecting the battle state, in: Proceedings of 23rd Army Science Conference, Orlando, FL, 2002, pp. 2–3.
[153] T.W. Liao, C.F. Ting, An adaptive genetic clustering method for exploratory mining of feature vector and time series data,, Int. J. Prod. Res. 44 (14) (2006) 2731–2748.
[154] L. Gupta, D.L. Molfese, R. Tammana, P.G. Simos, Nonlinear alignment and averaging for estimating the evoked potential, IEEE Trans. Biomed. Eng. 43 (4) (1996) 348–356.
[155] E. Caiani, A. Porta, G. Baselli, M. Turiel, S. Muzzupappa, F. Pieruzzi, C. Crema, A. Malliani, S. Cerutti, Warped-average template technique

to track on a cycle-by-cycle basis the cardiac filling phases on left ventricular volume,, Comput. Cardiol. 1998 (1998) 73–76.
[156] T. Oates, L. Firoiu, P. Cohen, Using dynamic time warping to bootstrap HMM-based clustering of time series, Seq. Learn. Paradig. ALGORITHMS, Appl 1 (1828) (2001) 35–52.
[157] W. Abdulla, D. Chow, Cross-words reference template for DTW-based speech recognition systems, in: TENCON 2003. Conference on Convergent Technologies for Asia-Pacific Region, vol. 4, 2003, pp. 1576–1579.
[158] F. Petitjean, A. Ketterlin, P. Gançarski, A global averaging method for dynamic time warping, with applications to clustering, Pattern Recognit 44 (3) (2011) 678–693.
[159] L. Bergroth, H. Hakonen, A survey of longest common subsequence algorithms, in: Proceedings of the Seventh International Symposium on String Processing and Information Retrieval, 2000. SPIRE 2000, 2000, pp. 39–48.
[160] S. Aghabozorgi, T.Y. Wah, A. Amini, M.R. Saybani, A new approach to present prototypes in clustering of time series, in: Proceedings of the 7th International Conference of Data Mining, vol. 28, no. 4, 2011, pp. 214–220.
[161] S. Aghabozorgi, M.R. Saybani, T.Y. Wah, Incremental clustering of time-series by fuzzy clustering, J. Inf. Sci. Eng. 28 (4) (2012) 671–688.
[162] G. Karypis, E.H. Han, V. Kumar, Chameleon: hierarchical clustering using dynamic modeling, Comput. (Long. Beach. Calif). 32 (8) (1999) 68–75.
[163] S. Guha, R. Rastogi, K. Shim, CURE: an efficient clustering algorithm for large databases,, ACM SIGMOD Rec. 27 (2) (1998) 73–84.
[164] T. Zhang, R. Ramakrishnan, M. Livny, BIRCH: an efficient data clustering method for very large databases, ACM SIGMOD Rec. 25 (2) (1996) 103–114.
Q14 [165] M. Vlachos, J. Lin, E. Keogh, A wavelet-based anytime algorithm for k-means clustering of time series, Proc. Work. Clust (2003) 23–30.
[166] J.J. Van Wijk and E.R. Van Selow, Cluster and calendar based visualization of time series data, in: Proceedings of 1999 IEEE Symposium on Information Vision, , 1999, pp. 4–9.
[167] T. Oates, M.D. Schmill, P.R. Cohen, A method for clustering the experiences of a mobile robot that accords with human judgments, in: Proceedings of the National Conference on Artificial Intelligence, 2000, pp. 846–851.
[168] S. Hirano, S. Tsumoto, Empirical comparison of clustering methods for long time-series databases, Act. Min 3430 (2005) 268–286.
[169] J. MacQueen, Some methods for classification and analysis of multivariate observations, in: Proceedings of the fifth Berkeley symposium Mathematical Statist. Probability, vol. 1, 1967, pp. 281–297.
[170] R.T. Ng, J. Han, Efficient and effective clustering methods for spatial data mining, in: Proceedings of the International Conference on Very Large Data Bases, 1994, pp. 144–144.
[171] U. Fayyad, C. Reina, P.S. Bradley, Initialization of iterative refinement clustering algorithms, in: Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, 1998, pp. 194–198.
[172] P.S. Bradley, U. Fayyad, C. Reina, Scaling clustering algorithms to large databases, Knowl. Discov. Data Min (1998) 9–15.
[173] J. Beringer, E. Hullermeier, Online clustering of parallel data streams, Data Knowl. Eng. 58 (2) (2006) 180–204. Aug..
Q15 [174] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Kluwer Academic Publishers, 1981.
[175] J.C. Dunn, A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters, Cybern. Syst. 3 (3) (1973) 32–57.
[176] R. Krishnapuram, A. Joshi, O. Nasraoui, L. Yi, “Low-complexity fuzzy relational clustering algorithms for web mining,”, Fuzzy Syst. IEEE Trans vol. 9 (no. 4) (2001) 595–607.
[177] D. Dembélé, P. Kastner, Fuzzy C-means method for clustering microarray data, Bioinformatics 19 (8) (2003) 973–980.
[178] J. Alon, S. Sclaroff, Discovering clusters in motion time-series data in: Proceedings of Computer Society Conference on Computer Vision and Pattern Recognition, 2003, pp. 375–381.
[179] J.W. Shavlik, T.G. Dietterich, Readings in Machine Learning, Morgan Kaufmann, 1990.
[180] D.H. Fisher, Knowledge acquisition via incremental conceptual clustering, Mach. Learn. 2 (2) (1987) 139–172.
[181] G.A. Carpenter, S. Grossberg, A massively parallel architecture for a self-organizing neural pattern recognition machine, Comput. vision, Graph. image Process 37 (1) (1987) 54–115.
[182] T. Kohonen, The self-organizing map, Proc. IEEE 78 (9) (1990) 1464–1480.
[183] C. Biernacki, G. Celeux, G. Govaert, Assessing a mixture model for clustering with the integrated completed likelihood, IEEE Trans. Pattern Anal. Mach. Intell 22 (7) (2000) 719–725.
[184] M. Bicego, V. Murino, M. Figueiredo, Similarity-based clustering of sequences using hidden Markov models, Mach. Learn. data Min. pattern Recognit 2734 (1) (2003) 95–104.
[185] J. Hu, B. Ray, L. Han, An interweaved hmm/dtw approach to robust time series clustering, in: Proceedings of 18th International Conference on Pattern Recognition, 2006. ICPR 2006, , vol. 3, 2006, pp. 145–148.
[186] B. Andreopoulos, A. An, X. Wang, , A roadmap of clustering algorithms: finding a match for a biomedical application, Brief. Bioinform. 10 (3) (2009) 297–314.
[187] M. Ester, H.P. Kriegel, J. Sander, A density-based algorithm for discovering clusters in large spatial databases with noise, in: Proceedings of 2nd International, vol. 1996, no. 6, 1996, pp. 226–231. Q16
[188] M. Ankerst, M. Breunig, H. Kriegel, OPTICS: Ordering points to identify the clustering structure, ACM SIGMOD Rec 28 (2) (1999) 40–60.
[189] S. Chandrakala, C. Chandra, A density based method for multivariate time series clustering in kernel feature space, in: Proceedings of IEEE International Joint Conference on Neural Networks IEEE World Congress on Computational Intelligence, vol. 2008, 2008, pp. 1885–1890.
[190] W. Wang, J. Yang, R. Muntz, STING: a statistical information grid approach to spatial data mining, in: Proceedings of the International Conference on Very Large Data Bases, 1997, pp. 186–195.
[191] G. Sheikholeslami, S. Chatterjee, A. Zhang, Wavecluster: A multi-resolution clustering approach for very large spatial databases, in: proceedings of the International conference on Very Large Data Bases, 1998, pp. 428–439.
[192] Y. Kakizawa, R.H. Shumway, M. Taniguchi, Discrimination and clustering for multivariate time series, J. Am. Stat. Assoc 93 (441) (1998) 328–340.
[193] S. Policker, A.B.B. Geva, Nonstationary time series analysis by temporal clustering, Syst. Man, Cybern. Part B 30 (2) (2000) 339–343.
[194] J. Qian, M. Dolled-Filhart, J. Lin, H. Yu, M. Gerstein, Beyond synexpression relationships: local clustering of time-shifted and inverted gene expression profiles identifies new, biologically relevant interactions1, J. Mol. Biol. 314 (5) (2001) 1053–1066.
[195] Z.J. Wang, P. Willett, Joint segmentation and classification of time series using class-specific features, Syst. Man, Cybern. Part B Cybern. IEEE Trans 34 (2) (2004) 1056–1067.
[196] X. Wang, K.A. Smith, R.J. Hyndman, Dimension reduction for clustering time series using global characteristics, Comput. Sci. 2005 (2005) 792–795.
[197] S.M.M. Focardi, et al., Clustering economic and financial time series: Exploring the existence of stable correlation conditions, Citeseer (2005). Q17
[198] J. Abonyi, B. Feil, S. Nemeth, P. Arva, Principal component analysis based time series segmentation-application to hierarchical clustering for multivariate process data, in: Proceedings of IEEE International Conference on Computational Cybernetics, 2005, pp. 29–31.
[199] V.S. Tseng, C.P. Kao, Efficiently mining gene expression data via a novel parameterless clustering method,, IEEE/ACM Trans. Comput. Biol. Bioinforma 2 (4) (2005) 355–365.
[200] T.W. Liao, Mining of Vector Time Series by Clustering, 2005.
[201] D. Bao, A generalized model for financial time series representation and prediction,, Appl. Intell. 29 (1) (2007) 1–11.
[202] D. Bao, Z. Yang, Intelligent stock trading system by turning point confirming and probabilistic reasoning, Exp. Syst. Appl. 34 (1) (2008) 620–627.
[203] W. Liu, L. Shao, Research of SAX in distance measuring for financial time series data, in: Proceedings of the First International Conference on Information Science and Engineering, 2009, no. 70572070, pp. 935–937.
[204] T.C. Fu, F.L. Chung, R. Luk, C.M. Ng, Financial time series indexing based on low resolution clustering, in: Proceedings of the 4th IEEE International Conference on Data Mining (ICDM-2004), 2010, pp. 5–14.
[205] C.-P.P. Lai, P.-C.C. Chung, V.S. Tseng, A novel two-level clustering method for time series data analysis,, Expert Syst. Appl. 37 (9) (2010) 6319–6326.
[206] X. Zhang, J. Liu, Y. Du, T. Lv, A novel clustering method on time series data, Expert Syst. Appl. 38 (9) (2011) 11891–11900.
[207] J. Zakaria, A. Mueen, E. Keogh, Clustering time series using unsupervised-shapelets, in: Proceedings of 2012 IEEE 12th International Conference on Data Mining, 2012, pp. 785–794.

Please cite this article as: S. Aghabozorgi, et al., Time-series clustering – A decade review, Information Systems (2015), http://dx.doi.org/10.1016/j.is.2015.04.007

[208] R. Darkins, E.J. Cooke, Z. Ghahramani, P.D.W. Kirk, D.L. Wild, R.S. Savage, Accelerating Bayesian hierarchical clustering of time series data with a randomised algorithm, PLoS One 8 (4) (2013) e59795.

[209] O. Seref, Y.-J. Fan, W.A. Chaovalitwongse, Mathematical programming formulations and algorithms for discrete k-median clustering of time-series data, INFORMS J. Comput. 26 (1) (2014) 160–172.

[210] S. Ghassempour, F. Girosi, A. Maeder, Clustering multivariate time series using hidden Markov models, Int. J. Environ. Res. Public Health 11 (3) (2014) 2741–2763.

[211] S. Aghabozorgi, T. Ying Wah, T. Herawan, H.A. Jalab, M.A. Shaygan, A. Jalali, A hybrid algorithm for clustering of time series data based on affinity search technique, Sci.World J. 2014 (2014) 562194.

[212] A. Bellaachia, D. Portnoy, Y. Chen, A.G. Elkahloun, E-CAST: a data mining algorithm for gene expression data, in: Workshop on Data Mining in Bioinformatics, 2002, pp. 49–54.

[213] J. Lin, M. Vlachos, E. Keogh, D. Gunopulos, J. Liu, S. Yu, J. Le, A MPAA-based iterative clustering algorithm augmented by nearest neighbors search for time-series data streams, Adv. Knowl. Discov. Data Min (2005) 369–377.

[214] R.J. Hathaway, J.C. Bezdek, Visual cluster validity for prototype generator clustering models, Pattern Recognit. Lett 24 (9–10) (2003) 1563–1569.

[215] M. Halkidi, Y. Batistakis, M. Vazirgiannis, On clustering validation techniques, J. Intell. Inf 17 (2) (2001) 107–145.

[216] C.D. Manning, P. Raghavan, H. Schutze, Introduction to Information Retrieval, vol. 1, Cambridge University Press Cambridge, 2008 no. c..

[217] E. Amigó, J. Gonzalo, J. Artiles, F. Verdejo, A comparison of extrinsic clustering evaluation metrics based on formal constraints, Inf. Retr. Boston 12 (4) (2009) 461–486.

[218] M. Meila, Comparing clusterings by the variation of information,, in: B. Schölkopf, M. Warmuth (Eds.), Comparing Clusterings by the Variation of Information Learning Theory and Kernel Machines, Springer, 2003, pp. 173–187.

[219] A. Rosenberg, J. Hirschberg, V-measure: a conditional entropy-based external cluster evaluation measure, in: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), 2007, no. June, pp. 410–420.

[220] G. Gan, C. Ma, Data Clustering: Theory, Algorithms, and Applications. 2007.

[221] H. Kremer, P. Kranen, T. Jansen, T. Seidl, A. Bifet, G. Holmes, B. Pfahringer, An effective evaluation measure for clustering on evolving data streams, in: Proceedings of the 17th ACM SIGKDD international conference on Knowledge Discovery and Data Mining, 2011, pp. 868–876.

[222] Y. Zhao, G. Karypis, Empirical and theoretical comparisons of selected criterion functions for document clustering,, Mach. Learn. 55 (3) (2004) 311–331.

[223] Y. Xiong, D.Y. Yeung, Time series clustering with ARMA mixtures,, Pattern Recognit 37 (8) (2004) 1675–1689.

[224] E. Fowlkes, C.L. Mallows, A method for comparing two hierarchical clusterings,, J. Am. Stat. Assoc 78 (383) (1983) 553–569.

[225] J. Wu, H. Xiong, J. Chen, Adapting the right measures for k-means clustering, in: Proceedings of the 15th ACM SIGKDD, 2009, pp. 877–886.

[226] W.M. Rand, Objective criteria for the evaluation of clustering methods,, J. Am. Stat. Assoc 66 (336) (1971) 846–850.

[227] L. Hubert, P. Arabie, Comparing partitions,, J. Classif. 2 (1) (1985) 193–218.

[228] G.W. Milligan,, M.C. Cooper, A study of the comparability of external criteria for hierarchical cluster analysis,, A Study Comparitive External Criteria Hierarchical Cluster Anal vol. 21 (no. 4) (1986) 441–458.

[229] D. Steinley, Properties of the Hubert-Arable adjusted rand index,, Psychol. Methods 9 (3) (2004) 386.

[230] K.K. Yeung, D.D. Haynor, W. Ruzzo, Validating clustering for gene expression data,, Bioinformatics 17 (4) (2001) 309–318.

[231] K. Yeung, C. Fraley, A. Murua, A.E. Raftery, W.L. Ruzzo, Model-based clustering and data transformations for gene expression data, Bioinformatics 17 (10) (2001) 977–987.

[232] C.J. Van Rijsbergen, Information Retrieval, Butterworths, London Boston, 1979.

[233] S. Kameda, M. Yamamura, Spider algorithm for clustering time series,, World Scientific and Engineering Academy and Society (WSEAS) vol. 2006 (2006) 378–383.

[234] C.J. Van Rijsbergen, A non-classical logic for information retrieval, Comput. J. 29 (6) (1986) 481–485.

[235] B. Larsen, C. Aone, Fast and effective text mining using linear-time document clustering, in: Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1999, pp. 16–22.

[236] C. Studholme, D.L.G. Hill, D.J. Hawkes, An overlap invariant entropy measure of 3D medical image alignment,, Pattern Recognit 32 (1) (1999) 71–86.

[237] A. Strehl, J. Ghosh, Cluster ensembles—a knowledge reuse framework for combining multiple partitions, J. Mach. Learn. Res. 3 (2003) 583–617.

[238] X.Z. Fern, C.E. Brodley, Solving cluster ensemble problems by bipartite graph partitioning, in: Proceedings of the Twenty-first International Conference on Machine Learning, 2004, p. 36.

[239] F. Rohlf, Methods of comparing classifications, Annu. Rev. Ecol. Syst. (1974) 101–113.

[240] S. Lin, M. Song, L. Zhang, Comparison of cluster representations from partial second-to full fourth-order cross moments for data stream clustering, in: Proceedings of the Eighth IEEE International Conference on Data Mining, 2008. ICDM '08, 2008, pp. 560–569.

[241] J. Han, M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, San Francisco, CA, 2011.