Energy Proceedings
Vol 27, 2022
ISSN 2004-2965

Time Series Graph Feature-Based Method for Operation Pattern Recognition in Oil Pipeline System

Li Zhang[1], Huai Su[1*], Karine Zeitouni[2], Laurent Yeh[2], Lin Fan[1], Luxin Jiang[1], Jinjun Zhang[1*]

1 National Engineering Laboratory for Pipeline Safety/MOE Key Laboratory of Petroleum Engineering/Beijing Key Laboratory of Urban Oil and Gas Distribution Technology, China University of Petroleum-Beijing, 102249, China (Corresponding Author)

2 DAVID Lab, University of Versailles, Université Paris-Saclay, 78000, France

ABSTRACT

Time Series (TS) analysis is a hot topic in Data Mining community. Currently, the detection of the operation state in energy system often relies only on human judgment or even on-site inspection. But TS analysis can help automatizing this task and has become attractive in the energy field. In this paper, we propose a method for detecting and recognizing system operating pattern based on change points and complex network features. We first explain how the change point detection method can be applied in different pipeline operation scenarios. The results obtained by this method can help TS to segment subsequences and extract shapelet. Then, the shapelet are transformed into the form of visibility graph. The structural features of such TS graphs corresponding to different operating patterns of the system are extracted. Finally, we validate the graph feature-based representation method on datasets from an oil pipeline system in China. We compare it with statistical features-based representation baseline method for classification tasks. The results show the interpretive and accuracy of our proposed method. The method could be a basis for intelligent detection and recognition in the field of energy systems.

Keywords: time series, change point detection, visibility graph, structural features.

NONMENCLATURE

Abbreviations	
CS	Candidate shapelets
DTW	Dynamic Time Warping
DP	Distance profiles
MP	Matrix profiles
RS	Representative shapelet
SCADA	Supervisory Control and Data Acquisition
TSC	Time series classification
Symbols	
a	The average shortest path distance
A_{ij}	The adjacency matrix
C_p	The local clustering coefficient
$d(s,t)$	The shortest path from s to t
$E(G)$	The average global efficiency
G	The network graph
h	The threshold
K	The degrees of nodes
L	The length of the shapelets
m	The total number of edges
md	The length of T_g-*shapelet*
n	The number of nodes
n_i	The total number of neighboring nodes of node i
nd	The length of T'_g
ns	The total number of CS
nt	The length of T
N	The set of all nodes in the complex network graph
Q	Modularity
r	Assortativity coefficient
$t_1, t_2, \ldots, t_x, t_{nt}$	The different time series data
T	The time series
T_g	The Subsequences
v	The drift

1. INTRODUCTION

Analysis and discussion of time series data plays an increasingly important role in applications in science and society [1]. One of these is time series classification (TSC), which helps one to quickly and accurately identify anomalous patterns that may arise in practical applications. The interpretive of the classification results is a challenge in dealing with the classification task. Traditional TSC methods can be classified as distance-based methods, dictionary-based methods, feature-based methods and shapelets, each of which has its own merits in classification applications, but all suffer from weak interpretive. Therefore, it is worth discussing how these methods can be used or improved to obtain better interpretive classification results.

Distance-based methods. Standard distance measures are Euclidean Distance (ED) and Dynamic Time

Warping (DTW). Other popular time series distance measures include modified Hausdorff (MODH) [2], edit distance with real penalty (ERP) [3] and Longest Common Sub-Sequence (LCSS) [4]. DTW has the advantage of handling unequal length time series data [5], and one nearest neighbour (1-NN) using DTW distance is shown to be very hard to beat on many datasets [5-7]. However, these distance-based TSC methods are usually not interpretable.

Dictionary-based methods. Methods such as Symbolic Aggregate approXimation (SAX) [8], Piecewise Aggregate Approximation (PAA) [6] reduce the original sequence dimension by converting the sequence to some symbols [9]. Then, the similarity measure is built on comparing the encoding sequences of time sequences. The development of this method allows traditional data analysis techniques to be applied to the field of time series. However, how to determine the time window length and how to develop a better way for subsequence representation still requires continuous research and discussion.

Feature-based methods. Sequences are classified by extracting global features and constructing classifiers. Typical global features include statistical features such as mean, variance and slope, or time-frequency domain features (e.g peak and frequency) [10, 11]. However, one disadvantage of these features for practical problems is that the same statistical features may correspond to different time series. The better classification results shown by this method may rely heavily on strong classifiers (e.g. SVM, Adaboost, Random forest, etc.) rather than better global features [12]. Moreover, the method also lacks interpretability in a practical sense. Therefore, better ways of time series representation still need to be explored.

Shapelet-based methods. Finding shape information for describing sub-time series. This method proposed by Ye and Keogh [13] and are applied in TSC. However, as every shapelets candidate is compared to a set of TS, the computation complexity could be huge. Therefore, most studies focus on how to compensate for the shortage and reduce the number of shapelet candidates [14, 15]. However, it is undeniable that the method has been widely adopted due to its reliability and interpretive.

In this paper, we propose a time series classification method based on shapelet and graph structure features. Our work is a hybrid TSC approach which combines shapelet and feature based methods. However, prior to shapelet extraction, Change Point Detection (CPD) can detect state shifts or anomalous states in a time series. This last is also the basis for segmentation of the time series. To this end, we leverage the Cumulative Sum algorithm (CUSUM) [16]. Then, we exploit an existing shapelet extraction method [17]. It consists of three major steps: (1) calculating distance profiles, (2) calculating matrix profiles and (3) constructing representative shapelet sets. Afterwards, we introduce the visibility graph algorithm, which can convert shapelet into the form of graph (i.e. time-graphlet). It is intended to represent the shapelet by means of graph domain features. Meanwhile, the set of representative shapelet can also be filtered again based on the converted time-graphlets. The number of candidate shapelet is reduced by removing shapelet with the same graph structure. In particular, complex network graph structure features are employed to embed time-graphlet for representation. The method reduces the dimensional of the subsequence into a spatial feature vector. Finally, a classification model is used to classify the time series for recognition. we compare our proposed method with a feature-based methods and apply on real oil pipeline system in China. In summary, the main contributions of this work are as follows:

1. The CUSUM method based on change point detection is employed to assist in the segmentation process of time series and gives better performance of anomaly detection than Isolation Forest.
2. The feature representation method, based on a complex network graph structure, enhances the interpretive of shapelets for different categories by embedding the physical information propagated behind the data.
3. For operational data of oil pipelines, classification results based on graph structure features are more accurate than those based on statistical features.

The rest of the paper is organized as follows. In section 2, we briefly describe the CUSUM algorithm to identify the change points. In section 3 we show the details of constructing time-graphlets. The section 4 mainly describes the demonstration scenario and discussion. The final section summarizes the work in this paper and indicates future research directions.

2. DETECTION OF OPERATING STATE CHANGE POINTS

2.1 Change Point Detection

The cumulative sum algorithm is the classical technique used to detect change points in univariate time series [16]. The method is based on probability density ratios and has no restrictions on the distribution and smoothness of the time series data [18]. The method detects change points in a time series by calculating whether the cumulative sum of positive changes (g_t^+) and negative changes (g_t^-) in the time series data (t) exceeds a threshold (h). When (g_t^+) or (g_t^-) exceeds h, an alarm is raised and the cumulative sum is reset [19]. The detection process of the CUSUM algorithm is defined as

$$\begin{cases} s_x = t_x - t_{x-1} \\ g_x^+ = m\left(g_{x-1}^+ + s_x - v, 0\right) \\ g_x^- = m\left(g_{x-1}^+ - s_x - v, 0\right) \end{cases} \quad (1)$$

$$if\; g_x^+ > h\, or\, g_x^- > h : \begin{cases} x_{alarm} = x \\ g_x^+ = 0 \\ g_x^- = 0 \end{cases}$$

Where, h is the threshold value of the cumulative sum. When h increases, false alarms decrease. v is the drift parameter, which is set to avoid false alarms or slow drift. By increasing v, the false alarm rate can be reduced, but at the cost of delayed detection. Here, h and v are two parameters that determine the effectiveness of the algorithm. Their purpose is to weigh up the number of true and false alarms, and a suitable value can improve the precision of the detection.

2.2 Metrics

We use the time period corresponding to the real event as a comparison group to evaluate the detection results of the above two methods separately. We applied four metrics to evaluate the anomaly detection results: accuracy (*acc*), recall (*rec*), precision (*prec*) and *F1* value [1].

$$acc = (TP + TN)/(TP + FP + FN + TN) \quad (2)$$

$$prec = TP/(TP + FP) \quad (3)$$

$$rec = TP/(TP + FN) \quad (4)$$

$$F1 = 2(prec \times rec)/(prec + rec) \quad (5)$$

Where TP is the number of correctly predicted activities, TN is the number of correctly non-predicted activities, FP is the number of predicted activities that do not match the true labels, and FN is the number of activities presenting in training dataset yet absent in predictions.

3. RECOGNITION OF OPERATING PATTERNS

The process of operating pattern recognition can be divided into three parts: extraction of shapelet, construction of graphlets and feature characterization. The relationship between these three parts is illustrated in Fig. 1. Below, we describe in detail each one of the above steps.

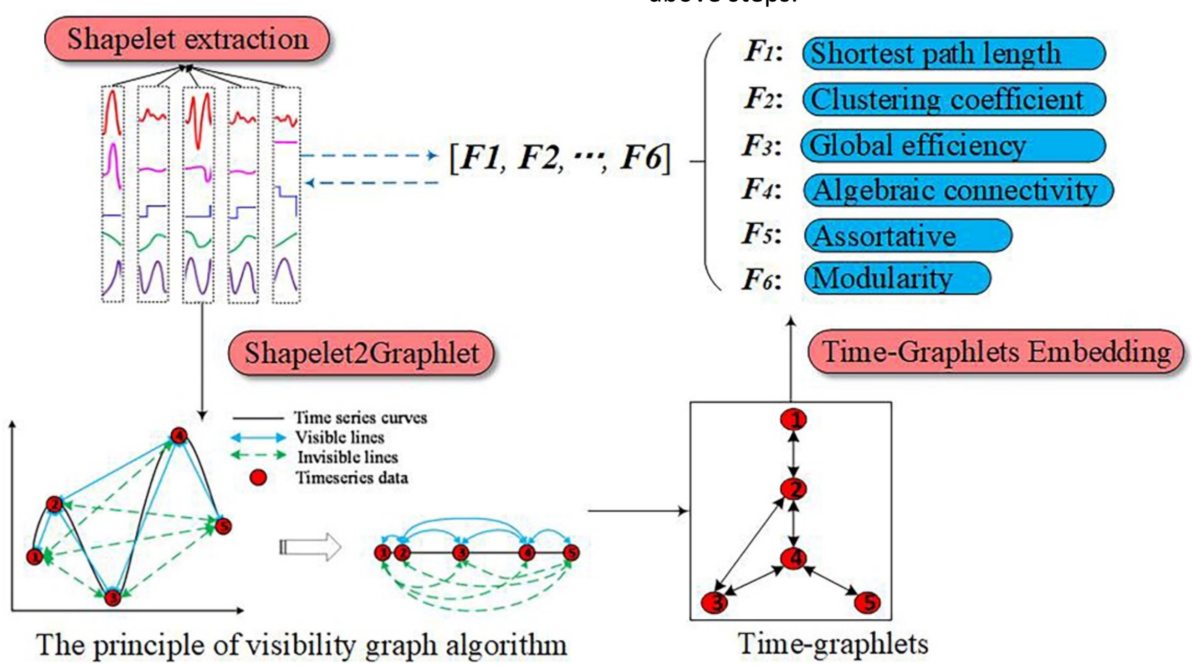

Fig. 1 The methodological framework for graph feature-based operation pattern recognition

3.1 Shapelet extraction

For a given time series T, after segmenting it using the change point detection method, we can extract the shapelet representing different categories from different subsequences. Below, we describe each part of the extraction process. The full process of shapelet extraction is shown in Fig. 2.

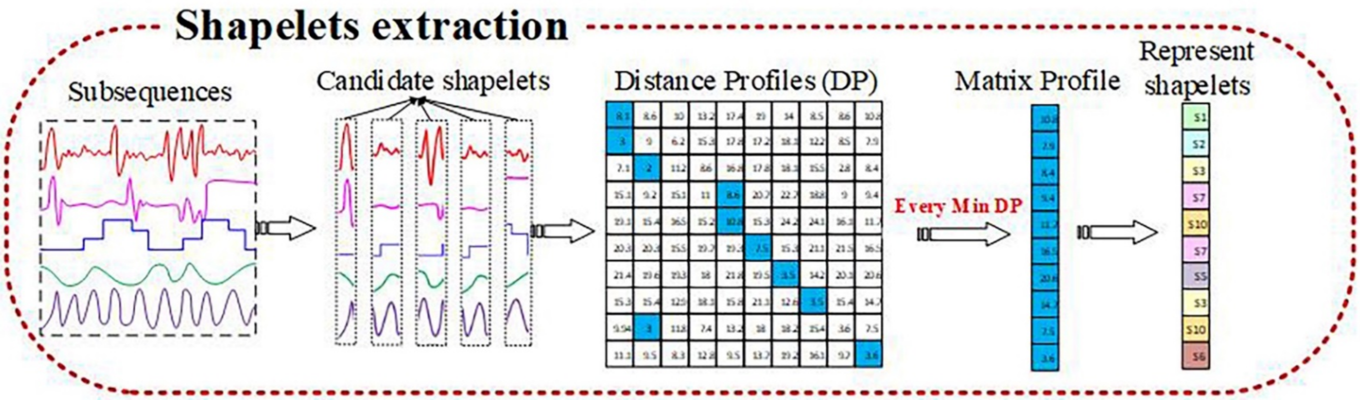

Fig. 2 The process of shapelet extraction

Definition 1. (Time series). The time series (T) is the historical operating data of the oil pipeline system. $T = \{t_1, t_2, t_3, \ldots, t_{nt}\}$, where nt is the length of T.

Definition 2. (Subsequence). Subsequences (T_g), $T_g = \{t_j, \ldots, t_k\}$, where $k < n$ and $j > 1$. The length of T_g is based on the results obtained by the change point detection method.

Definition 3. (Shapelet). Shapelet are defined as subsequences that can be used to describe the operational state of the system for a small period of time. The length of shapelet is the variable in the case of Section 4.2.

Definition 4. (Candidate shapelet). It represents all sub-sequences, which are selected according to the length of the shapelet. The time step is a regular duration (e.g. 1s) in order to avoid missing any run states. All subsequences in each target subsequence are extracted, thus forming the set of candidate shapelet (CS), $CS = \{CS_1, CS_2, CS_3, \ldots, CS_{ns}\}$, where ns is the total number of candidate shapelet.

Definition 5. (Distance profiles). Distance Profiles (DP) describe all distances between candidate shapelet and new shapelet in the unknown subsequence, which are measured using the DTW method: $DP = \{DP_1, DP_2, DP_3, \ldots, DP_{nd-md+1}\}$, nd is the length of T'_g, md is the length of $T_{g\text{-shapelet}}$.

For illustration, Fig. 3 takes a shapelet in a known subsequence and calculates the distance between each unknown $T'_{g\text{-shapelet}}$ and $T_{g\text{-shapelet}}$ in turn, and records all distances in the distance profile.

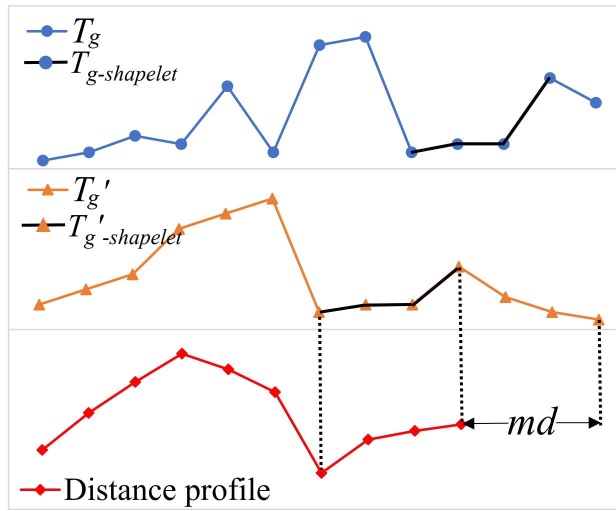

Fig. 3 Diagram of the distance profiles calculation

Definition 6. (Matrix profiles). Matrix profiles (MP) are vectors that record the shortest distance for each of the distance profiles, describing the $T_{g\text{-shapelet}}$ with the highest similarity to the $T'_{g\text{-shapelet}}$: $MP = \{MP_1, MP_2, MP_3, \ldots, MP_{nd-md+1}\}$.

Definition 7. (Representative shapelet). The $T_{g\text{-shapelet}}$ corresponding to each shortest distance in the matrix profiles is the representative shapelet that characterizes T'_g: $RS(T_g) = \{RS_1, RS_2, RS_3, \ldots, RS_{nd-md+1}\}$.

3.2 Shapelets to Graphlets

In this step, we first create a node for each time series data t_n. The edge exists between any two nodes (t_a and t_b) if another node (t_c) between them satisfies the Eq. (7), then, there is an edge between t_a and t_b. That is, if two arbitrary data can be connected by a straight line and there are no other data in between, then there is an edge between the nodes represented by t_a and t_b. Conversely, there is no edge between two nodes[20, 21].

$$y_c < y_b + (y_a - y_b)\left((t_b - t_c)/(t_b - t_a)\right) \tag{6}$$

The construction process of the time-graphlet is based on the visibility graph algorithm. The relevant processes are defined as follows.

Definition 8. (Time-graphlet). Time-graphlets are defined as time-series complex network graphs whose nodes are based on the data trends of shapelet.

Definition 9. (Shapelet2Graphlet). Shapelet2Graphlet is used to describe the process of transforming a shapelet into a graphlet with the aim of analyzing the operational characteristics of time-series data from the perspective of complex networks.

3.3 Time-Graphlets Embedding

At this point, we are ready to extract graph structure-related hidden information represented by multidimensional vectors. The idea is to embed each graph into the feature parameters of a complex network graph. The purpose of using complex network feature metrics to characterize the structural features of time-graphlet is to distil the implicit information of the shapelet from multiple perspectives, such as information transfer efficiency, connectivity and fault tolerance.

Meanwhile, we use statistical features based on time series as a baseline to compare the impact of the two sebsequence representation methods on the TSC results. The statistical features used include: mean, median, skewness, standard deviation values, variance, complexity, fluctuation and absolute energy, as many paper usually used [10, 11]. These basic features characterize the amplitude information of the subsequence.

Below, we describe in detail each of the feature parameters used for graph embedding.

Definition 10. (Average shortest path length). The average shortest path distance (a) can be used to measure the average of the shortest distance over all node pairs. It characterizes the efficiency of information transmission and the fluctuation of data on the network [22]:

$$a = \sum_{s,t \in N} d(s,t) / n(n-1) \tag{7}$$

where N is the set of all nodes in the complex network graph, $d(s, t)$ is the shortest path from s to t, and n is the number of nodes in the complex network graph.

Definition 11. (Assortativity coefficient). The assortativity coefficient can be used to characterize the connectivity in a time-graphlet based on the assortative or heterogeneous network of a given running state [23]:

$$r = \sum_{ij}(A_{ij} - K_i K_j / 2m) K_i K_j \Big/ \sum_{ij}(K_i \delta_{ij} - K_i K_j / 2m) K_i K_j \tag{8}$$

where m is the total number of edges, K_i and K_j represent the degrees of nodes i and j, respectively, and A_{ij} represents the adjacency matrix. If $r > 0$, the network is homogeneous. If $r < 0$, the network is heterogeneous. If $r = 0$, the two nodes are nonlinearly correlated with each other.

Definition 12. (Algebraic connectivity). The algebraic connectivity is employed to evaluate the connectivity and robustness of the complex network. The greater the algebraic connectivity, the less likely the time-graphlet is to be divided and the more stable the trend of the shapelet. It is expressed as the second eigenvalue of the Laplace matrix [24].

Definition 13. (Modularity). Modularity can be used as a further measure of the connectivity between similar nodes in the network [25]. If the modularity is significant, it is highly likely that the shapelet does not correspond to a particular operational state, and can be used to characterize the stability of the shapelet, as a side effect:

$$Q = \frac{1}{2m} \sum_{ij} (A_{ij} - k_i k_j / 2m) \delta(c_i, c_j) \tag{9}$$

where $\frac{1}{2}\sum_{ij} A_{ij}\delta(c_i, c_j)$ represents the sum of the number of edges between similar nodes, and $\frac{1}{2}\sum_{ij}(k_i k_j / 2m)\delta(c_i, c_j)$ represents the expected value of edges between similar nodes.

Definition 14. (Local clustering coefficient). The local clustering coefficient is used to measure the ability of a node to control the propagation of information among all its neighboring nodes [26]. The smaller the local clustering coefficient of a node, the greater the influence of the node:

$$C_p = 1 / K_p \left(K_p - 1\right) \sum_{q,r=1}^{N} A_{pq} A_{qr} A_{rp} \tag{10}$$

where A is the adjacency matrix. p, r, and q represent the starting point, process point, and endpoint of an edge, respectively, and K_p represents the degree of node p.

Definition 15. (Average global efficiency). The average global efficiency ($E(G)$) is expressed as Eq. (8) [27], which is used to describe whether the network is efficient at communicating.

$$E(G) = 1 / N(N-1) \sum_{i \neq j \in \mathbf{G}} \left(1 / d_{ij}\right) \tag{11}$$

where d_{ij} is the distance between nodes i and j, and N denotes the number of nodes in the network G.

4. CASE STUDY

We collected historical operational data and events from the Supervisory Control and Data Acquisition (SCADA) system of oil pipeline in China for nine stations in March-April 2020 and October-November 2020 (two months in total), with a time sampling of 1s. The types of parameters collected include inlet and outlet pressure, flow rate, density, temperature, etc. The historical events collected include the operation status, action and alarm information of the pipeline or equipment at each moment.

4.1 Operational State Detection

The valve states and pump shutdown are detected separately. The object of valve states detect is the difference between the inlet and outlet pressure of the outlet valve. Considering the delay of the sensor data during the states change, the maximum permissible time interval is set to 60s. Also, the time interval for adjacent change points was set to 3600s to ensure that only unique change points occur in each valve opening and closing operation, eliminating the influence of adjacent points.

The optimization results of the detection method for the two key parameters *h* and *v* are shown in Fig. 4. As we see, the changes of *h* have no effect on the accuracy and recall. This means that the detection results cover all true values. It is observed from the precision corresponding to the different parameters that the lowest false negative rate is found when *h* is 0.04 and *v* is 5×10^{-7}. Meanwhile, the *F*1 value exceeds 85.7%, which makes parameters combination the optimal parameter configuration corresponding to valve status.

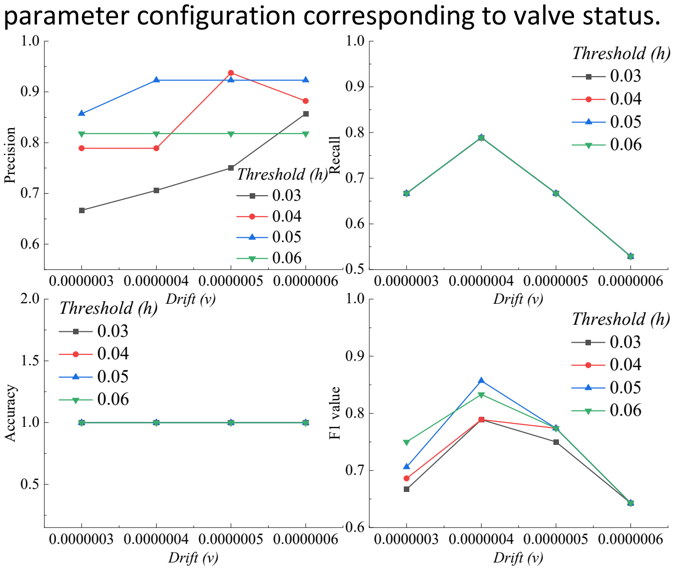

Fig. 4 Parameter optimization for valve states

The object of pump shutdown is the difference between the inlet and outlet pressure of the pump. During the experimental study, we found that the impact of v on the results was weak. Then we set the drift to 3×10^{-4}. The labels obtained by logical rules cover more time points and indicate a continuous state of operation. In contrast, change points are only for certain time points where the change is obvious and are more likely to detect sudden changes in state. Based on this, we need to pay attention to the effect of different time delays on the precision of the experiment.

The optimization results of the detection method for the two key parameters *h* and time delay are shown in Fig. 5. The detection accuracy increases with increasing time delay. When the time delay is 7h and the *h* of change point is ≥ 0.3, the precision for pump shutdown can reach more than 90%, and the *F*1 value is 0.689, which has a low false negative rate and false positive rate. It is the best parameter choice for this scenario.

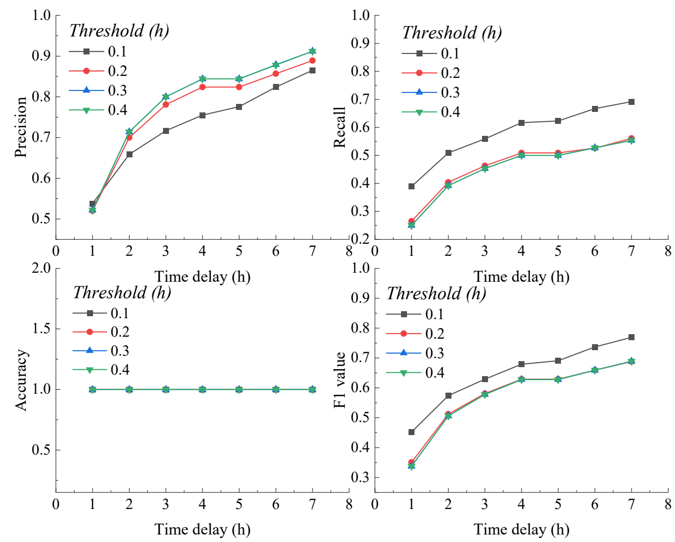

Fig. 5 Parameter optimization results for pump shutdown

The detection effectiveness of the two operational events was evaluated using the change point-based detection method and the isolated forest-based detection method respectively, and the results are shown in Table 1. The precision of both detection methods can reach over 90%, but the change point detection method has a higher recall than the isolated forest detection method. It means that the false negative rate obtained using is lower and the overall performance is better (as can be seen by the F1 value). On the other hand, when there is a lack of data labels, the results obtained from the change point detection method can be used as a basis for segmenting time series data. It provides more accurate annotation for large and complex systems lacking data annotation, or for early warning of different operational pattern.

Table 1 The evaluation results in difference scenarios

Operational event	Evaluation metrics	Methods comparison	
		Change point recognition	Isolation forest
Valve states	Precision	0.94	0.9
	Recall	0.79	0.53
	Accuracy	0.99	0.99
	F1	0.86	0.67

Pump shutdown	Precision	0.91	0.91
	Recall	0.55	0.383
	Accuracy	0.99	0.99
	F1	0.69	0.54

4.2 Recognition of pipeline operational state

The valve state of a sub-transmission station in an oil pipeline network is used as an example. Three classification models, Random Forest (RF), Adaptive Boosting (AdaBoost) and Support Vector Machine (SVM), are chosen to compare the classification effects of two representation methods based on statistical features and graph structure features, respectively. The results are shown in Table 2.

Table 2. The classification results of valve state

Subsequence length	Statistical features-based			Graph features-based		
	RF	AdaBoost	SVM	RF	AdaBoost	SVM
60	0.7734	0.8344	0.6488	0.9	0.989	0.949
120	0.7771	0.8411	0.6834	0.895	0.987	0.957
180	0.7811	0.8468	0.6738	0.897	0.988	0.958
240	0.7809	0.8293	0.6765	0.893	0.988	0.958
300	0.7810	0.8491	0.6970	0.884	0.988	0.968
360	0.7740	0.8565	0.6962	0.899	0.989	0.969

It can be seen that the classification accuracy based on graph feature is higher than that of the classification results based on statistical features. Among them, the classification results of the AdaBoost classifier are the most outstanding, reaching 98%. This is because there may be cases where the subsequence and the statistical feature have high similarity, but their corresponding operational events are not the same. In practical oil and gas pipeline engineering applications, some operating conditions do not bring about significant data changes. Therefore, the same statistical features do not necessarily mean that the operating events are also the same for the subsequences corresponding to different operating events. However, the graph structure features retain the shape characteristics of the original time series and also reflect the physical information propagated behind the data, so the classification obtained is better. On the other hand, the length of the input subsequence has a weaker effect on the classification effect.

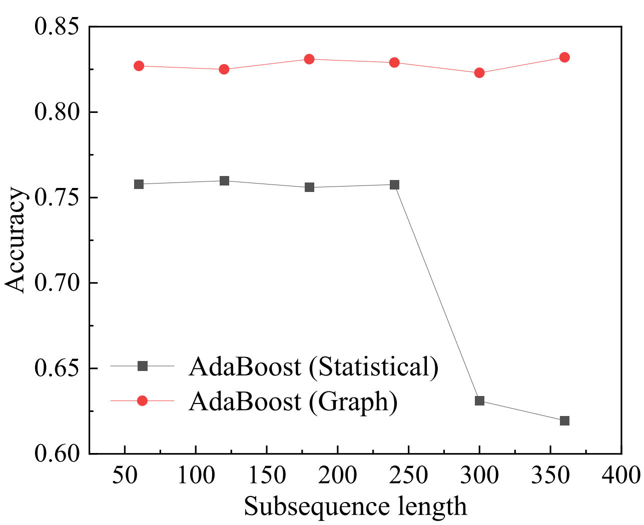

Fig. 6 Effect of different subsequence lengths on classification results

The best performing AdaBoost classifier was chosen to classify the composite-labels. The simultaneous occurrence of valve closed and valve internal leakage was used as an example. Fig. 6 shows the effect of different subsequence lengths on the classification results. As can be seen from the figure, the accuracy of classification based on statistical features decreases substantially when the sub-sequence length exceeds 240. In contrast, the impact of subsequence length on the classification accuracy based on graph structure features is weaker, and the accuracy is stable at over 80% in all cases. The results further demonstrate the value of combining shapelet with complex network theory for exploration.

5. CONCLUSION

In this paper, we propose a method for the representation of shapelets that improve the interpretive of TCS: the time-graphlet feature-based representation method. The approach is based on shapelet and graph feature embedding, both of which are interpretable. In particular, we have show that the efficient of taking CUSUM algorithm based on change point detection as a basis for the process of subsequence segmentation is better than isolation forest, especially for those time series datasets with incomplete labels. The visibility graph algorithm is used to convert shapelet into time-graphlets, which largely reduces the number of shapelet candidates. The features, which is used to evaluate the graph structure and information transfer capability of complex network, are adjusted to multi-dimensional vectors embedded in time-graphlets. This allows the feature vectors hold not only the shape characteristics of the original subsequence, but also the physical information conveyed behind the data sequence. The proposed method is applied to solve the problem of detecting and recognizing the operational state of industrial systems. We have measured that the classification results obtained by the graph features representation approach is more accurate than use the traditional statistics features.

In future work, we aim at carrying out online experiments to enrich the existing dataset and build multi-classification models to further improve the classification breadth and recognition accuracy of the models. Meanwhile, due to the complexity of large industrial systems, more feature dimensions describing the operating state will be considered, so that complex network features can be incorporated into the classification study of multivariate time series.

ACKNOWLEDGEMENT

Thanks to "Pipechina South Company" for the system and data support. This work is supported by the National Natural Science Foundation of China (51904316) and the research fund provided by the China University of Petroleum, Beijing (2462018YJRC038).

REFERENCE

[1] Liu L, Peng Y, Wang S, Liu M, Huang Z. Complex activity recognition using time series pattern dictionary learned from ubiquitous sensors. Information Sciences. 2016;340:41-57.

[2] Aghabozorgi S, Shirkhorshidi AS, Wah TY. Time-series clustering–a decade review. Information Systems. 2015;53:16-38.

[3] Chen L, Ng R. On the marriage of lp-norms and edit distance. Proceedings of the Thirtieth international conference on Very large data bases-Volume 302004. p. 792-803.

[4] Vlachos M, Kollios G, Gunopulos D. Discovering similar multidimensional trajectories. Proceedings 18th international conference on data engineering: IEEE; 2002. p. 673-84.

[5] Batista GE, Wang X, Keogh EJ. A complexity-invariant distance measure for time series. Proceedings of the 2011 SIAM international conference on data mining: SIAM; 2011. p. 699-710.

[6] Keogh E, Chakrabarti K, Pazzani M, Mehrotra S. Dimensionality reduction for fast similarity search in large time series databases. Knowledge and information Systems. 2001;3:263-86.

[7] Ding H, Trajcevski G, Scheuermann P, Wang X, Keogh E. Querying and mining of time series data: experimental comparison of representations and distance measures. Proceedings of the VLDB Endowment. 2008;1:1542-52.

[8] Lin J, Keogh E, Lonardi S, Chiu B. A symbolic representation of time series, with implications for streaming algorithms. Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery2003. p. 2-11.

[9] Bagnall A, Bostrom A, Large J, Lines J. The great time series classification bake off: An experimental evaluation of recently proposed algorithms. Extended version. arXiv 2016. arXiv preprint arXiv:160201711. 2016.

[10] Baydogan MG, Runger G, Tuv E. A bag-of-features framework to classify time series. IEEE transactions on pattern analysis and machine intelligence. 2013;35:2796-802.

[11] Deng H, Runger G, Tuv E, Vladimir M. A time series forest for classification and feature extraction. Information Sciences. 2013;239:142-53.

[12] Zhao J, Itti L. Classifying time series using local descriptors with hybrid sampling. IEEE Transactions on Knowledge and Data Engineering. 2015;28:623-37.

[13] Ye L, Keogh E. Time series shapelets: a new primitive for data mining. Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining2009. p. 947-56.

[14] Malinowski S, Chebel-Morello B, Zerhouni N. Remaining useful life estimation based on discriminating shapelet extraction. Reliability Engineering & System Safety. 2015;142:279-88.

[15] Li G, Yan W, Wu Z. Discovering shapelets with key points in time series classification. Expert systems with applications. 2019;132:76-86.

[16] Page ES. Continuous inspection schemes. Biometrika. 1954;41:100-15.

[17] Zuo J, Zeitouni K, Taher Y. Incremental and Adaptive Feature Exploration over Time Series Stream. 2019 IEEE International Conference on Big Data (Big Data): IEEE; 2019. p. 593-602.

[18] Aminikhanghahi S, Cook DJ. A survey of methods for time series change point detection. Knowledge and information systems. 2017;51:339-67.

[19] El Hafyani H, Zeitouni K, Taher Y, Abboud M. Leveraging change point detection for activity transition mining in the context of environmental crowdsensing. Actes de la conférence BDA2020. p. 64.

[20] Lacasa L, Luque B, Ballesteros F, Luque J, Nuno JC. From time series to complex networks: The visibility graph. Proceedings of the National Academy of Sciences. 2008;105:4972-5.

[21] Bezsudnov I, Snarskii A. From the time series to the complex networks: The parametric natural visibility graph. Physica A: Statistical Mechanics and its Applications. 2014;414:53-60.

[22] Wong A, Tan S, Chandramouleeswaran KR, Tran HT. Data-driven analysis of resilience in airline networks. Transportation Research Part E: Logistics and Transportation Review. 2020;143:102068.

[23] Newman ME. Mixing patterns in networks. Physical review E. 2003;67:026126.
[24] Wei P, Spiers G, Sun D. Algebraic connectivity maximization for air transportation networks. IEEE Transactions on Intelligent Transportation Systems. 2013;15:685-98.
[25] Clauset A, Newman ME, Moore C. Finding community structure in very large networks. Physical review E. 2004;70:066111.
[26] Zou Y, Donner RV, Marwan N, Donges JF, Kurths J. Complex network approaches to nonlinear time series analysis. Physics Reports. 2019;787:1-97.
[27] Latora V, Marchiori M. Efficient behavior of small-world networks. Physical review letters. 2001;87:198701.