JOURNAL OF SCIENCE
OF LAC HONG UNIVERSITY
ISSN: 2525 - 2186
Tạp chí Khoa học Lạc Hồng, **2025**, 22, 040-046

NGHIÊN CỨU VÀ PHÁT TRIỂN PHƯƠNG PHÁP LỰA CHỌN THUẬT TOÁN PHÂN ĐOẠN CHO BÀI TOÁN PHÁT HIỆN BẤT THƯỜNG TRONG DỮ LIỆU CHUỖI THỜI GIAN

Nguyễn Hòa Nhật Quang[1*], Võ Khương Lĩnh[2], Khâu Văn Bích[1]
[1]*Trường Đại học Trần Đại Nghĩa, Thành Phố Hồ Chí Minh, Việt Nam*
[2]*Trường Đại học Nguyễn Huệ, tỉnh Đồng Nai, Việt Nam*
* Tác giả liên hệ: *nhatquanghvkt@gmail.com*

THÔNG TIN BÀI BÁO

Ngày nhận: 16/4/2025
Ngày hoàn thiện: 31/5/2025
Ngày chấp nhận: 07/6/2025
Ngày đăng: 15/9/2025

TỪ KHÓA

Phương pháp phân đoạn;
Phát hiện bất thường;
Chuỗi thời gian.

TÓM TẮT

Nghiên cứu và cải tiến phương pháp phân đoạn ứng dụng cho bài toán phát hiện bất thường trong dữ liệu chuỗi thời gian nhằm ứng dụng vào bài toán phát hiện bất thường trong dữ liệu chuỗi thời gian. Bài báo sẽ trình bày về quá trình thu thập và xây dựng các bộ dữ liệu cần thiết, bao gồm bộ dữ liệu chuỗi thời gian không theo chu kỳ và có chu kỳ, cùng với mô tả chi tiết về bộ dữ liệu SWAT 2019. Tiếp theo, bài báo sẽ đi sâu vào quá trình phân đoạn, bao gồm việc trích xuất các phân đoạn bất thường và đánh giá kết quả thực nghiệm. Cuối cùng, bài báo sẽ thảo luận về việc lựa chọn sai số tối đa (max_error), một yếu tố quan trọng trong việc tối ưu hóa quá trình phân đoạn và phát hiện bất thường.

RESEARCH AND DEVELOPMENT OF A METHOD TO SELECT THE SEGMENTATION ALGORITHM FOR THE PROBLEM OF DETECTING IRREGULARITIES IN TIME SERIES DATA

Nguyen Hoa Nhat Quang[1*], Vo Khuong Linh[2] , Khau Van Bich[1]
[1]*Tran Dai Nghia University, Ho Chi Minh City, Vietnam*
[2]*Nguyen Hue University, Dong Nai Province, Vietnam*
*Corresponding Author: *nhatquanghvkt@gmail.com*

ARTICLE INFO

Received: Apr 16th, 2025
Revised: May 31st, 2025
Accepted: Jun 7th, 2025
Published: Sep 15th, 2025

KEYWORDS

Segmentation methods;
Anomaly detection;
Time series.

ABSTRACT

Research and Enhancement of Segmentation Methods for Anomaly Detection in Time Series Data. This paper presents a study and improvement of segmentation techniques applied to the problem of anomaly detection in time series data. The work outlines the process of collecting and constructing essential datasets, including both periodic and non-periodic time series, along with a detailed description of the SWAT 2019 dataset. Subsequently, the paper delves into the segmentation process, focusing on the extraction of anomalous segments and the evaluation of experimental results. Finally, the study discusses the selection of the maximum allowable error (max_error), a critical parameter for optimizing the segmentation process and improving the performance of anomaly detection.

Doi: https://doi.org/10.61591/jslhu.22.718
Available online at: https://js.lhu.edu.vn/index.php/lachong

1. INTRODUCTION

Currently, the development of variable-length segmentation algorithms remains limited. Historically, most research has primarily focused on fixed-length segmentation algorithms, where segments are divided into equal-length intervals. This approach introduces several issues and inherent limitations when applied to real-world scenarios, especially in the context of anomaly detection [10]. Fixed-length segmentation algorithms are often constrained by their rigid structure, which may result in poor adaptability to dynamic and complex data patterns. Therefore, variable-length segmentation algorithms offer significant advantages, particularly in anomaly detection tasks, by allowing more flexible and adaptive segmentation based on the intrinsic characteristics of the time series data. [1, 2, 6]:

High Flexibility: Variable-length segments can be dynamically adjusted based on the characteristics of the data, allowing for more accurate representation of changes or critical events within the time series.

Improved Detection Capability: Variable-length segmentation enables the identification of subtle anomalies that fixed-length segments may overlook, thereby increasing the accuracy and reliability of anomaly detection systems.

Broad Applicability: This approach is suitable for a wide range of data types, from real-time streaming data to unstructured data, and across various domains including industrial systems, healthcare, and finance.

Despite these notable advantages, most current research has yet to focus deeply on the development of variable-length segmentation algorithms, particularly for anomaly detection tasks [5]:

Lack of In-depth Research: Existing studies still predominantly concentrate on fixed-length segmentation algorithms, resulting in a shortage of innovative models and methods tailored to more complex problem domains.

Limited Real-world Adoption: Variable-length segmentation models have not been widely implemented in anomaly detection systems, which compromises the effectiveness and precision of such systems.

High Computational Cost: Traditional approaches, such as exhaustive search techniques, are often used to determine optimal max-error values and to identify suitable segmentation strategies for each specific type of time series. These methods are time-consuming and computationally expensive.

2. RELATED WORK

2.1 Overview of the Method

Survey of Existing Segmentation Methods: A comprehensive evaluation of current segmentation algorithms is conducted based on key criteria such as accuracy, computational complexity, and adaptability to highly volatile data. Each method's strengths and weaknesses are analyzed to better understand their scope of application and inherent limitations.

Application to Time Series Anomaly Detection: The surveyed segmentation methods are then applied to the problem of anomaly detection in time series data [9]. The performance of each method is assessed in terms of its effectiveness in identifying anomalies and detecting significant changes within the time series.

Proposing a Method for Selecting Appropriate Segmentation Algorithms: A methodology is developed for selecting the most suitable segmentation algorithm tailored to specific types of time series data. This includes proposing a set of evaluation criteria and assessment methods to assist users in identifying the most appropriate algorithm for their anomaly detection tasks.

Addressing the max_error Parameter Selection Problem: The max_error value is a critical parameter in time series segmentation algorithms, enabling users to control the trade-off between segmentation accuracy and computational complexity. Proper configuration of the max_error helps to optimize the segmentation process by balancing approximation precision with processing efficiency. Therefore, this paper investigates and proposes an optimized method for the fast and accurate selection of the max_error value. Multiple max_error configurations are experimented with to determine the optimal setting for each specific type of time series data.

Evaluation and Real-world Experiments: Extensive real-world experiments are conducted on diverse time series datasets to validate the effectiveness of the proposed algorithms and methods. Experimental results are analyzed to draw meaningful conclusions and provide recommendations for practical applications.

2.2 Sliding Windows Segmentation Algorithm

The Sliding Window Algorithm, also known as the brute-force or one-pass approach [4, 7], is one of the most commonly used methods for time series segmentation. This algorithm begins by selecting the first data point as an anchor point. An initial window size is then defined, and based on this size, the approximation error for the potential segment is computed. Next, the window size is incrementally increased until the approximation error exceeds a predefined threshold. At that point, a segment is created using the largest possible window size that still satisfies the error constraint. This process is repeated until the sliding windows have covered the entire time series. The anchor point is updated to the data point immediately following the previously formed segment to continue the procedure [8].

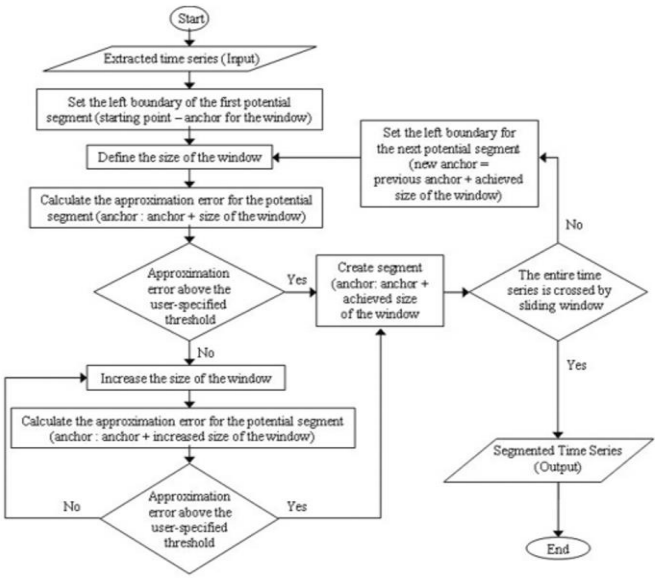

Figure 1. *Sliding Windows Algorithm Flowchart.*

2.3 Top-Down Segmentation Algorithm

The Top-Down Segmentation Algorithm begins by treating the entire time series as a single initial segment. The algorithm then searches for a split point that divides the series into two subsequences such that the difference between the two resulting segments is maximized. Next, the approximation error is computed for both subsegments and compared against a predefined error threshold. This splitting process is recursively repeated on each segment until the approximation error for all segments falls within the acceptable threshold [8].

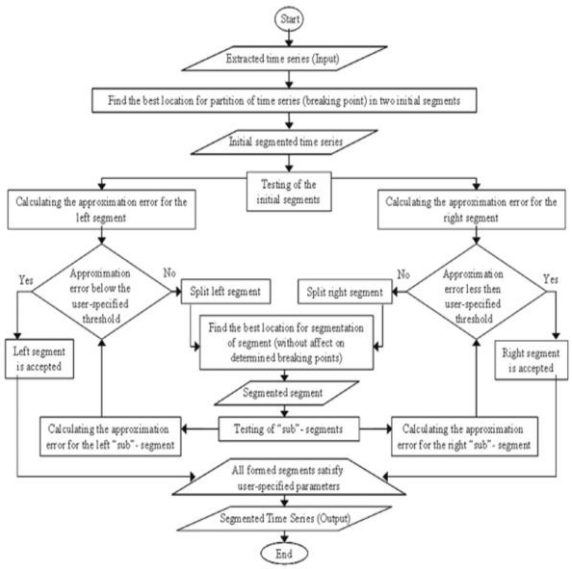

Figure 2. *Top-Down Algorithm Flowchart.*

2.4 Bottom-Up Segmentation Algorithm

The Bottom-Up Segmentation Algorithm operates in the reverse manner of the previously described Top-Down approach. It begins by dividing the original time series of length n into $n-1$ individual segments. The algorithm then iteratively decides whether to merge a segment with its left or right neighbor, based on the increase in approximation error that would result from the merge. This merging process continues until the approximation error of a candidate segment exceeds a predefined threshold [8].

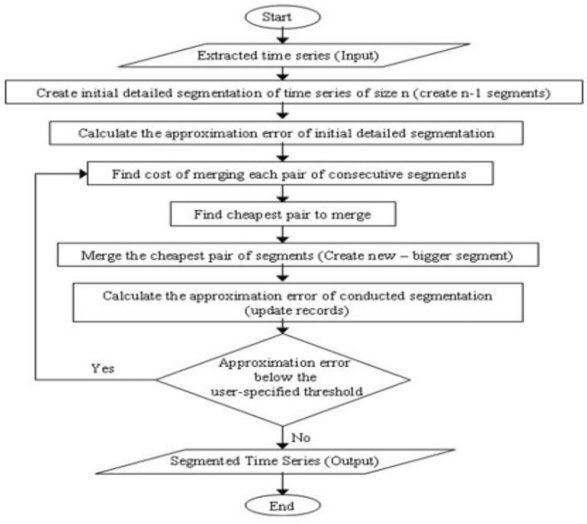

Figure 3. *Bottom – Up Algorithm Flowchart.*

2.5 SWAB Segmentation Algorithm

The SWAB Algorithm (Sliding Window And Bottom–Up) is a hybrid segmentation method that combines the Sliding Window approach with the Bottom-Up algorithm. This method is designed to handle streaming data by incrementally incorporating new data points into a buffer as they arrive, making it suitable for potentially unbounded data streams. However, properly defining the initial buffer size is crucial for the algorithm's effectiveness [8].

In SWAB, the buffer size is fixed and chosen to be large enough to generate 5 to 6 initial segments. If the buffer is too large, the resulting segmentation will closely resemble that of the Bottom-Up algorithm. Conversely, if the buffer is too small, the segmentation result will be similar to that produced by the Sliding Window method.

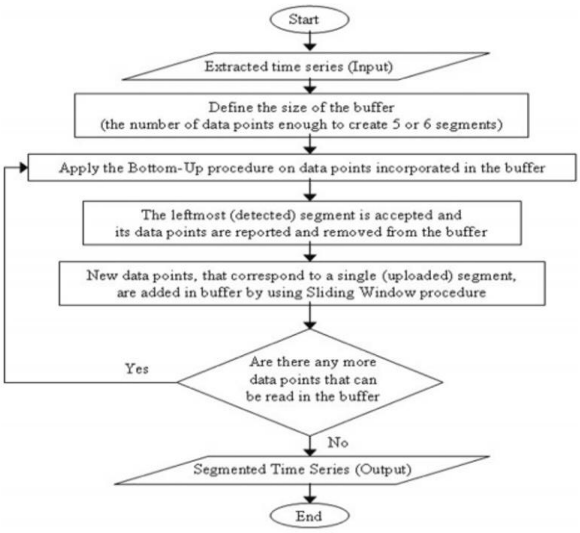

Figure 4. *SWAB Algorithm Flowchart.*

3. SURVEY AND EVALUATION OF THE EFFECTIVENESS OF SEGMENTATION ALGORITHMS

To verify the feasibility and effectiveness of the proposed method for selecting appropriate segmentation algorithms tailored to different types of time series data, the paper conducts a series of experiments focusing on variable-length segmentation algorithms. During these experiments, the study applies Sliding Window, Bottom-Up, Top-Down, and SWAB (Sliding Window and Bottom-Up) segmentation techniques across multiple datasets. This enables the paper to draw conclusions regarding which segmentation algorithm is best suited for each specific type of time series.

Each algorithm produces multiple segments. In order to evaluate the capability of each algorithm to detect anomalies and significant changes within the time series when applied following the proposed method performance is measured using the F1-Score, a common metric in classification tasks that balances precision and recall.

The first step is to identify which time points are considered anomalous. The study begins by applying the Bottom-Up algorithm to a segmented time series extracted

from a dataset that will be introduced in the following section of the paper.

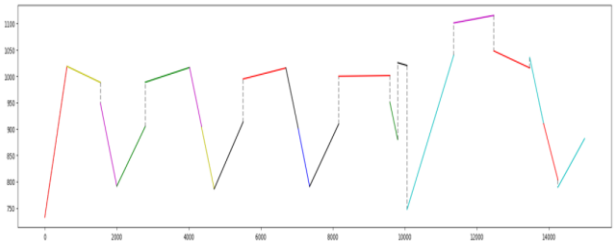

Figure 5. *Applying the Bottom-Up Algorithm to a Pre-Segmented Time Series*

We use a simple formula to determine whether a segment is anomalous by assigning a segment s_i defined by two endpoints (x_{i1}, y_{i1}) và (x_{i2}, y_{i2}). We then compute the distance (d) from this segment to both the preceding and succeeding segments using the following formula:

$$d_{S_i} = |(y_{i_1} - y_{i-1_2}) + (y_{i_2} - y_{i+1_1})|$$

This formula is used to calculate the deviation between the preceding and succeeding points of a segment. Based on these deviations, the segment with the largest deviation is selected as a candidate for anomaly detection.

By selecting the optimal max_error value for each dataset (the selection process for max_error will be described in a subsequent section), and identifying the segment that best describes the anomaly, the paper evaluates segmentation performance using classification metrics. Specifically, performance is measured using the F1-Score, a harmonic mean that combines both precision and recall. The concepts of precision and recall are defined as follows [3]:

Precision measures the proportion of segments detected as anomalous that are actually true anomalies. In other words, it is the ratio between the number of correctly detected anomalous segments (True Positives) and the total number of segments identified as anomalies (i.e., True Positives + False Positives). The formula is:

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}}$$

Where:

True Positives (TP): The number of truly anomalous segments that were correctly identified.

False Positives (FP): The number of segments that were incorrectly identified as anomalies (i.e., they are normal segments labeled as anomalous).

Recall measures the proportion of actual anomalous segments that were successfully detected. In other words, it is the ratio between the number of correctly identified anomalous segments (True Positives) and the total number of actual anomalies (i.e., True Positives + False Negatives). The formula is:

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}}$$

Where:

True Positives (TP): The number of truly anomalous segments that were correctly detected.

False Negatives (FN): The number of truly anomalous segments that were not detected by the algorithm.

F1-Score is a harmonic mean of Precision and Recall, providing a balanced metric that reflects the segmentation model's performance in terms of both correctness and completeness. It is especially useful when there is an uneven class distribution or when both false positives and false negatives are costly. The formula is:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Thus, precision indicates the proportion of segments identified by the algorithm as anomalous that are actually true anomalies, while recall reflects the proportion of actual anomalous segments that were successfully detected by the algorithm. For segments with an F1-Score greater than 0.7, we consider the corresponding anomalous sequence to be accurately detected [10].

4. EXPERIMENTAL RESULTS

4.1 Dataset

The dataset used in this study is publicly available at the following link:

https://drive.google.com/drive/folders/1OD2DSt4T-3WysolSLCda7hNYnoc62Mtq.

This dataset was collected from a testbed of the Secure Water Treatment (SWaT) system a scaled-down but fully operational water treatment plant prototype composed of six distinct operational stages. The data collection spanned 11 consecutive days, with SWaT operating 24 hours a day. During the first 7 days, the system functioned under normal conditions without any cyberattacks. In the remaining 4 days, multiple attack scenarios were introduced. Sensor and actuator data were logged every second to a central Historian server. In total, 946,722 data points were collected, covering 51 attributes. The system took approximately 5 hours to stabilize from an initial empty state. Data recording started with the filling of empty tanks, each of which required a different amount of time to reach steady-state operations depending on its stage in the treatment process. Network traffic data was collected using a commercial monitoring device from Check Point Software Technologies R2, which was deployed within the SWaT testbed to capture comprehensive network flows for security analysis. However, only network traffic relevant to intrusion detection was retained. This monitoring began as soon as the testbed entered operational mode. The attacks were executed on Level 1 of the SWaT network, which records communication between the SCADA system and the Programmable Logic Controllers (PLCs). These attacks involved packet hijacking, where communication between SCADA and PLCs was intercepted and modified to inject spoofed sensor values into the system.

Labeling in this dataset was facilitated by a well-controlled experimental setup.

During the execution of the testbed, all system operations and attack events were meticulously logged. As a result, every attack launched during the data collection phase was recorded and documented, with detailed information presented in Table 1. For labeling of the physical process variables, each data stream originating from a sensor or actuator was individually collected and stored in separate CSV files, making it easier to manage and apply accurate labels for anomaly detection tasks.

Table 1. *Attack Logs*

Information	Description
Start time	Time when attack starts
End time	Time when attack ends
Attack Points	Sensors or actuator which will be compromised
Start State	Current status of the point
Attack	Description of attack
Attack Value	Substituted value of sensor (based on the attack)
Attacker's Intent	The intended affect of the attack

Below are some of the fields from the SWaT dataset, presented as time series:

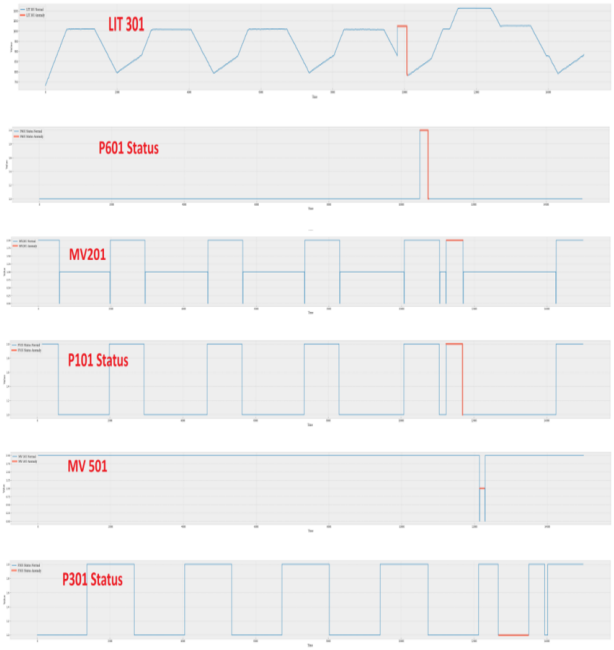

Figure 6. *The fields P601 Status, MV501, LIT301, MV201, P101 Status, and P301 Status from the SWaT dataset are represented as time series.*

Aperiodic time series: P601 Status, MV501

Periodic time series: LIT301, MV201, P101 Status, P301 Status

In addition to the SWaT dataset, the study also conducts experiments on synthetically generated datasets, which include both aperiodic and periodic time series. These datasets are designed to further evaluate the performance and generalizability of the proposed segmentation and anomaly detection methods under varying data characteristics.

4.2 Experimental Results

The authors applied segmentation algorithms including Sliding Window, Bottom-Up, Top-Down, and SWAB on the previously introduced SWaT dataset, in combination with synthetically generated datasets. The results obtained from these experiments are presented in the following tables:

Table 2. *Performance on Aperiodic Data (✗ indicates that the segmentation algorithm did not detect all anomaly points, ✓ indicates that the segmentation algorithm successfully detected all anomaly points, N stands for No, and Cyclic refers to periodic time series).*

Id	Topdown	Bottom-up	SlidingWindow	SWAB	Cyclic
1	✓	✓	✓	✓	N
2	✓	✓	✓	✓	N
3	✓	✓	✓	✓	N
4	✓	✗	✓	✗	N
5	✓	✓	✓	✓	N
MV501	✓	✓	✓	✓	N
P301Status	✓	✗	✗	✗	N

Table 3. *Performance on Low RSD Data (< 0,1) (✗ indicates that the segmentation algorithm did not detect all anomaly pointsm, ✓ indicates that the segmentation algorithm successfully detected all anomaly points, N stands for No, Y stands for Yes, Cyclic refers to periodic time series, and RSD stands for Relative Standard Deviation).*

Id	Topdown	Bottom-up	SlidingWindow	SWAB	Cyclic	RSD
3	✓	✓	✓	✓	N	0.049803
4	✓	✗	✓	✗	N	0.017071
7	✓	✓	✓	✓	Y	0.035958
8	✓	✓	✓	✓	Y	0.004309
9	✓	✓	✓	✓	Y	0.004903
10	✗	✗	✓	✓	Y	0.004551
LIT 301	✓	✓	✓	✓	Y	0.097361
MV 501	✓	✓	✓	✓	N	0.057789

Table 4. *Performance on Periodic Datasets (✗ indicates that the segmentation algorithm did not detect all anomaly points, ✓ indicates that the segmentation algorithm successfully detected all anomaly points, Y stands for Yes, and Cyclic refers to periodic time series).*

Id	Topdown	Bottom-up	SlidingWindow	SWAB	Cyclic
6	✓	✗	✓	✗	Y
7	✓	✓	✓	✓	Y
8	✓	✓	✓	✓	Y
9	✓	✓	✓	✓	Y
10	✗	✗	✓	✓	Y
LIT 301	✓	✓	✓	✓	Y
MV 201	✗	✓	✗	✓	Y
P101Status	✗	✗	✗	✓	Y
P601Status	✓	✓	✓	✓	Y

Table 5. *Performance on High RSD Data (RSD > 0.1) (✗ indicates that the segmentation algorithm did not detect all anomaly points, ✓ indicates that the segmentation algorithm successfully detected all anomaly points, Y là Yes, N for No, Cyclic indicates whether the time series is periodic, and RSD refers to Relative Standard Deviation).*

Id	Topdown	Bottom-up	SlidingWindow	SWAB	Cyclic	RSD
1	✓	✓	✓	✓	N	0.142541
2	✓	✓	✓	✓	N	0.201003
6	✓	✗	✓	✗	Y	0.685629
5	✓	✓	✓	✓	N	inf
MV 201	✗	✓	✗	✓	Y	0.121035
P101Status	✗	✗	✗	✓	Y	0.338061
P301Status	✓	✗	✗	✗	N	0.352146
P601Status	✓	✓	✓	✓	Y	0.361290

Overall, for aperiodic data, the Top-Down method emerges as the best choice. For both periodic and aperiodic datasets, the SWAB and Sliding Window methods demonstrate higher effectiveness. However, the Sliding Window method still performs worse than Top-Down and SWAB in certain scenarios.

Next, we will examine the time complexity and runtime performance of each segmentation method:

Table 6. *Average Runtime Across All 16 Datasets and the Dataset with the Highest Latency (Id = 5).*

	Top – down	Bottom - up	Sliding window	SWAB
Average time (s)	77.045958	12.259271	5.436719	31.828437
ID = 5(s)	454.333333	78.800000	22.700000	74.313333

Based on the survey across 16 datasets, it is observed that the Top-Down algorithm is 15 to 20 times slower than the Sliding Window algorithm. Compared to the Bottom-Up and SWAB methods, Top-Down is also approximately 6 times slower.

4.3 Selection of the Maximum Allowable Error (max_error)

All experimental results primarily depend on the selection of the maximum allowable error (max_error). Therefore, to determine the optimal max_error value quickly and accurately, this paper calculates the correlation between max_error and various time series characteristics such as relative standard deviation, number of data points, mean value, median, variance, kurtosis, among others.

The comprehensive analysis results are compiled into a correlation table that illustrates the influence level of each feature on the max_error value across different segmentation algorithms. This table highlights which time series characteristics most significantly affect the selection of max_error for optimal segmentation performance.

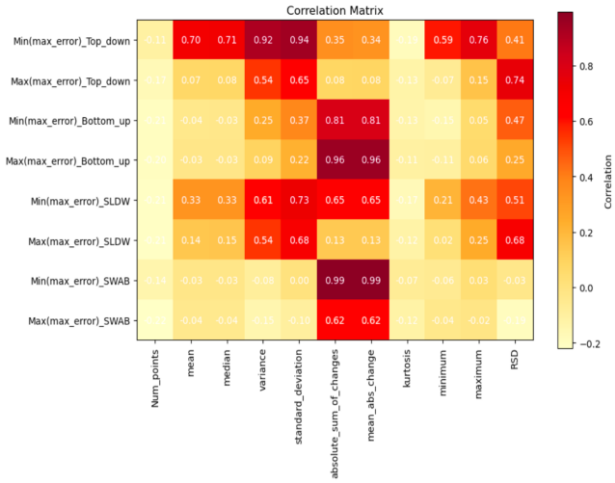

Figure 7. *Correlation Between Max-Error and Certain Time Series Characteristics*

Where: min(max_error) and max(max_error) represent the range of maximum allowable errors within which anomalies can actually be segmented. This range generally corresponds to cases where the F1-Score exceeds 0.7. In situations where algorithms fail to detect anomalies correctly (i.e., incorrect segmentation), we resort to the most intuitive visualization of the anomalous segment.

Based on Figure 7, it can be observed that the Sliding Window algorithm, and especially the Top-Down algorithm, are most strongly correlated with standard deviation and variance. On the other hand, the Bottom-Up and SWAB models exhibit a strong correlation with the absolute sum of change and the mean absolute change.

5. CONCLUSION

This paper has presented a detailed approach to developing and enhancing segmentation methods for application in anomaly detection within time series data. By collecting and constructing comprehensive datasets, the study has established a solid foundation for experimentation and evaluation.

Through the segmentation process and analysis of empirical results, the improved methods have been demonstrated to be effective in detecting anomalous points. The selection of the maximum allowable error

(max_error) has also been thoroughly discussed, enabling optimization of both accuracy and efficiency for the segmentation techniques.

The findings and insights from this paper provide a valuable basis for future research and practical applications in the field of anomaly detection in time series data.

6. REFERENCES

[1] Ana Azevedo, " Advanced Methodologies and Technologies in Network Architecture, Mobile Computing, and Data Analytics", Data Mining and Knowledge Discovery in Databases, **2019**.

[2] A. Blázquez-García, A. Conde, U. Mori, and J. A. Lozano, "A review on outlier/anomaly detection in time series data," ACM Computing Surveys (CSUR), vol. 54, no. 3, pp. 1-33, **2021**.

[3] Ángel Carmona-Poyato, Nicolás Luis Fernández-García, Francisco José Madrid-Cuevas, Antonio Manuel Durán-Rosal, " Pattern Recognition Letters", A new approach for optimal time-series segmentatio, pp. 153-159, **2020**.

[4] Ehsan Jolous Jamshidi, Yusri Yusup, John Stephen Kayode, Mohamad Anuar Kamaruddin, " Ecological Informatics", Detecting outliers in a univariate time series dataset using unsupervised combined statistical methods: A case study on surface water temperature", **2022**.

[5] Evimaria Terzi, Panayiotis Tsaparas (**2006**), "Proceedings of the 2006 SIAM International Conference on Data Mining", Efficient algorithms for sequence segmentation, 316-327.

[6] H. Cheng, P.-N. Tan, C. Potter, and S. Klooster, "Detection and characterization of anomalies in multivariate time series," in Proceedings of the 2009 SIAM international conference on data mining, pp. 413-424, **2009**: SIAM.

[7] Kumar G. Ranjan, Debesh S. Tripathy, B Rajanarayan Prusty, Debashisha Jena, "International Journal of Numerical Modelling Electronic Networks, Devices and Fields", An improved sliding window prediction-based outlier detection and correction for volatile time-series, **2020**.

[8] Lovrić, Miodrag; Milanović, Marina; Stamenković, Milan "Algoritmic methods for segmentation of time series: An overview", *Journal of Contemporary Economic and Business Issues*, 31-53, **2014**.

[9] Lovrić, Miodrag; Milanović, Marina; Stamenković, Milan, "Journal of Contemporary Economic and Business Issues", Algoritmic methods for segmentation of time series: An overview, pp. 31-53, **2014**.

[10] S.-E. Benkabou, K. Benabdeslem, and B. Canitia, "Unsupervised outlier detection for time series by entropy and dynamic time warping," *Knowledge and Information Systems*, vol. 54, no. 2, pp. 463-486, **2018**.