GUISSE: A Graphical User Interface For Snippet Selection and Evaluation in Time Series Data

Guilherme Fernandes[1], Lucas Peres Gaspar [1], Lívia Almada Cruz[1], Regis Pires Magalhães[1], Jose Antonio Macêdo[1]

[1]Insight Data Science Lab
Federal University of Ceará – Fortaleza – CE – Brazil

{guilherme.sales,lucasperes, livia, regis, jose.macedo}@insightlab.ufc.br

Abstract. *Time series snippet discovery offers powerful tools for summarizing complex temporal data via representative subsequences. While advanced methods, such as Matrix Profile-based approaches and our RS4 (Restricted Search Space for Snippet Selection), achieve strong efficiency and pattern fidelity, their adoption remains limited by usability barriers. Existing algorithms typically produce raw numerical outputs that require technical expertise to interpret, and few accessible interfaces support configuration or visualization. In this demo, we present GUISSE, an interactive graphical interface for executing and evaluating snippet discovery algorithms. The platform enables flexible experimentation with RS4 and baseline methods, allowing users to configure clustering techniques, distance weightings, and snippet parameters. By bridging algorithmic execution with interactive exploration, GUISSE empowers a broad range of users to apply, interpret, and validate snippet discovery results without programming expertise, promoting transparent and reproducible time series summarization workflows.*

1. Introduction

Pattern discovery is a widely adopted strategy for summarizing complex time series data across many application domains. For example, vital signs such as ECG and blood pressure can be monitored for early detection of abnormal sleep patterns [Imani et al. 2018]; electricity demand can be forecasted from consumption time series [Liu et al. 2022]; and meteorological data can support the analysis of extreme weather events [Das and Ghosh 2018]. Traditional approaches rely on detecting motifs, clusters, or predictive sub-sequences. More recently, *snippet discovery* has emerged as a promising method to summarize time series by extracting representative subsequences that frequently occur throughout the data [Imani et al. 2018]. Figure 1 is an illustrative example of representative snippets that cover 31.37% of the time series. The cover area quantifies the proportion of the series best explained by this snippet.

While advanced algorithms, such as Matrix Profile-based methods and RS4 (Restricted Search Space for Snippet Selection), offer strong accuracy and efficiency in discovering meaningful snippets, their practical adoption remains limited. A key barrier is usability; most existing implementations produce raw numerical outputs that require programming expertise to interpret. Additionally, configuring and executing these algorithms often involves writing complex code, limiting their accessibility to domain experts.

To address these challenges, we present GUISSE (Graphical User Interface for Snippet Selection and Evaluation), an interactive tool that simplifies the application of snippet discovery algorithms through a user-friendly visual interface. GUISSE allows users to execute and compare multiple algorithms, including our previously validated method, RS4 [Fernandes et al. 2025], baseline approaches, configure clustering and distance parameters, and explore discovered snippets through visualizations embedded in the original time series.

Existing tools for time series analysis, such as STUMPY [Law and Cervone 2021], UCR Suite [Zhu et al. 2016], and Sktime [Löwe et al. 2022], provide functionalities for motif discovery, anomaly detection, and subsequence search, but offer limited or no support for snippet selection. While STUMPY implements the Snippet-Finder algorithm, it lacks a graphical interface and flexibility for algorithmic comparison. To our knowledge, GUISSE is the first tool to provide an interactive graphical interface specifically designed for snippet selection.

The remainder of this paper is organized as follows: Section 2 provides an overview of the supported snippet discovery algorithms; Section 3 describes the main features of the GUISSE interface; Section 4 presents a demonstration scenario using real-world data; and Section 5 concludes the paper and outlines future work.

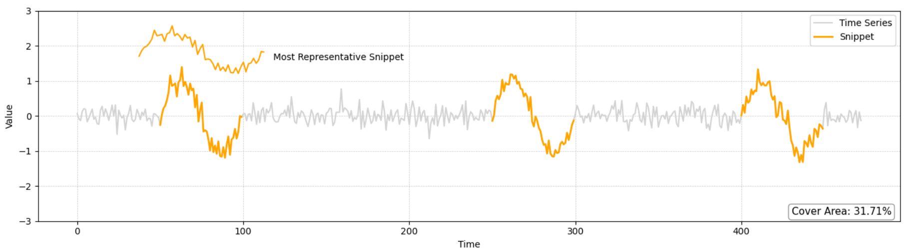


Figure 1. Visualization of a representative snippet matched at multiple, dispersed locations in the time series (gray).

2. Algorithms for Snippet Selection

In this section, we briefly describe the two snippet discovery algorithms supported by the GUISSE platform: Snippet-Finder [Imani et al. 2018] and RS4 [Fernandes et al. 2025]. Both aim to extract representative subsequences (*snippets*) that summarize frequent patterns in time series data.

2.1. Snippet-Finder

Snippet-Finder [Imani et al. 2018] is a state-of-the-art algorithm for snippet discovery based on the Matrix Profile framework [Yeh et al. 2016]. It performs an exhaustive search to identify the set of k snippets that minimize the overall profile area, ensuring high coverage and representativeness.

The algorithm computes an MPDist-based distance profile for each candidate snippet against the entire time series, iteratively selecting snippets that contribute most to explaining the data. While highly robust, this approach incurs a computational cost of $O(M^2)$, limiting its scalability for long-duration or high-resolution time series.

2.2. RS4: Restricted Search Space for Snippet Selection

RS4 [Fernandes et al. 2025] is a hybrid algorithm designed to improve the scalability of snippet discovery while preserving pattern fidelity. It combines clustering techniques with a restricted Matrix Profile Distance (MPDist) search, reducing the computational complexity compared to exhaustive approaches.

The algorithm operates in two stages. First, all overlapping subsequences of fixed length m are extracted from the input time series $\mathcal{T}$ and z-normalized. These subsequences are clustered into n groups using a configurable clustering method (e.g., Mini-Batch K-Means, K-Shape, etc.). From each cluster, the *medoid* is selected, the subsequence that minimizes the average distance to all others in its cluster.

In the second stage, a restricted MPDist profile is computed for each medoid against the full set of subsequences. The profiles are merged, and the k subsequences with the lowest scores are selected as the final snippets. This strategy balances coverage and diversity while avoiding the quadratic cost of computing pairwise distances for all subsequences, achieving an overall complexity of $O(Mn)$, where M is the number of subsequences and $n \ll M$.

3. GUISSE: Graphical User Interface for Snippet Selection and Evaluation

GUISSE is an interactive open-source platform designed to support the complete pipeline of time series snippet discovery and evaluation. It was developed to fulfill three primary objectives: (i) the execution of snippet selection algorithms; (ii) the visualization and inspection of the extracted snippets; and (iii) the comparative analysis of algorithmic performance.

The platform allows the execution of both the RS4 algorithm and the baseline Snippet-Finder approach. It also supports the import and visualization of results generated by external methods, making it extensible and suitable for benchmarking.

The snippet discovery process in GUISSE is fully configurable. Users can define key hyperparameters such as the snippet length, the clustering strategy, and the number of clusters. Additionally, the matching threshold can be specified to control the sensitivity of snippet matching within the original time series. GUISSE includes access to the publicly available *MixedBag* dataset, a benchmark composed of 100 time series from diverse domains, each containing two distinct behavioral segments, and supports the upload of custom datasets to enable broader experimentation and evaluation.

The visualization module presents an interactive and dynamic representation of the snippets, overlaying them directly on the original time series. This enables intuitive inspection of how representative each snippet is in the context of the full series. Zooming, panning, and series selection functionalities are provided to enhance user exploration and facilitate detailed qualitative analysis.

Furthermore, GUISSE offers a comparative metrics module to support the systematic evaluation of different algorithms. Users can assess a variety of performance indicators, including coverage, memory usage, and execution time. These metrics are presented in both tabular and graphical formats.

The GUISSE platform is publicly available as open-source software and can be accessed through its official GitHub repository[1], which includes the full implementation, usage instructions, and examples. A demonstration video highlighting the core functionalities of the tool is also available online[2]. Additionally, an online deployment of GUISSE is accessible for immediate testing[3].

4. Demonstration

This section describes the application of GUISSE for extracting, visualizing, and comparing time series snippets. The process involves the configuration of algorithm parameters, execution of snippet discovery, and analysis of the results.

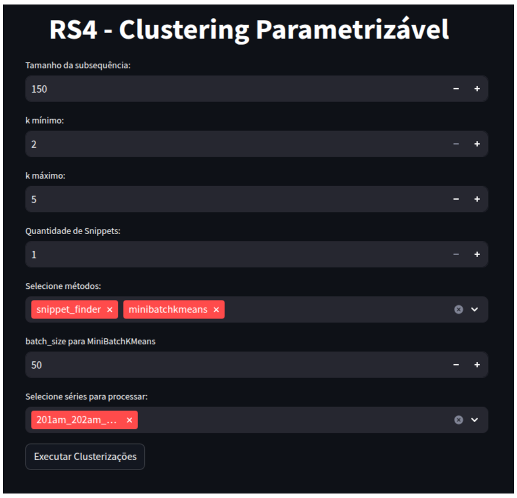
Data Upload: By default, GUISSE provides support for evaluating the public Mixed-Bag dataset. Additionally, users can upload custom datasets for specific experiments.

Execution: This scenario illustrates the execution workflow of the snippet selection process within the GUISSE interface. The user begins by specifying key configuration parameters, including the size of the subsequences to be extracted, the range of clusters to evaluate (defined by minimum and maximum values of k), the number of representative snippets to retrieve, and the clustering method to be used in the RS4 algorithm. Depending on the chosen clustering method, the interface dynamically provides access to the relevant hyperparameters. Figure 2 presents the execution screen during the configuration stage (2a) and after the computation has been completed (2b). After execution, the system displays summary metrics such as execution time, memory consumption, and the coverage and profile areas obtained by the resulting snippets; results can be downloaded as a compressed `.zip` file. Alternatively, downloading is not required for immediate visualization or comparison purposes, the interface automatically detects and loads the results for integrated exploration.

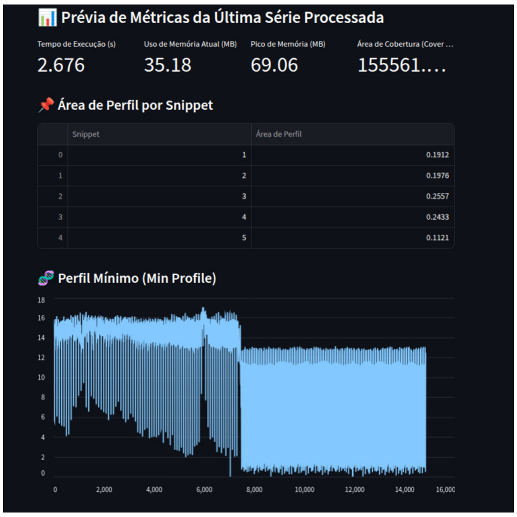
Data Visualization: This scenario demonstrates how GUISSE can be used to visualize snippets obtained by one or more algorithms executed in previous steps. Figure 3a shows the main interface of the visualization tab. Figure 3b displays the complete set of discovered snippets and the original time series with the corresponding snippets precisely overlaid on the segments where they match the data. The interface allows detailed navigation through the time series using zoom-in and zoom-out operations. It is also possible to enable or disable the visualization of results from different algorithms.

Metrics comparison: Finally, users can compare the performance metrics of different algorithms applied to the same time series. Figure 4a shows a screenshot of metrics tables of approaches, while Figure 4b displays metrics such as execution time, memory usage, and number of matches.

[1] https://github.com/GuiSales404/GUISSE
[2] http://tiny.cc/guisse
[3] https://guisales404-guisse-apphome-bwauu7.streamlit.app/

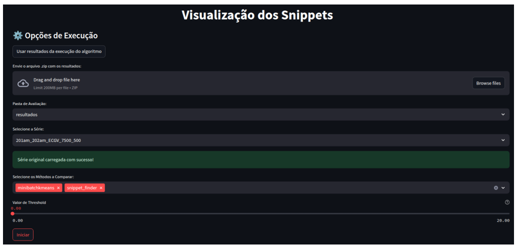


(a)

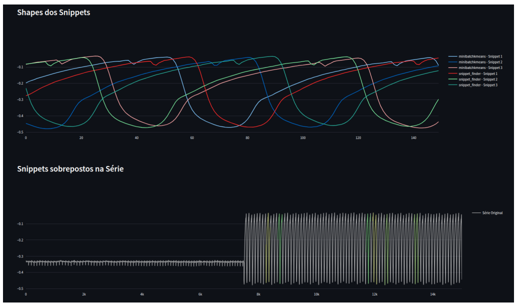


(b)

Figure 2. Graphical user interface of GUISSE: (a) snippet extraction configuration and (b) visualization.



(a)



(b)

Figure 3. Graphical user interface of GUISSE: (a) Snippet visualization matches on time series configuration and (b) visualization of snippets obtained from distinct approaches.

5. Conclusion

In this work, we introduced **GUISSE**, a comprehensive and extensible graphical interface for the discovery, visualization, and comparative evaluation of representative snippets in time series data.

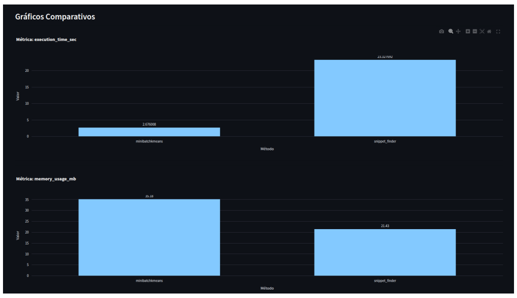
The platform offers interactive visualizations that overlay discovered snippets on the original time series, alongside detailed performance metrics such as execution time, memory usage, and coverage indicators. These capabilities promote reproducibility, enable insightful analysis, and facilitate benchmarking across different algorithms and parameter configurations.

By lowering the barrier to entry for experimenting with time series snippet extraction, GUISSE provides researchers and practitioners with a powerful tool to investigate temporal patterns. As future work, we aim to extend GUISSE by incorporating additional state-of-the-art algorithms, enhanced visualization modules, and automated evaluation pipelines to support more robust and scalable analyses in real-world time series applications.

Tabela de Métricas

	Método	Métrica	Valor
0	minibatchkmeans	execution_time_sec	2.572107
1	minibatchkmeans	memory_usage_mb	35.77
2	minibatchkmeans	peak_memory_mb	70.47
3	minibatchkmeans	profile_area	[0.48806784563802513, 0.5119321543619749]
4	minibatchkmeans	min_profile_area	[16.13797104282483, 16.372055003245055, 16.592433101797972, 16.585530924681205, 16.658
5	minibatchkmeans	cover_area	247825.02117918024
6	snippet_finder	execution_time_sec	22.802551
7	snippet_finder	memory_usage_mb	23.12
8	snippet_finder	peak_memory_mb	47.11
9	snippet_finder	profile_area	[0.16336861481822365, 0.24995069357701663, 0.21747419630530537, 0.19775162711195846, 0

(a)



(b)

Figure 4. Graphical user interface of GUISSE: (a) Snippet distinct approaches metrics table and (b) visualization of metrics from distinct approaches.

Acknowledgments. Part of the results presented in this work were obtained through the project "Center of Excellence in Artificial Intelligence - AI4Wellness", funded by Samsung Eletrônica da Amazonia Ltda., in accordance with the Brazilian Informatics Law No. 8.248/91.

References

Das, M. and Ghosh, S. K. (2018). Data-driven approaches for meteorological time series prediction: a comparative study of the state-of-the-art computational intelligence techniques. *Pattern Recognition Letters*, 105:155–164.

Fernandes, G., Gaspar, L. P., Cruz, L. A., Magalhães, R. P., and Macedo, J. A. (2025). Rs4: Restricted search space for snippet selection. In *Intelligent Systems*. Springer Nature Switzerland.

Imani, S., Madrid, F., Ding, W., Crouter, S., and Keogh, E. (2018). Matrix profile xiii: Time series snippets: A new primitive for time series data mining. In *2018 IEEE International Conference on Big Knowledge (ICBK)*, pages 382–389.

Law, S. S. and Cervone, D. (2021). Stumpy: A powerful and scalable library for time series data mining in python. *Journal of Open Source Software*, 6(60):2794.

Liu, X., Sun, H., Han, S., Han, S., Niu, S., Qin, W., Sun, P., and Song, D. (2022). A data mining research on office building energy pattern based on time-series energy consumption data. *Energy and Buildings*, 259:111888.

Löwe, S., Schmidl, S., Fambom, K., et al. (2022). sktime: A unified interface for machine learning with time series. *arXiv preprint arXiv:2209.06844*.

Yeh, C.-C. M., Zhu, Y., Ulanova, L., Begum, N., Ding, Y., Dau, H. A., Silva, D. F., Mueen, A., and Keogh, E. (2016). Matrix profile i: All pairs similarity joins for time series: A unifying view that includes motifs, discords and shapelets. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pages 1317–1322.

Zhu, Y., Zimmerman, Z., Senobari, N. S., Yeh, C.-C. M., Funning, G., Mueen, A., and Keogh, E. (2016). Matrix profile ii: Exploiting a novel algorithm and gpus to break the one hundred million barrier for time series motifs and joins. In *IEEE ICDM*, pages 739–748.