Multivariate Time Series Retrieval with Symbolic Aggregate Approximation, Regular Expression, and Query Expansion

Y. Yu\textsuperscript{1,2}, T. Becker\textsuperscript{1,2} and M. Behrisch\textsuperscript{1}

\textsuperscript{1}Utrecht University, Netherlands
\textsuperscript{2}IAV GmbH Ingenieurgesellschaft Auto und Verkehr, Germany

Figure 1: Pipeline of the proposed method: 1) Stringify the time series to allow text retrieval methods; 2) zipping tracks to enable multivariate retrieval; 3) extract regex from the stringified query; 4) expand the query to tackle distortions; 5) search with regex engine.

Abstract

We present SAXRegEx, a method for pattern search in multivariate time series in the presence of various distortions, such as duration variation, warping, and time delay between signals. For example, in the automotive industry, calibration engineers spontaneously search for event-induced patterns in fresh measurements under time pressure. Current methods do not sufficiently address duration (horizontal along the time axis) scaling and inter-track time delay. One reason is that it can be overwhelmingly complex to consider scaling and warping jointly and analyze temporal dynamics and attribute interrelation simultaneously. SAXRegEx meets this challenge with a novel symbolic representation modeling adapted to handle time series with multiple tracks. We employ methods from text retrieval, i.e., regular expression matching, to perform a pattern retrieval and develop a novel query expansion algorithm to deal flexibly with pattern distortions. Experiments show the effectiveness of our approach, especially in the presence of such distortions, and its efficiency surpassing the state-of-the-art methods. While we design the method primarily for automotive data, it is well transferable to other domains.

1. Introduction

When a car starts, a series of events are recorded for analysis and control. These event sequences are used in various use cases. Engine engineers await a signal to pinpoint the best entry point of their novel algorithm for a smoother air-fuel ratio control. Transmission engineers trace gear change intervals to analyze the moments on different shafts for less traction loss. Noise Vibration Harshness (NVH) engineers hunt for periods when some cylinders are deactivated or reactivated for cars with variable displacement to study the abrupt vibration during these periods. These use cases are all backed up by techniques for pattern search in time series.

Locating sub-sequences in time series similar to the given query is a frequent prerequisite for further data processing. Univariate retrieval with little distortion is efficiently solved with numerous time series indexing techniques [AJ02, GDK\textsuperscript{*}17]. Univariate case with certain distortion types, especially time shifts and sometimes horizontal scaling along the time axis, is also addressed [Keo97, JL18a, SSA\textsuperscript{*}18]. Recently, multivariate cases have received increasing attention [MJE15, LL18]. Multivariate distortion-invariant time series retrieval, however, remains largely untouched.

Our target signals constantly vary in duration and are often distributed in several tracks with significant inter-track time delay. Horizontal length scaling and time shifts within a track can be regarded as heterogeneous horizontal resampling. The dummy data in Figure 2 illustrate the distortions. These distortions are also likely to plague the patterns in data from other domains. Heterogeneous horizontal scaling is hard to capture, due to the de facto preprocessing step with a sliding window, usually assuming a fixed pattern length silently. Inter-track time shifts are even more challenging due to the complexity during simultaneous consideration of the temporal dynamics and the interrelation between attributes.

We propose an easy-to-implement method, SAXRegEx, based on symbolic representation to enable text retrieval methods and regular expressions as our search engine. Query expansion is designed...
to deal with the mentioned distortions. Our method shows comparable accuracy to state-of-the-art techniques on datasets without these distortions and better performance in multivariate datasets with mentioned distortions. Additionally, SAXRegEx outperforms the state-of-the-art in terms of speed.

2. Related Work

Research on time series retrieval mainly focused on two directions: first, novel similarity measures between two time series that better describe the notion of similarity (accuracy) [SSA*18, JL18a]; second, indexing techniques focusing on efficiency [LS95, GDK*21, PFP21]. Usually, sliding windows are assumed for preprocessing. Our work breaks the convention by employing neither a sliding window approach nor an explicit measure of similarity.

Besides random noise, patterns in the time series can have various distortion forms. Horizontal translation along the time axis is trivial; horizontal scaling is only addressed in a few works, such as [Kee97, MA18]; vertical translation and scaling on the value/amplitude axis is often handled by normalizing time series windows [LPH*20, GDK*21]. Warping is captured by a variety of “elastic” distances [CN04, AVG13], especially Dynamic Time Warping (DTW) [BC94, MJE15, JL18a]. In multivariate cases, most methods handle the temporal dynamics and interrelation between tracks independently, assuming that the tracks are synchronized. For instance, DTW_t calculates distance profiles for each track individually and merges them subsequently [MJE15]; Locality-Sensitive Hashing (LSH) merges tracks followed by univariate processing [CLL*19]. All these approaches assume that the pattern is synchronized in all tracks. Driven by our domain problems, we target this problem without this assumption and scan the temporal dynamics with inter-track relation jointly.

Recently, machine learning penetrates time series retrieval [JL18b, LXZ19, LPH*20]. They do not suit, however, highly dynamic or user-driven retrieval cases, where the queries and the datasets change rapidly. In our case, engineers start the search spontaneously and wish an immediate answer.

Regarding accuracy, it is accepted that no similarity measure accounts for the human notion of similarity [MM16]; therefore, different datasets favor different similarity measures. Some works propose an active learning strategy to adapt the similarity measure [KP98, LPH*20, YKBB21]. The extensive benchmark in [BLB*17] shows that DTW enjoys top ranking accuracy among 20 methods for time series classification. As for speed, Mueen’s Algorithm for Similarity Search (MSS) is so far the fastest similarity search algorithm [YZZ*16, MAA*21]. We, consequently, include both for benchmarking SAXRegEx (Section 4).
2, we zip the query while taking time shifts between the shapes in different tracks into consideration. We can add this flexibility by introducing the wildcard character “.” for “gaps.” This way, the query ignores the potentially interfering context in each track and focuses only on the given shape in the track. As a result, the shape in a track within the query does not have to span the whole query length. Note in Figure 3, the term ( . . . ) {2, 6}, which captures the elastic gap between the shapes in Track 3 and in Track 1.

4. Evaluation

We evaluated the accuracy and speed of SAXRegEx with eight labeled datasets, among which two contain patterns with heterogeneous horizontal scaling or inter-track time shifts. We chose four state-of-the-art methods for our comparative evaluation of SAXRegEx: 1) correlation: one of the standard similarity measures; 2) DTW: the most popular elastic distance measure for time series with “hard-to-beat” accuracy for time series classification [BLB+17]; 3) Euclidean Distance (ED): specifically MASS, the so-far fastest similarity search tool [YZU+16, MAA+21]; 4) SAX: our baseline time series representation, together with its similarity measure [LKWL07], can be used for similarity search. Details of the benchmark methods are in Appendix C and Appendix D.

All experiments were conducted on a standard laptop running on 64-bit Windows 10 Enterprise with Intel i7-8650U CPU, 16GB RAM, and 1TB HDD. Details of the datasets, hardware and software setup are in Appendix B and Appendix C, respectively.

4.1. Accuracy Benchmark

We compared the performance of all methods quantitatively with standard metrics and qualitatively through a visual inspection. Besides the standard metrics accuracy, balanced accuracy, precision, recall, and F1-score, which regard the problem as the binary classification for each time step (inside/outside a pattern), we want to introduce the metric mean Average Precision (mAP) from object detection in computer vision from a segment-wise perspective. We have chosen 30% and 50% as the Intersection over Union (IoU)-threshold required by this metric. They are denoted mAP30 and mAP50, respectively. For details, please refer to Appendix C. The table in Figure 4 (complete version in Appendix E) depicts our five evaluation metrics for all methods on various datasets. The result suggests that different data favor different methods and no method constantly outperforms the other, confirming the finding in [MM16]. However, SAXRegEx can better capture heterogeneous horizontal scaling and inter-track time shifts. We can observe this specifically on the last two datasets in Figure 4 and Figure E.7.

We have plotted the predictions and ground truth labels with all methods on all publishable datasets in Appendix D, together with the description of the plots. Figure 5 shows an example. From the visual inspection, we can infer pitfalls of the benchmark methods. Firstly, the benchmark methods spread focus evenly over all time steps rather than critical transitions (usually large ramps). This leads to the false positive Prediction 5 with ED, which misses an upward ramp at the beginning in the first track and a downward ramp at the end in the second track. Next, the benchmark methods take the context of the pattern shape in each track into account (please note the difference between the query for SAXRegEx and the benchmark methods). Consequently, the search can be misled by a changing context, as implied by the missing ground truth label 2 for ED and DTW. In this case, the target pattern contains a distracting plateau in each track. Finally, while DTW well addresses the time shifts within a track, as proven by the performance on the CAN 1 dataset in Figure 4 and Figure D.5, it has a hard time when it comes to time shifts between tracks, as indicated by Prediction 2, 3 and 4 for DTW. In the presence of such distortions, SAXRegEx outperforms the benchmarks and fulfills our needs.

4.2. Speed Benchmark

We calculated the elapsed time of all methods on all datasets. Every experiment is repeated five times. The complete result can be found...
in Figure F.8 in the appendix. In Figure F.9, we report on the relative performance gain compared to DTW.

In short, SAXRegEx is nearly x50 faster than DTW and x4.6 faster than MASS, the so-far fastest similarity search tool for time series retrieval. We attribute this speed boost to two reasons. First, SAXRegEx naturally captures horizontally scaled patterns. In contrast, the benchmarking methods use eight sliding windows with increasing window lengths, costing roughly x8 the time. However, even reducing the number of sliding windows to four, which is already too coarse, SAXRegEx still outperforms all benchmarking methods. Second, SAXRegEx has the notion of early termination as soon as the search engine notices a partial mismatch.

5. Discussion and Limitations

Our industry collaborator IAV deploys SAXRegEx in a software-as-a-service environment and applies it to data from engine control units, transmission control units and CAN bus data, where heterogeneous horizontal scaling of the pattern and inter-track time shifts occur regularly. It satisfactorily meets automotive engineers’ flexible needs for prompt search results.

SAXRegEx does not calculate similarity profiles. It makes sense to use a relatively large threshold, tolerating more false positives and filter the predictions with fine-grained distance metrics.

Our current query expansion does not cover vertical translation and scaling of the pattern perpendicular to the time axis. The user should ensure that they do not affect the patterns in the time series. One remedy for the vertical translation problem can be the usage of the derivatives instead of the original data.

Finally, the current implementation requires a user interface for query definition, progress monitoring, result inspection, and parameter tuning. We plan to collect requirements from the domain experts for the UI, survey the current Visual Query Systems (VQSs), and design a user interface emphasizing the query definition, e.g. for inter-track time-shifted patterns.

6. Conclusion

With SAXRegEx, we present a method for multivariate time series retrieval based on SAX, regex, and query expansion. It excels in capturing distorted patterns of various types. In particular, our search engine is ideal to find patterns that are scaled horizontally along the time axis heterogeneously or patterns showing inter-track time shifts, while significantly outperforming state-of-the-art methods in terms of speed. SAXRegEx assists automotive engineers quickly find patterns related to various events in the measurement. Nonetheless, the method itself is not limited to any domain-specific prerequisites and can be used in other domains as well. In the future, we plan to design a user interface focusing on the query definition with various kinds of distortions.
References


© 2022 The Author(s)
Europgraf突破s Proceedings © 2022 The Europgraf Association.