Efficient Query Processing in Time Series

Yuhong Li
Expected Graduation: 2016
Supervisors: Zhiguo Gong & Leong Hou U
Dept. of Computer and Information Science, University of Macau
Avenida da Universidade, Taipa, Macau
{yb27407,fstzgg,ryanlhu}@umac.mo

ABSTRACT

With the rapid development over the last decade, time series data become one of the most frequently used data in real world applications (e.g., finance analysis, medical diagnosis, environmental monitoring, etc.). As expected, the volume of the time series data will even grow larger in near future. It is important to design efficient and effective algorithm and index to handle various tasks for these data. Thereby, my PhD study focuses on how to extract meaningful time series patterns from large volume of data efficiently. Specifically, two types of extraction queries are discussed in this work, including longest-lasting correlated subsequence query and time series motif query. The applications and solutions of these two queries are thoroughly introduced and discussed in this paper. Moreover, some potential pattern extraction queries will also be discussed in this paper.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data mining*

Keywords

Time Series; Longest-lasting Correlation; Time Series Motif

1. INTRODUCTION

Time series data can be found in a wide variety of domains, including medical diagnosis, speech processing, financial analysis and environmental monitoring, etc. As a consequence, processing and mining of time series data have been developed in the last decade, such as subsequence matching [7, 26], classification and clustering [17, 23], prominent streak and anomaly detection [6, 9]. In this work, we focus on discovering longest-lasting correlated subsequence [15] and time series motif [16, 20, 24].

Longest-lasting Correlation. The first problem studied in this paper is discovering longest-lasting correlated subsequence (LCS). Given two time series q and o, a subsequence pair is longest-lasting correlated if and only if the length of the subsequence ℓ is maximized subject to $\rho(q, o, \tau, \ell) \geq \delta$, where $\rho(q, o, \tau, \ell)$ is the Pearson correlation between q and o in the segment $[\tau, \tau + \ell - 1]$.

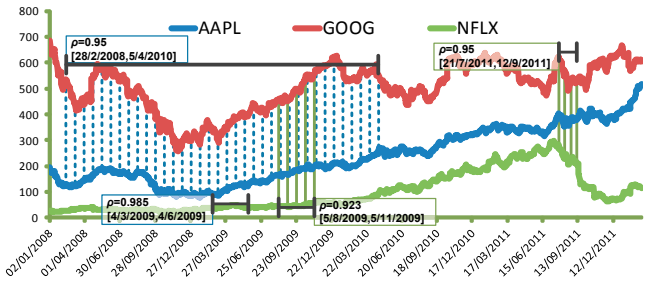


Figure 1: Example of financial data analysis

Longest-lasting correlated subsequence are particularly useful in helping those analyses without prior knowledge of the query length ℓ. For instance, a stock analyst wants to find a stock whose price variance is similar to Google, Inc. in some segment from 2008 to 2011. This question can be answered easily by subsequence matching [7] if we have prior knowledge about the segment length ℓ. However, ℓ is not easy to be specified as the most appropriate value of ℓ heavily depends on queries, time, data, and application domains. Instead of finding fixed length results, our task is to return the longest-lasting segment whose correlation is larger than a threshold δ. We claim that the correlation threshold δ is a more natural parameter than the segment length ℓ since analysts can evaluate how the correlation score reflects the relevance of the result in their application domain.

We demonstrate longest-lasting correlated subsequence query using the stock data collected from Yahoo! Finance [1]. Fig. 1 illustrates the price variance of *GOOG* (Google, Inc.), *NFLX* (Netflix, Inc.), and *AAPL* (Apple, Inc.) in 2008 - 2011. A typical analysis query is '*find the most correlated stock to GOOG for every 3 months data in 2008-2011*'. This query returns

[1]http://finance.yahoo.com/

SIGMOD'15 PhD Symposium, May 31, 2015, Melbourne, Victoria, Australia.

http://dx.doi.org/10.1145/2744680.2744688 .

AAPL in [4/3/2009,4/6/2009] as the result where the correlation is 0.985. The query result may be more meaningful to analysts if it becomes '*find the longest-lasting period of a stock who performs similar to GOOG in 2008-2011*'. Suppose that the correlation threshold is set to 0.95, this query returns *AAPL* in [28/2/2008,5/4/2010] as the result. It is not surprising that their prices change similarly over such long period as both of them are the leading companies in IT sector. Besides, the second type of queries can identify prominent periods more precisely based on the correlations. For instance, the longest correlated time span of *GOOG* and *NFLX* fulfilled the correlation threshold is [21/7/2011,12/9/2011] while the fixed length query (ℓ=3 months) returns a lower correlation ($\rho = 0.923$) in time span [5/8/2009,5/11/2009].

Time Series Motif. The second problem studied in this paper is the time series motif discovery. Time series motif has been shown to have great utility for several data mining algorithms, including clustering, classification, sequence summarization, and rule discovery [5, 14, 21, 22, 24]. Given a time series, it reports the *motif* as the most correlated pair of subsequences in this time series. The correlation between subsequences is measured by the normalized Euclidean distance. As an example, Fig. 2 illustrates a weekly motif discovered in a power consumption dataset [13].

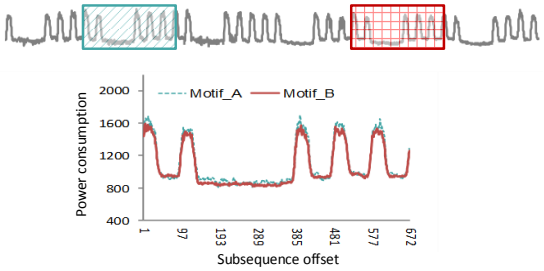


Figure 2: Weekly motif discovered in a time series (35,000 values) that records the average power consumption for a Dutch research facility in the year 1997

Discovering motif is a core subroutine for activity discovery of humans and animals, with applications in elder care, surveillance and sports training [21]. Besides, clustering enumerated motifs is shown to be more meaningful than clustering all the subsequences in a long time series [27]. As in other domains, this approximately repeated structure may be conserved for some reason that is of interest to domain specialists [22].

The remainder of this paper is organized as follows. The notations and related works are presented in Section 2. The works we have done are presented in Section 3, followed by future work in Section 4. Lastly, we conclude the paper in Section 5.

2. PRELIMINARIES

2.1 Subsequence, Z-normalization, Distance measures

Definition 1 (Subsequence of o). *Given a time series o of length m (i.e., $o = o[0]...o[m-1]$), a valid subsequence of length ℓ (i.e., $0 < \ell \leq m$) in o is denoted as $o(\tau, \ell) = o[\tau]...o[\tau+\ell-1]$, where τ is the offset of the subsequence and $0 \leq \tau < m-\ell+1$.*

Z-normalization To reasonably compare the similarity of two time series, the sequence values should be Z-normalized [15, 26]. The Z-normalization is to transform a time series into its normalized from whose mean is approximately zero, and the standard deviation is in a range close to 1. Mathematically, the i-th value of a Z-normalized time series $\hat{o}$ can be calculated by

$$\hat{o}[i] = \frac{o[i] - \mu_o}{\sigma_o} \tag{1}$$

where μ_o and σ_o are the mean and standard deviation of o, respectively.

Distance measures In motif discovery, we use normalized Euclidean distance as the underline distance measure. Generally, the normalized Euclidean distance between two time series is calculated by their normalized form. For clarity, the definition of normalized Euclidean distance is given as follows.

$$dist(\hat{q}, \hat{o}) = \sqrt{\sum_{i=0}^{m-1} (\hat{q}[i] - \hat{o}[i])^2} \tag{2}$$

It is obviously unfair to compare the subsequence similarity of different lengths. As an example, the normalized Euclidean distance of two longer subsequences is more likely larger than the normalized Euclidean distance of two shorter subsequences. Thus in longest-lasting correlation query, we use Pearson correlation as the underline similarity measure since it not only reveals the true similarity of time series by Z-normalization but also makes the similarity comparison fairer by *length normalization* [15]. For clarity, the definition of Pearson correlation is given as follows.

$$\rho(q, o) = \frac{\sum_{i=0}^{m-1} q[i]o[i] - m\mu_q\mu_o}{m\sigma_q\sigma_o} \tag{3}$$

Actually, the Pearson correlation between two time series can be represented by their normalized Euclidean distance as follows.

$$\rho(q, o) = 1 - \frac{(dist(\hat{q}, \hat{o}))^2}{2m} \tag{4}$$

where $dist(\hat{q}, \hat{o})$ is normalized by the length m in the Pearson correlation.

2.2 Piecewise Aggregate Approximation

In order to boost up these two queries, i.e., LCS and time series motif, the proposed methods utilize indexable dimensionality reduction methods for time series data. More specifically, an indexable dimensionality reduction must obeys the *lower bound lemma* in order to filter unpromising candidates without false dismissals. In these two works, we use Piecewise Aggregate Approximation (PAA) [12, 29] as the dimensionality reduction method as it is simple yet shown to be competitive with other dimensionality reduction representations like SVD, DFT and DWT as discussed in [19].

Specifically, a normalized time series $\hat{o}$ can be represented by ϕ line segments of equal length $\frac{m}{\phi}$. Formally, given a normalized time series $\hat{o}$, the k-th element (i.e., k-th line segment) of its ϕ-dimensional PAA representation is defined as follows.

$$e_{\hat{o}}[k] = \frac{\phi}{m} \sum_{x=\frac{m}{\phi}\cdot k}^{\frac{m}{\phi}(k+1)-1} \hat{o}[x] \quad (5)$$

By [12, 29], the PAA distance d_{PAA} between two PAA representations $e_{\hat{q}}$ and $e_{\hat{o}}$ serves as the lower bound of the Euclidean distance between their representative time series $\hat{q}$ and $\hat{o}$:

$$dist(\hat{q}, \hat{o}) \geq d_{PAA}(e_{\hat{q}}, e_{\hat{o}}) = (\frac{m}{\phi} \sum_{x=0}^{\phi-1} (e_{\hat{q}}[x] - e_{\hat{o}}[x])^2)^{\frac{1}{2}} \quad (6)$$

2.3 Related Work

Processing and mining time series data have received plenty of attention in database and data mining community in the last two decades, such as similarity search and subsequence matching [1, 2, 3, 4, 7, 8, 10, 25, 26], classification and clustering [17, 23], prominent streak and anomaly detection [6, 9], etc.

In order to discover LCS efficiently, our proposed method required a multi-length index to support batch pruning for subsequence queries of different lengths. A related work called Multi-Resolution Index (MRI) [10] deals with arbitrary length queries more efficiently than I-adaptive [7] which employs prefix search to support longer query. MRI is a collection of I-adaptive indexes at different based-2 resolutions. At query time, hierarchical prefix search can be applied on MRI. However, as mentioned in [26], prefix search can only work for non-normalized distance measures. Rakthanmannon et al. [26] proposed an non-index method to boost up the normalized arbitrary length subsequence search for both normalized Euclidean distance and Dynamic Time Warping (DTW) using the *reordering early abandon* and *cascading lower bounds*. A main drawback of the non-index solutions is that they do not support batch pruning, e.g., if a query q is not similar to a subsequence, then q is unlikely similar to its neighbor subsequences.

Regarding to motif discovery, most of the literature focuses on fast algorithms for approximate motif discovery [5, 14, 18]; however, they do not provide guarantees on the result quality. Recently, Mueen et al. [20, 24] propose two efficient algorithms for exact motif discovery. The smart brute force method (SBF) [20] examines subsequence pairs in a specific ordering in order to compute the distance of each pair incrementally in (amortized) constant time. However, this method always examines $O((m-\ell)^2)$ subsequence pairs as it cannot prune any subsequence pair. On the other hand, the reference indices method (MK) [24] examines subsequences by the order of distance to a reference, and employs a pruning technique to discard unpromising subsequence pairs. Then, it computes the distance for each remaining pair (in $O(\ell)$ time). Nevertheless, its pruning effectiveness relies heavily on the data distribution. Also, MK requires considerable memory space for storing all normalized subsequences and reference indices.

3. RESEARCH UNDERTAKEN

In the past few years, I have been working on designing efficient methods for time series query processing. More specifically, I have studied two queries: discovering longest lasting correlated subsequence [15] and discovering time series motif [16].

3.1 Longest-lasting Correlation

Problem We formally define the Longest-lasting Correlated Subsequence (LCS) Query as follows.

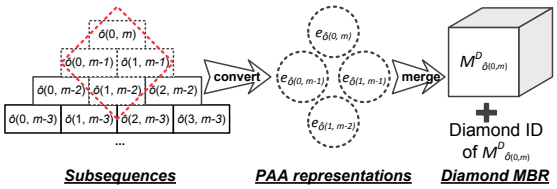
PROBLEM 1. (LONGEST-LASTING CORRELATED SUBSEQUENCE QUERY) *Given a query time series q, a time series database O, and a threshold δ, a Longest-lasting Correlated Subsequence (LCS) Query returns a subsequence $o_{lcs}(\tau_{lcs}, \ell_{lcs}), o_{lcs} \in O$ such that $\rho(q, o_{lcs}, \tau_{lcs}, \ell_{lcs}) > \delta$ and the length ℓ_{lcs} is the longest among all possible subsequences $o_i(\tau_i, \ell_i), o_i \in O$ having $\rho(q, o_i, \tau_i, \ell_i) > \delta$.*

Discovering LCS is a challenging problem due to (1) its potentially huge search space and (2) no monotonic property (with regard to subsequence length) held for normalized distance measures [15]. To find the longest-lasting correlation subsequence, a näive method is to search every possible subsequence from the longest length to the shortest length until it finds a subsequence $o_i(\tau, \ell)$ such that ℓ is maximized subject to the correlation constraint δ. Thereby, the total search space is $O(nm^2)$ and the time complexity is $O(Cnm^2)$, where n is the number of time series in O, m is the maximum length of the time series, C is the complexity of a correlation computation.

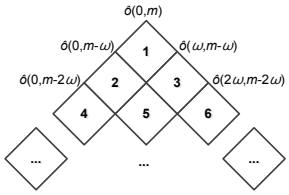
Methodology Obviously, the aforementioned naïve method which checks every possible subsequence combination is not scalable for large datasets. To the best of our knowledge, there is no known index that can support normalized distance queries of arbitrary lengths [26]. In this work, we first introduce a space-constrained index to prune unnecessary subsequences by batch. Our index exploits an observation that subsequences in a time series o having similar offsets τ and lengths ℓ are likely to have similar correlation with q. For instance, if $\rho(q, o, \tau, \ell) < \delta$, then $\rho(q, o, \tau, \ell - 1)$ and $\rho(q, o, \tau + 1, \ell - 1)$ are likely to be smaller than δ. This suggests that we can group similar subsequences together. Specifically, if the correlation upper bound of a subsequence group is below δ, then such a group can be safely pruned.

Based on this observation, we first group normalized subsequences of o (with arbitrary lengths) by their PAA representations into Minimum Bounding Rectangles (MBRs), and we show that the minimum distance between PAA MBRs can be used to derive the correlation upper bounds. Fig. 3 illustrates our proposed diamond covering grouping strategy. It groups all m^2 normalized subsequences in o into a set of ω-diamond MBRs (cf. Fig. 3(b)), where each ω-diamond MBR contains w^2 subsequences (cf. Fig. 3(a)).

In order to further reduce the index size, we propose to group similar ω-diamond MBRs with the same id (e.g., $M^D_{o_1(\hat{0},m)}$ and $M^D_{o_9(\hat{0},m)}$) in different time series into higher level MBRs called *compact* MBRs. Fig. 4 illustrates this kind of inter-object grouping. By using this intra-object and inter-object grouping, the built index can be controlled under a manageable size.



(a) Diamond MBR example, with $\omega = 2$



(b) Diamond cover grouping

Figure 3: Illustration of ω-diamond cover grouping

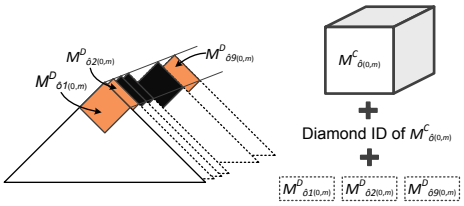


Figure 4: Inter-object grouping

During the discovery of LCS, we construct ω-diamond MBRs of q on demand, and perform filter-and-refinement based on these *compact* MBRs. For the un-pruned subsequences, we propose to reduce the correlation computation cost by pre-computing the α-skipping cumulative arrays.

Inspired by [23], the Pearson correlation of all subsequence pairs can be computed in $O(1)$ time if we compute the five cumulative arrays for q and o. These arrays are defined mathematically as follows.

$$S_q[u] = \sum_{i=0}^{u} q[i], \quad S_{q^2}[u] = \sum_{i=0}^{u} q[i]^2, \quad S_{qo}[u] = \sum_{i=0}^{u} q[i]o[i] \qquad (7)$$
$$S_o[u] = \sum_{i=0}^{u} o[i], \quad S_{o^2}[u] = \sum_{i=0}^{u} o[i]^2,$$

where $u \in [0..m-1]$.

However, the total space overhead is $3mn$ that is three times larger than the raw data.[2] Instead, we present a technique called α-skipping requires only $\frac{3mn}{\alpha}$ total space overhead (α can be chosen based on memory size). It can compute every Pearson correlation in $O(\alpha)$ as been proved in [15]. The α-skipping cumulative array is defined as follows.

Definition 2 (α-skipping cumulative array, $S^{\alpha}_{\mathcal{X}}$). *Let the skip factor be α. A cumulative value $S_{\mathcal{X}}[u]$ is kept into $S^{\alpha}_{\mathcal{X}}$ if and only if $u \mod \alpha = 0$. Here $\mathcal{X} \in \{o, o^2, qo\}$.*

[2]Every object o_i is required to construct 3 extra arrays (e.g., $S_{o_i}, S_{o_i^2}, S_{qo_i}$, and the space overhead (i.e., $O(2m)$) of S_q and S_{q^2} is negligible.

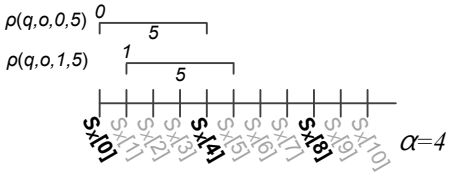


Figure 5: An α-skipping cumulative array

Fig. 5 shows a 4-skipping cumulative array ($\alpha = 4$). Note that only the cumulative values in bold color are kept in $S^{\alpha}_{\mathcal{X}}$, where the size of $S^{\alpha}_{\mathcal{X}}$ is $1/\alpha$ to the original size.

Contributions The contribution of this work can be summarized as below: 1. we define a new query, *longest-lasting correlated subsequence query*, in time series databases, which is useful in many real-world applications. 2. we propose an α-skipping technique to reduce the computation time of Pearson correlation from $O(\ell)$ to $O(\alpha)$. 3. we propose a size-tunable index, *diamond cover index*, to efficiently compute LCS, which is the first index to support arbitrary length subsequences search under normalized distance measure.

Outcomes Extensive experimental evaluation on both real and synthetic time series datasets, i.e., RAND, STOCK and TAO [15], verifies the efficiency and effectiveness of our proposed methods, and our best method is up to one order of magnitude faster than the state-of-the-art adaption.

3.2 Time Series Motif

Problem The formal definition of motif discovery is as follows:

Problem 2 (Motif Discovery). *Given a time series o of length m and the targeted motif length ℓ, the motif discovery is to return a pair of subsequences $\{o(i,\ell), o(j,\ell)\}$, where the normalized Euclidean distance of $o(i,\ell)$ and $o(j,\ell)$ is minimum among all non-trivial subsequence pairs.*

It is time consuming to solve the motif discovery problem. Note that a time series of length m contains $m-\ell+1$ subsequences of length ℓ. The brute force method would (i) examine all pairs of subsequences (i.e., $O((m-\ell)^2)$ pairs) and then (ii) compute the distance for each pair (in $O(\ell)$ time). This method takes $O((m-\ell)^2 \cdot \ell)$ time, which is too expensive for a long time series.

Methodology To the best of our knowledge, prior solutions

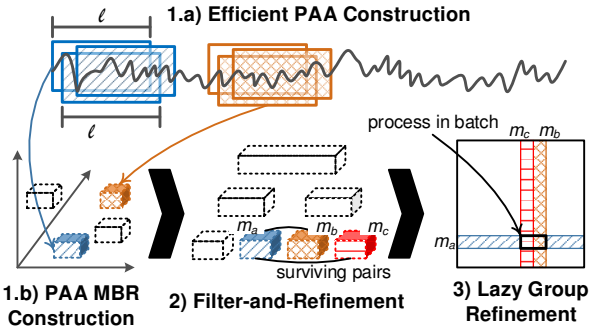


Figure 6: The framework of Quick-Motif

focus on one narrow aspect to boost the query performance. For instance, MK exploits the pruning capability using reference indices but it takes $O(\ell)$ time per distance calculation; SBF exploits the fast distance calculation in $O(1)$ time but without any pruning support. Thus, none of these solutions can offer acceptable performance for emerging applications when (i) the time series is long, e.g., millions of values, or (ii) the motif discovery query is issued frequently, e.g., every minute. In this work, we exploit more broadly such that our solution, Quick-Motif, is more *smarter* than SBF as equipped with batch pruning capability and it can be used to answer the motif discovery problem at scale.

Fig. 6 illustrates the work flow of our framework. To reduce the dimensionality of the problem, we first transform each subsequence into their PAA representation. To support fast distance calculation, we group consecutive PAA representations into PAA MBRs such that the pairwise subsequence distances of two MBRs can be computed in $O(1)$ time. To support batch pruning, we manage the MBRs into a Hilbert R-tree and apply a filter-and-refinement framework to prune unpromising MBR pairs. To further improve the performance, we propose a lazy group refinement technique which attempts to process surviving MBRs in one refinement batch. By taking the advantages of all these techniques, Quick-Motif outperforms the state-of-the-art approaches, which addresses the need of emerging applications.

Contributions In this work, we present a novel framework named Quick-Motif which adopts two-level approach to enable batch pruning at the outer level and enable fast distance calculation in inner level. While prior works (i.e., MK and SBF) can not offer fast distance computations and prune subsequence pairs at the same time, as these two techniques require different orderings on examining subsequence pairs. Furthermore, we propose two optimization techniques for both the outer and the inner levels: (i) a locality-based searching strategy for discovering the *true* motif as soon as possible, and (ii) a batch refinement technique that shares the processing cost of surviving group-pairs (i.e., promising pairs).

Outcomes We evaluate the proposed framework on both real and synthetic datasets, i.e., ECG, EEG, EPG, TAO and RAND [16]. The experimental results show that Quick-Motif outperforms the state-of-the-art methods. To the best of our knowledge, we are the first work that discovers motif in a time series of million lengths in ∼20s on a commodity machine (while other approaches take several hours to complete the same discovery task). The performance of our approach enables the possibility to offer online motif discovery in emerging applications.

4. FUTURE WORK

My previous works only studied query processing for one-dimensional time series data based on lock-step distance measures (e.g., normalized Euclidean distance and Pearson correlation). In the remaining study period of my PhD, I plan to propose efficient methods to support the query processing for *multi*-dimensional time series data based on other *flexible* distance measures (e.g., Dynamic Time Warping (DTW)).

4.1 Multi-Dimensional Time Series

A multi-dimensional time series consists of k individual time series ($k \geq 2$) where each individual time series is of the same length. Multi-dimensional time series are prevalent in diverse applications such as GPS tracking, motion capture and medical measurements. While considerable methods have been developed for query processing in an individual time series, relatively little work on processing multi-dimensional time series has been reported.

4.2 Dynamic Time Warping

Dynamic time warping (DTW) is a distance measure that is robust to misalignments and time warps, and it is widely used extensively in many applications [2, 11, 28]. In general, DTW finds the optimal alignment (i.e., minimum distance) between two time series by warping their offsets in a nonlinear fashion. Mathematically, DTW can be represented by a recursion as follows.

$$DTW(q,o) = (q[0]-o[0])^2 + \min \begin{cases} DTW(q[1...last], o) \\ DTW(q[1...last], o[1...last]) \\ DTW(q, o[1...last]) \end{cases} \quad (8)$$

where $q[1...last]$ denotes the subsequence of q containing values from the 2-nd to the last offset. To avoid pathological warping, many research works [11, 26] suggest to apply a constraint r in warping length such that $q[i]$ is matched with $o[j]$ if and only if $|i-j| \leq r$. This reduces the complexity of DTW from $O(m^2)$ to $O(mr)$.

Discussion: For processing a query on multi-dimensional time series, one possible solution is to use Principal Component Analysis (PCA) that projects the multi-dimensional time series into an one-dimensional time series [30] and then solve it by any one-dimensional solution for this query. However, this approach does not guarantee the accuracy of query results. In the future, I plan to study efficient methods which can avoid checking some individual time series of unnecessary dimensions while provides exact query result.

For DTW, Rakthanmannon et al. [26] proposed several lower bound techniques to speed up DTW computation without index support. It is still possible to further improve the performance. For example, distance calculation between a query sequence and several consecutive subsequences will share a lot of points, thus it holds a high possibility to prune all the subsequences by refining only a subsequence among them.

5. CONCLUSION

In this paper, two kinds of query processing for time series data are studied. In particular, longest-lasting correlation focus on discovering longest-lasting highly correlated subsequence in massive time series databases, and it is particularly useful in helping those analyses without prior knowledge about the query length. Motif discovery reports the most similar/correlated subsequence pair in a long time series which can be used as a core subroutine in a variety of domain applications.

In the future, we intend to study efficient query processing techniques for multi-dimensional time series under more flexible distance measurements, i.e., dynamic time warping.

6. REFERENCES

[1] I. Assent, R. Krieger, F. Afschari, and T. Seidl. The ts-tree: efficient time series search and retrieval. In *EDBT*, pages 252–263, 2008.

[2] V. Athitsos, P. Papapetrou, M. Potamias, G. Kollios, and D. Gunopulos. Approximate embedding-based subsequence matching of time series. In *SIGMOD conference, 2008*, pages 365–378, 2008.

[3] N. Beckmann, H.-P. Kriegel, R. Schneider, and B. Seeger. The r*-tree: An efficient and robust access method for points and rectangles. In *SIGMOD Conference*, pages 322–331, 1990.

[4] A. Camerra, T. Palpanas, J. Shieh, and E. J. Keogh. isax 2.0: Indexing and mining one billion time series. In *ICDM*, pages 58–67, 2010.

[5] N. Castro and P. J. Azevedo. Multiresolution motif discovery in time series. In *SDM*, pages 665–676, 2010.

[6] V. Chandola, V. Mithal, and V. Kumar. Comparative evaluation of anomaly detection techniques for sequence data. In *ICDM*, pages 743–748, 2008.

[7] C. Faloutsos, M. Ranganathan, and Y. Manolopoulos. Fast subsequence matching in time-series databases. In *SIGMOD*, pages 419–429, 1994.

[8] R. F. S. Filho, A. J. M. Traina, C. T. Jr., and C. Faloutsos. Similarity search without tears: The omni family of all-purpose access methods. In *ICDE*, pages 623–630, 2001.

[9] X. Jiang, C. Li, P. Luo, M. Wang, and Y. Yu. Prominent streak discovery in sequence data. In *KDD*, pages 1280–1288, 2011.

[10] T. Kahveci and A. K. Singh. Optimizing similarity search for arbitrary length time series queries. *IEEE TKDE*, 16(4):418–433, 2004.

[11] E. J. Keogh. Exact indexing of dynamic time warping. In *VLDB*, pages 406–417, 2002.

[12] E. J. Keogh, K. Chakrabarti, M. J. Pazzani, and S. Mehrotra. Dimensionality reduction for fast similarity search in large time series databases. *Knowl. Inf. Syst.*, 3(3):263–286, 2001.

[13] E. J. Keogh, J. Lin, and A. W.-C. Fu. Hot sax: Efficiently finding the most unusual time series subsequence. In *ICDM*, pages 226–233, 2005.

[14] Y. Li, J. Lin, and T. Oates. Visualizing variable-length time series motifs. In *SDM*, pages 895–906, 2012.

[15] Y. Li, L. H. U, M. L. Yiu, and Z. Gong. Discovering longest-lasting correlation in sequence databases. *PVLDB*, 6(14):1666–1677, 2013.

[16] Y. Li, L. H. U, M. L. Yiu, and Z. Gong. Quick-motif: An efficient and scalable framework for exact motif discovery. *ICDE*, 2015.

[17] T. W. Liao. Clustering of time series data - a survey. *Pattern Recognition*, 38(11):1857–1874, 2005.

[18] J. Lin, E. Keogh, S. Lonardi, and P. Patel. Finding motifs in time series. In *Proc. of 2nd Workshop on Temporal Data Mining*, 2002.

[19] J. Lin, E. J. Keogh, L. Wei, and S. Lonardi. Experiencing sax: a novel symbolic representation of time series. *Data Min. Knowl. Discov.*, 15(2):107–144, 2007.

[20] A. Mueen. Enumeration of time series motifs of all lengths. In *ICDM*, 2013.

[21] A. Mueen and E. J. Keogh. Online discovery and maintenance of time series motifs. In *KDD*, pages 1089–1098, 2010.

[22] A. Mueen, E. J. Keogh, and N. B. Shamlo. Finding time series motifs in disk-resident data. In *ICDM*, pages 367–376, 2009.

[23] A. Mueen, E. J. Keogh, and N. Young. Logical-shapelets: an expressive primitive for time series classification. In *KDD*, pages 1154–1162, 2011.

[24] A. Mueen, E. J. Keogh, Q. Zhu, S. Cash, and M. B. Westover. Exact discovery of time series motifs. In *SDM*, pages 473–484, 2009.

[25] P. Papapetrou, V. Athitsos, M. Potamias, G. Kollios, and D. Gunopulos. Embedding-based subsequence matching in time-series databases. *ACM TODS*, 36(3):17, 2011.

[26] T. Rakthanmanon, B. J. L. Campana, A. Mueen, G. E. A. P. A. Batista, M. B. Westover, Q. Zhu, J. Zakaria, and E. J. Keogh. Searching and mining trillions of time series subsequences under dynamic time warping. In *KDD*, pages 262–270, 2012.

[27] T. Rakthanmanon, E. J. Keogh, S. Lonardi, and S. Evans. Time series epenthesis: Clustering time series streams requires ignoring some data. In *ICDM*, pages 547–556, 2011.

[28] D. Sart, A. Mueen, W. A. Najjar, E. J. Keogh, and V. Niennattrakul. Accelerating dynamic time warping subsequence search with gpus and fpgas. In *ICDM 2010, The 10th IEEE International Conference on Data Mining, Sydney, Australia, 14-17 December 2010*, pages 1001–1006, 2010.

[29] B.-K. Yi and C. Faloutsos. Fast time sequence indexing for arbitrary lp norms. In *VLDB*, pages 385–394, 2000.

[30] T. Yoshiki, I. Kazuhisa, and U. Kuniaki. Discovery of time-series motif from multi-dimensional data based on mdl principle. *Machine Learning*, 58(2-3):269–300, 2005.