Location-Awareness in Time Series Compression

Xu Teng[1]*, Andreas Züfle[2], Goce Trajcevski[1]**, and Diego Klabjan[3]

[1] Dept. of Electrical and Computer Engineering
Iowa State University
Ames, IA USA
xuteng,gocet25@iastate.edu
[2] Dept. of Geography and Geoinformation Science
George Mason University
Fairfax, VA USA
azufle@gmu.edu
[3] Dept. of Industrial Engineering
Northwestern University
Evanston, IL USA
d-klabjan@northwestern.edu

Abstract. We present our initial findings regarding the problem of the impact that time series compression may have on similarity-queries, in the settings in which the elements of the dataset are accompanied with additional contexts. Broadly, the main objective of any data compression approach is to provide a more compact (i.e., smaller size) representation of a given original dataset. However, as has been observed in the large body of works on compression of spatial data, applying a particular algorithm "blindly" may yield outcomes that defy the intuitive expectations – e.g., distorting certain topological relationships that exist in the "raw" data [7]. In this study, we quantify this distortion by defining a measure of similarity distortion based on Kendall's τ. We evaluate this measure, and the correspondingly achieved compression ratio for the five most commonly used time series compression algorithms and the three most common time series similarity measures. We report some of our observations here, along with the discussion of the possible broader impacts and the challenges that we plan to address in the future.

1 Introduction and Motivation

Modern advances in sensing technologies – e.g., weather stations, satellite imagery, ground and aerial LIDAR, weather radar, and citizen-supplied observation – have enabled representing the physical world with high resolution and fidelity. The trend of *Next Generation Sensor Networks and Environmental Science* [9] aims at integrating various data sources (e.g., offered by the state-of-the-art

* Research supported by NSF grant III 1823267.

** Research supported by NSF grants III-1823279 and CNS-1823267, and ONR grant N00014-14-1-0215.

GEOS-5 data assimilation system [26]) and make them publicly available. An example of such large scale dataset is the MERRA-2 data, provided by NASA [19] – covering the whole time period of the modern era of remotely sensed data, from 1979 until today, and recording a large variety of environmental parameters, e.g., temperature, humidity and precipitation; on a spatial resolution of 0.5 degrees latitude times 0.67 degrees longitude produced at one-hour intervals. This, in turn, enables access to many Terabytes of historic evolution in time of environmental data.

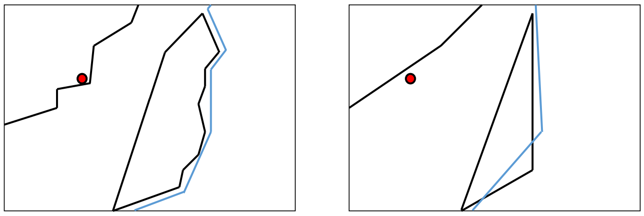


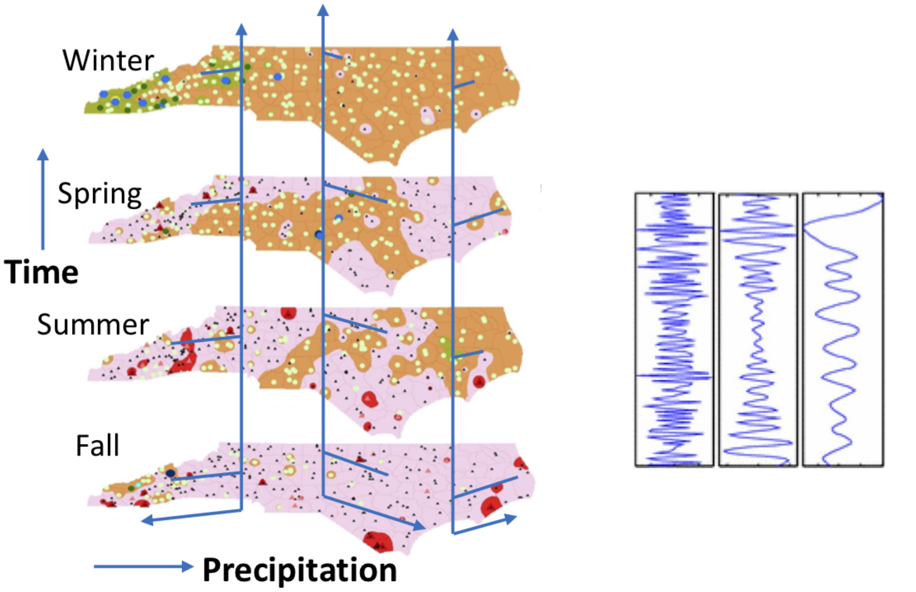
Fig. 1: Simplification and Topological Distortions (based on [7])

Although the focus of this work is on the peculiarities of compressing time series and the interplay with other contexts, to better understand the motivation we briefly turn the attention to compression in spatial data. In the mid 1990s, concurrently with the advances in cartography and maps management [30], the multitude of application domains depending on geographic properties (e.g., distributions) of various phenomena in agriculture, health, demographics, etc. [11], brought about the field of Spatial databases [24]. Most of the compression techniques applied in spatial datasets rely on some kind of a *line simplification* approach, and many variants have also been extensively studied by the Computational Geometry (CG) community [3, 28]. Among of the most popular line simplification approaches is Douglas-Peuker[4] (DP) [6]. However, as demonstrated in [20], applying the DP algorithm to reduce the polylines bounding the polygons in a given subdivision, may often cause topological inconsistencies, as illustrated in Figure 1, in the following sense:

- Boundaries of regions which were not intersecting in the original representation may end up intersecting after the simplification is applied. Similarly, the simplified polylines corresponding to different regions may intersect each other.
- Relative position of point-locations with respect to a boundary or a polyline may change after the simplification is applied – e.g., a city which was on the north bank of the river may end up in its south bank after the polyline representing the river has been simplified.

[4] Around the same time, there were other algorithms developed for polyline simplification, some of which had almost-identical methodologies with the DP algorithm. Most notably [16] which is the reason that sometimes the name Ramer-Douglas-Peuker is used in the literature.

One of the canonical problems in time series is the *similarity search* – i.e., given a collection/database of time series and a particular query-sequence, detect which particular time series is most similar to the querying one, with respect to a given distance function [5]. Since time series databases are large in size, much research has been devoted to speeding up the search process. Among the better known and used paradigms are the ones based on techniques that perform dimensionality reduction on the data, which enables the use of spatial access methods to index the data in the transformed space [14]. Many similarity measurements and distance functions for time series have been introduced in the literature [5] – however, what motivates our work is rooted at the observation that large datasets that are time series by nature, are often tied with other context attributes. Sources of such time series exist in many different domains – such as location-aware social networks [8, 31] and atmospheric and precipitation data [21, 23] (but two examples).



(a) Precipitation Measurements in Discrete Locations

(b) Corresponding Time-series

Fig. 2: Precipitation Time Series

Our key observations are:

O_1: Given the size of such datasets, one would naturally prefer to store the data in a compressed/simplified representation.

O_2: Many queries of interest over such datasets may involve values from > 1 context/domain.

For example, Fig. 2(a) (cf. [21], [17]) illustrates the *spatial* distribution of the measurements of precipitation in discrete locations. However, in each individual location, the collection of the measurements from different time-instants actually form a time series – as illustrated in Fig 2(b) which shows the detailed corresponding precipitation time series. In the spirit of O_1 and O_2 above, consider the following query:

Q_1: *Which location in the continental US has the most similar distribution of monthly precipitation with Ames?*

The main motivation for this work is to investigate the impact of different compression approaches on variability of the answer(s) to Q_1 above. While in the case of Q_1 the additional context is the *location*, we postulate that other queries pertaining to time series with additional contexts may suffer from distortion of their answers. Such distortions, in turn, may affect the choice of a particular compression algorithm to be used – e.g., as part of materializing the data in dimensions-hierarchy of warehouses [27]. In this work, we report our initial findings in this realm.

In the rest of the paper, Section 2 defines the problem settings and Section 3 reviews the compression approaches and respective measures. In Section 4 we discuss in detail our observations to date, and in Section 5 we summarize and outline directions for future work.

2 Problem Definition

In broadest terms, *data compression* can be perceived as a science or an art – or a mix of both – aiming at development of efficient methodologies for a compact representation of information [10, 22]. Information needs a representation – be it a plain text file, numeric descriptors of images/video, social networks, etc. – and one can rely on properties of structure, semantics, or other statistically-valid features of that representation when developing the methodologies for making the underlying representation more compact. Speaking a tad more formally, *data compression* can be defined as any methodology that can take a dataset D with a size β bits as an input, and produce a dataset D' as a representation of D and having a size β' bits, where $\beta' < \beta$ (hopefully, $\beta' << \beta$).

To measure the capability of a data compression algorithm to reduce the size of a dataset, in this work we simply rely on the typical measure – the compression ratio [10].

Definition 1 (Compression Ratio). *Let D be a dataset represented by β_D bits. Let $\mathcal{C}$ be a compression function, which maps D to a compressed dataset $\mathcal{C}(D)$ represented by $\beta_{\mathcal{C}(D)}$ bits. We define the compression ratio of $\mathcal{C}$ on D as:*

$$\mathcal{R}_{\mathcal{C}}(D) = \frac{\beta_D}{\beta_{\mathcal{C}(D)}}.$$

We note that the representation size β_D is not necessarily equal to the entropy $E(D)$ of D [18]: The entropy of D is the smallest possible number of bits required to represent D. Thus, it must hold that $E(D) \leq \beta_D$. The aim of this study is not to evaluate the information aspects of time series theoretically, but rather, to see the impact of the loss of a particular type of information incurred by compression algorithm on practical queries related to similarity search on time series.

Clearly, one can easily find a compression algorithm that maximizes the compression ratio of Definition 1, by a "brute force" discarding any and all information. However, such an approach would inhibit any meaningful similarity search among the compressed time series, as all of them would be equally-valid candidates for an answer. Thus, the challenge approached in this work is to maximize the compression ratio while maintaining similarity search results as accurate as possible.

To measure how a compression algorithm $\mathcal{C}$ can maintain similarity search results among a set D of time series tied with other context attributes, we compute similarity rankings between all the time series. A similarity ranking, using a query time series $T \in D$, ranks all other time series in $D \setminus T$ by their similarity to T. To quantify the similarity ranking before vs. after the compression, we employ Kendall's rank correlation coefficient τ [12], which measures how many pairs of relative ranking positions are preserved and discordant between the two rankings. Formally,

Definition 2 (Ranking Similarity). *Let D be a set of time series. For a query time series $T \in D$, let $Rank(T, D)$ be the similarity ranking of T to all other time series $T' \in D \setminus T$. Further, let $\mathcal{C}$ be a compression algorithm, let $\mathcal{C}(D)$ denote the compressed representation of D, and let $Rank(T, \mathcal{C}(D))$ denote the similarity ranking of T after the compression. Then, we describe the similarity of these two rankings as:*

$$\tau(Rank(T, D), Rank(T, \mathcal{C}(D))) = \sum_{T_i, T_j \in D, i \leq j} \frac{I(conc(T_i, T_j)) - I(disc(T_i, T_j))}{(|D|^2 - |D|)/2},$$

where either $(I(conc(T_i, T_j)))$ or $(I(disc(T_i, T_j)))$ is an indicator function that returns 1 if time series T_i and T_j are concordant or discordant in both rankings (that is, if the relative ranking order between T_i and T_j is maintained or not in both rankings) and 0 otherwise.

As an example, consider a case where we have four time series $T_1, ..., T_4$, and assume that the similarity ranking of T_1 is (T_2, T_3, T_4), implying that T_2 is most similar to T_1, while T_4 is the least similar one. Further, assume that after compression, the ranking becomes (T_2, T_4, T_3). In this case, the relative order between T_2 and T_3 is preserved, as is the relative order between T_2 and T_4. The only "discordant" order is between T_3 and T_4, yielding $\tau((T_2, T_3, T_4), (T_2, T_4, T_3)) = \frac{1}{3}$.

To quantify the overall information maintained between all of the time series, we compute the average τ score of all time series in D.

Definition 3 (Average Ranking Similarity). *Let D be a set of time series and let $\mathcal{C}(D)$ denote the compressed representation of D. We define the average* *ranking similarity between D and $\mathcal{C}(D)$ as*

$$\tau(D, \mathcal{C}(D)) = \frac{\sum_{T \in D} \tau(Rank(T, D), Rank(T, \mathcal{C}(D)))}{|D|}.$$

We reiterate that our goal is to evaluate how different compression algorithms $\mathcal{C}$ affect the balance between compression ratio (Definition 1) and average ranking similarity (Definition 3).

3 Compressions and Distances

For self-containment, we now briefly survey the compression techniques and distance measures used in this study.

3.1 Compression Approaches

We have used two broad categories of compression techniques, as described in detail in the sequel.

Dimensionality Reduction Instead of being viewed as a collection of n time-instant phenomenons, a time series, $\{t_1, t_2, ..., t_n\}$, can be considered as a point in n-dimensional space. Dimensionality reduction approaches focus on reducing the dimensionality – from n in the "native", to m ($m < n$) in the lower dimensional space – while minimizing the loss of explained variance. We use two representative techniques:

• *Discrete Fourier Transform*:

The key idea of Discrete Fourier Transform (DFT) [2] is based on the observation that that any n-length time series can be represented in the frequency domain with n sine and cosine waves, that can be used to reconstruct the original time series. The compression stems from the observation that the waves with low amplitudes can be neglected without losing too much valuable information.

• *Piecewise Aggregate Approximation*:

The basic concept behind the Piecewise Aggregate Approximation (PAA) [14] is dividing the original time series into N equally sized windows, where (N is the desired dimensionality of transformed space. Then, each window/frame is represented by the mean value of all the data within that particular frame. The formula used for performing PAA on an n-dimensional time series and transforming it into the N-dimensional space is shown in Equation 1:

$$\overline{t_i} = \frac{N}{n} \sum_{j=\frac{N}{n}(i-1)+1}^{\frac{N}{n}i} t_j, \quad i = 1, 2, ..., N \tag{1}$$

One may observe that a small window-size can achieve a better performance on preserving information, but yields a poor compression ratio – e.g., when the window size is equal to n, the transformed representation is identical to the original time series.

Native-Space Compression Another kind of compression approaches reduces the size of the initial time series in its "native space":

- *(Adapted) Douglas-Peucker Algorithm*:

  Given a sequence of time series and a user-defined tolerance threshold ε, the Douglas-Peucker (DP) [6] algorithm recursively sub-divides the input sequence based on an "anchor". An "anchor" is a point that has a largest distance exceeding ε from the line segment connecting the initiator (first point initially) and the terminus (last point initially). The DP algorithm is traditionally used to compress polylines. To adapt it to time series, we use vertical (instead of perpendicular) distance in this study. Vertical distance between point t_k and line segment(t_i, t_j), $i < k < j$, is defined as $\left|t_k^{'} - t_k\right|$, where $t_k^{'}$ is the intersection of line segment(t_i, t_j) and the line passing t_k and perpendicular to the time-axis.
- *Visvalingam-Whyatt Algorithm*:

  The key aspect of Visvalingam-Whyatt (VW) [29] algorithm is the "effective area", which indicates the surface area of the triangle formed by a point with its two neighbors. For a time series of lentgh n, a total (n-2) triangles can be formed. The main idea behind the VW algorithm is to iteratively drop the middle point of the triangle with the least "effective area" and keep on updating the triangles related to that displaced point until the "effective area" is larger than the user-given parameter ε.
- *(Adapted) Optimal Algorithm*:

  The main idea of optimal algorithm (OPT) [4] is to consider two directions (*forward* and *backward*), for each point of a time series. For instance, (t_{i+1}, t_{i+2}, ..., t_n) is *forward* for t_i, and (t_{i-1}, t_{i-2}, ..., t_1) is *backward*. The i-th ($1 \leq i \leq n$) pass of the algorithm draws circles with radius ε, centered at each the *forwards* and *backward* points of t_i – denoted $Circle_{i+1}$, $Circle_{i+2}$, ..., $Circle_n$ and $Circle_{i-1}$, $Circle_{i-2}$, ..., $Circle_1$. Take *forward* chain as instance. While touching a new point, t_k, $i < k \leq n$, let U_k and L_k indicate the upper and the lower ray emanating from t_i, passing through the top and bottom point of $Circle_k$ - in a sense, defining a wedge pertaining to t_k and with the apex at t_i. For as long as the intersection of successive wedges is not empty, nothing needs to be updated except of recording the lowest-upper and highest-lower boundary of the intersection maintained so far. Otherwise, denote t_k as the *event point* which generates an empty intersection. We keep t_i and t_{k-1} into the result and repeat the procedure from the event point t_{k-1} forwards. Similarly for the *backward* chain of t_i.

3.2 Distance Measures

Existing literature has identified many scenarios where similarity cannot be simply evaluated by any single distance function [5]. Thus, for validity, we used three measurements in this work, as described next.

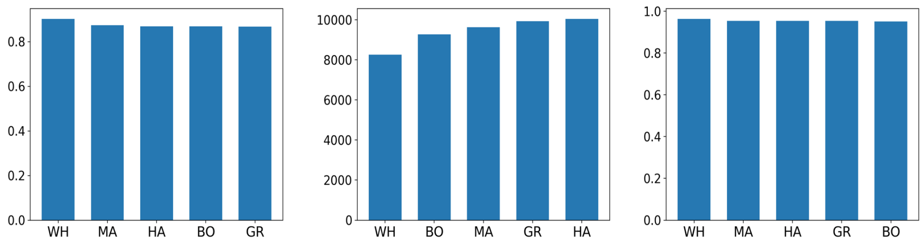
Pearson Correlation Coefficient The Pearson product-moment correlation coefficient [15] (denoted r) is a widely used lock-step measure for relationship. By Cauchy-Schwartz inequality, the range of r is established to the interval [-1, +1], where +1 denotes total positive linear correlation, 0 is no linear correlation, and -1 indicates negative linear correlation.

Dynamic Time Warping Dynamic Time Warping (DTW) [13] is an elastic similarity measure between two temporal sequences. In general, it focuses on calculating an optimal match between two given time series that may vary in speed/frequency. Unlike lock-step methods, DTW alignment may match a point from one sequence to one or more points of another sequence.

Cosine Similarity Cosine similarity [25] aims at evaluating the orientation difference between two time series, and is independent of the magnitude of the samples. If two sequences are with a same orientation, their cosine similarity will be 1; and if their orientation difference is 90°, then their similarity will be zero.

4 Experimental Observations

In this section, we present the experimental evaluations of the approaches discussed in Section 3 in terms of compression rate and average ranking similarity. Our data sets are obtained from the University Corporation for Atmospheric Research (UCAR) and the National Center for Atmospheric Research (NCAR) at the Global Precipitation Climatology Centre [1].



(a) Pearson Correlation Coefficient

(b) Dynamic Time Warping

(c) Cosine Similarity

Fig. 3: Location-Based Similarity Scores

Recall the motivational question stated in Section 1: *Which location in the continental US has most similar distribution of monthly precipitation with Ames?* In this spirit, we extract 50 years worth of monthly precipitation data for Ames and other 500 land areas in the United States. Fig. 3 shows the top five locations having highest similarity with Ames measured by Pearson Correlation

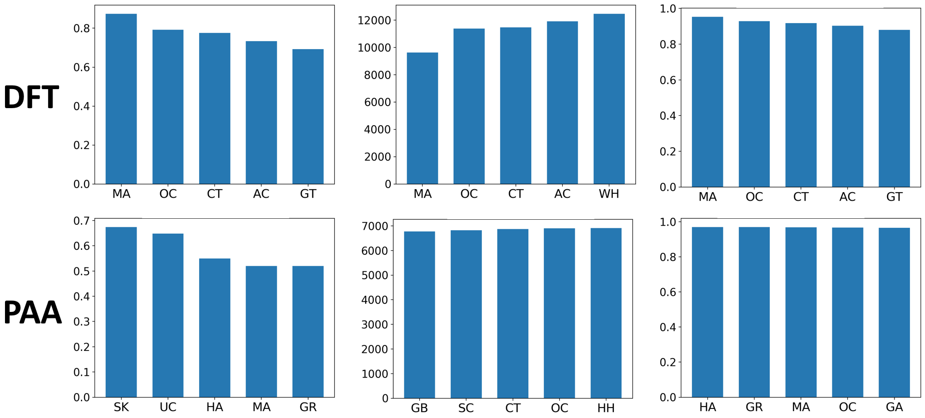
Location	Abbreviation	Location	Abbreviation
Wheeler, IA	WH	Massena, IA	MA
Hartford, IA	HA	Boyer, IA	BO
Grant, IA	GR	Garfield, IA	GA
South Kidder, ND	SK	Union County, NM	UC
Otter Creek, IA	OC	Courtland Township, MN	CT
Anoka County, MN	AC	Grant Township, SD	GT
Grantsburg, WI	GB	Scott, WI	SC
Hazelhurst, WI	HH		

Table 1: Locations used in the reported experiments

Coefficient, DTW and Cosine Similarity, respectively. The horizontal axis shows the abbreviation of each locations, and the corresponding full name can be found at Table 1. The vertical axis shows the similarity score of each locations. As discussed in Sec. 3, higher score means better performance for Pearson Correlation Coefficient and Cosine Similarity, and DTW pursues lower distance. We can figure out that the five locations listed in Fig. 3(a), Fig. 3(b) and Fig. 3(c) are same though the ranking has some differences.

Fig. 4 states the effect of two Dimensionality Reduction methods on ranking. 0.7 in terms of multiples of the maximum value of each time series is defined as *error tolerances*. We can discover the only half of the locations have no difference with Fig. 3.

Fig. 5 illustrates the influence of three different Native-Space compression approaches mentioned in Section 3 on ranking of similarity. For DP and VW approaches, the *error tolerances* are set to be 5. For the OPT algorithm, the values of tolerance are set to be the half of those of DP and VM algorithms.

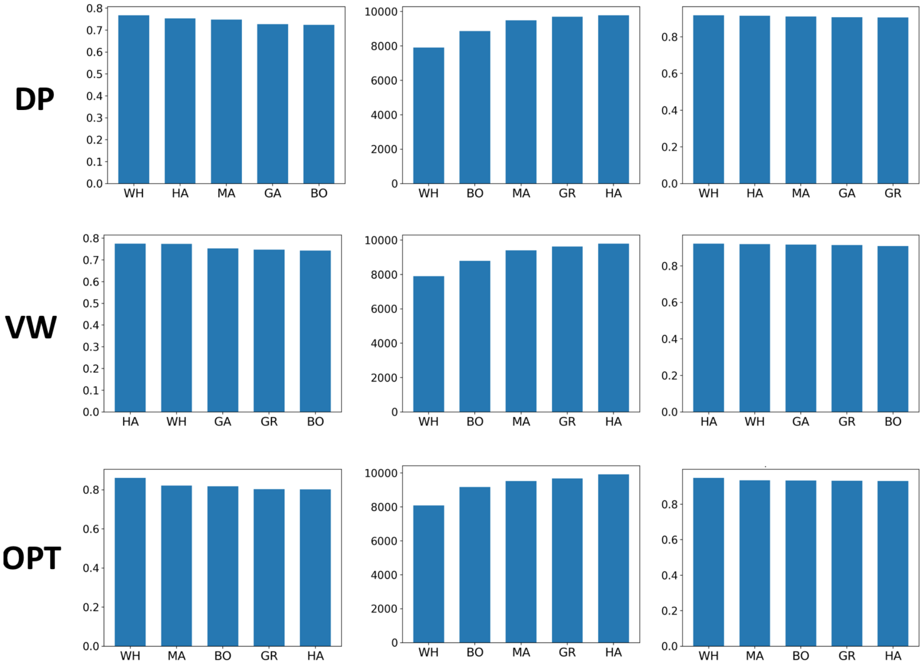


(a) Pearson Correlation Coefficient

(b) Dynamic Time Warping

(c) Cosine Similarity

Fig. 4: Dimensionality Reduction Compression



(a) Pearson Correlation Coefficient

(b) Dynamic Time Warping

(c) Cosine Similarity

Fig. 5: Native-Space Compression

As can be seen from results, though the compression do influence the ranking of top five locations, all the locations in the result are the same as the ranking of ground truth except of Grant, IA. And performance is similar to Fig. 3 and better than Fig. 4 while checking the similarity score

To evaluate the influence of different compression approaches on time series in a more general way, we randomly samples 100 land areas in the United States, and fetched 50 years of monthly precipitation data for each location in our work. Similarly to the above experiment, for each selected area, we can get three different rankings of similarity across the rest of 99 locations measured by Pearson Correlation Coefficient, DTW and Cosine Similarity. As mentioned in Section 2, the goal is to evaluate the influence of different compression approaches on compression ratio and impacts on the average ranking similarity. In order to perform such comparison, we set the single parameter *error tolerance* for different algorithms. For DFT and PAA approaches, the *error tolerances* are set to be 0.7, 0.75, 0.8, 0.85, 0.9, 1 in terms of multiples of the maximum value of each dataset. For DP and VW approaches, the *error tolerances* are 5, 10, 15, 20, 25, 30. Lastly, for the OPT algorithm, the values of tolerance are set to be the half of those of DP and VM algorithms.

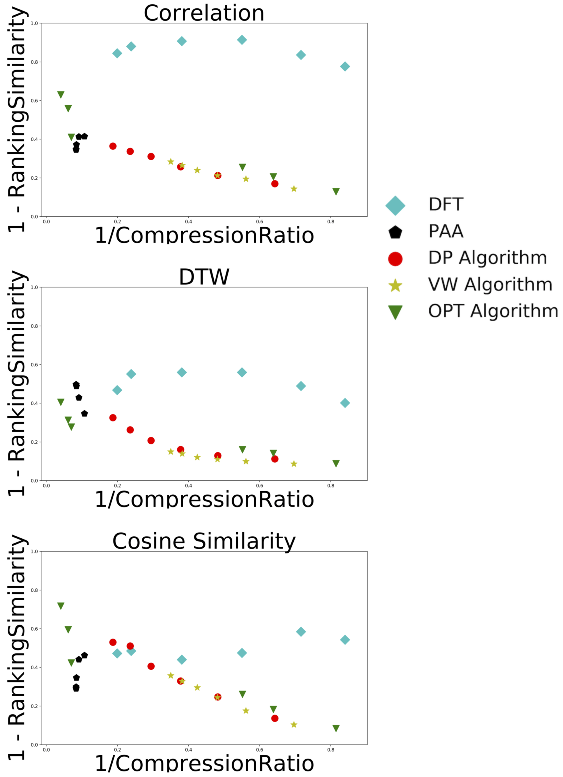


Fig. 6: Global Similarity Distortions

Fig. 6 illustrates one more of our experimental observations that we now discuss. Note that, in a sense, we have transformed the coordinates and the respective values are: — the x-axis for each of the three graphs represents $1/\mathcal{R}_{\mathcal{C}}(D)$; — the y-axis in each of the three graphs represents $1 - \tau(D, \mathcal{C}(D))$. This transformation was made to ensure that a "good" approach is close to the origin of the chart. For instance, a point at the $(0, 0)$ origin would correspond to a perfect τ of 1, and a perfect compression rate of ∞.

Firstly, we observe that there exists no single approach that clearly dominates all the other approaches, in terms of both τ-score and compression rate. However, we also observe that DFT performs rather poorly in comparison with the other measures. Thus, we conclude that even when achieving a fairly low compression rate, DFT looses most of information required to maintain the original similarity ranking. This loss is not compensated by additional frequency features, as these low amplitude features mostly incur additional noise.

Secondly, we observe that PAA achieves a relatively high compression rate, but the averaging of consecutive time stamps yields a high loss of information that drops τ to roughly 60% for all the other applied distance functions.

In contrast, we observe that all three native-space compression algorithms have comparable performances. We note that the OPT algorithm generally achieves worse results, and is dominated by DP and VW – which, in part, may be a consequence of setting a lower error-threshold. We observe minor difference between DP and VW, expect when DTW is used a distance measure: namely, VW is able to maintain more ranking for the similarity-based semantics in the DTW space.

We note that, for reproducibility, the source code for all the implementations used in our experiments, along with the corresponding dataset, is publicly available[5].

5 Summary and Future Directions

Satellites and other sensory devices have enabled a generation of extremely large environmental time series datasets. Ultimately, this data has the potential to transform our understanding of the world for a plethora of applications of societal relevance, such as meteorology, agriculture, urban development, traffic management, etc. However, this understanding is hindered by the overwhelming deluge of O(Petabytes) of such data. To reduce this data, the state-of-the-art offers many time series compression algorithms.

In this study, we experimentally evaluated the trade-off between the data reduction and the loss of semantics when an additional context – location in this work – is associated with each time series. Rather than measuring the theoretical loss of entropy, we measured how the incurred distortion changes similarity search results on environmental time series, using precipitation time series as a case-study.

Our main experimental finding is that dimensionality reducing methods, such as Discrete Fourier Transform and Piecewise Aggregate Approximation incur a high loss of similarity between compressed time series, relative to the original ones. In contrast, native space compression algorithms obtain similar compression rates, but maintain much more of the similarity information between time series. In particular, the Visvalingam-Whyatt algorithm and the Douglas-Peucker algorithm yield the best trade-off. Moreover, when Dynamic Time Warping is used as a similarity metric, Visvalingam-Whyatt has a significant advantage over Douglas-Peucker.

Our main objectives for the future are: (1) extend this study to include more compression algorithms, and include different types of environmental time series other than precipitation; and (2) investigate the impact of compression on semantics of other context attributes – e.g., in addition to location, exploit the (joint) impact on other social networks features; (3) evaluate the potential impacts of running time of the algorithms, especially in the sense of updating the datasets from newly available observations.

Acknowledgments: We thank Praxxal Patel and Yash Thesia for their help in finalizing part of the experiments.

References

1. GPCC: GLOBAL PRECIPITATION CLIMATOLOGY CENTRE. https://climatedataguide.ucar.edu/climate-data/gpcc-global-precipitation-climatology-centre.

[5] https://github.com/XuTengNU/ADBIS2018.git

2. Rakesh Agrawal, Christos Faloutsos, and ArunSwami. Efficient similarity search in sequence databases. In *Foundations of Data Organization and Algorithms*, 1993.
3. Gill Barequet, Danny Z. Chen, Ovidiu Daescu, Michael T. Goodrich, and Jack Snoeyink. Efficiently approximating polygonal paths in three and higher dimensions. *Algorithmica*, 33(2):150–167, 2002.
4. W. S. Chan and Francis Chin. Approximation of polygonal curves with minimum number of line segments. *International Journal of Computational Geometry and Applications*, 6, 1992.
5. Hui Ding, Goce Trajcevski, Peter Scheuermann, Xiaoyue Wang, and Eamonn J. Keogh. Querying and mining of time series data: experimental comparison of representations and distance measures. *PVLDB*, 1(2), 2008.
6. David Douglas and T. Peuker. Algorithms for the reduction of the number of points required to represent a digitised line or its caricature. *The Canadian Cartographer*, 10(2), 1973.
7. Regina Estkowski and Joseph S. B. Mitchell. Simplifying a polygonal subdivision while keeping it simple. In *Symposium on Computational Geometry*, 2001.
8. Laura Ferrari, Alberto Rosi, Marco Mamei, and Franco Zambonelli. Extracting urban patterns from location-based social networks. In *Proc. of the 3rd ACM SIGSPATIAL Int.l Workshop on Location-Based Social Networks*, LBSN '11, 2011.
9. Tony Hey, Stewart Tansley, Kristin M Tolle, et al. *The fourth paradigm: data-intensive scientific discovery*, volume 1. Microsoft research Redmond, WA, 2009.
10. D. Hirschberg and D. A. Lelewer. Data compression. *Computing Surveys*, 19(3), 1987.
11. Jeanne X. Kasperson, Roger E. Kasperson, and B. L. Turner II. *Regions at Risk: Comparisons of Threatened Environments*. Inited Nations University Press, 1995.
12. Maurice G Kendall. A new measure of rank correlation. *Biometrika*, 30:81–93, 1938.
13. Eamonn Keogh and Chotirat Ann Ratanamahatana. Exact indexing of dynamic time warping. *Knowledge and information systems*, 7:358–386, 2005.
14. Eamonn J. Keogh, Kaushik Chakrabarti, Michael J. Pazzani, and Sharad Mehrotra. Dimensionality reduction for fast similarity search in large time series databases. *Knowl. Inf. Syst.*, 3(3), 2001.
15. Karl Pearson. Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, 1895.
16. Urs Ramer. An iterative procedure for the polygonal approximation of plane curves. In *Computer Graphics and Image Processing*, page 244256, 1972.
17. Janga Reddy and Adarsh Sankaran. Time–frequency characterization of subdivisional scale seasonal rainfall in india using the hilbert–huang transform. *Stochastic Environmental Research and Risk Assessment*, 2016.
18. Alfréd Rényi. On measures of entropy and information. Technical report, HUNGARIAN ACADEMY OF SCIENCES Budapest Hungary, 1961.
19. Michele M Rienecker, Max J Suarez, Ronald Gelaro, Ricardo Todling, Julio Bacmeister, Emily Liu, Michael G Bosilovich, Siegfried D Schubert, Lawrence Takacs, Gi-Kong Kim, et al. Merra: Nasas modern-era retrospective analysis for research and applications. *Journal of climate*, 24(14):3624–3648, 2011.
20. Alan Saalfeld. Topologically consistent line simplification with the douglas-peucker algorithm. *Cartography and Geographic Information Science*, 26(1):718, 1999.
21. Mohammad Sayemuzzaman and Manoj K. Jha. Seasonal and annual precipitation time series trend analysis in north carolina, united states. *Atmospheric Research*, 137, 2014.

22. Khalid Sayood. *Introduction to Data Compression*. Morgan Kauffman, 1996.
23. Chandra Shekhar Sharma, Sudhindra N. Panda, Rudra P. Pradhan, Amanpreet Singh, and Akira Kawamura. Precipitation and temperature changes in eastern india by multiple trend detection methods. *Atmospheric Research*, 180, 2016.
24. Shashi Shekhar and Sanjay Chawla. *Spatial Databases: A Tour*. Prentice Hall, 2003.
25. Michael Steinbach, George Karypis, Vipin Kumar, et al. A comparison of document clustering techniques. In *KDD workshop on text mining*, volume 400, pages 525–526, 2000.
26. Max J Suarez, MM Rienecker, R Todling, J Bacmeister, L Takacs, HC Liu, W Gu, M Sienkiewicz, RD Koster, R Gelaro, et al. The geos-5 data assimilation system-documentation of versions 5.0. 1, 5.1. 0, and 5.2. 0. 2008.
27. Alejandro A. Vaisman and Esteban Zimányi. *Data Warehouse Systems - Design and Implementation*. Data-Centric Systems and Applications. Springer, 2014.
28. Marc J. van Kreveld and Jun Luo. The definition and computation of trajectory and subtrajectory similarity. In *GIS*, 2007.
29. M. Visvalingam and J. D. Whyatt. Line generalisation by repeated elimination of points. *The Cartographic Journal*, 30(1), 1993.
30. Robert Weibel. Generalization of spatial data: Principles and selected algorithms. In *Algorithmic Foundations of Geographic Information Systems*. LNCS Springer Verlag, 1997.
31. Guolei Yang and Andreas Züfle. Spatio-temporal prediction of social connections. In *Proceedings of the Fourth International ACM Workshop on Managing and Mining Enriched Geo-Spatial Data, Chicago, IL, USA, May 14, 2017*, 2017.