MPdist Haskell Implementation: A distance metric for Time Series

Asif Mallik (am5086)

November 2021

1 MPdist

MPdist (Matrix Profile distance) is a distance metric that captures the similarity between different time series [1]. Among its various features are its ability to be able to compare time series of varying lengths, being robust to anomalies, missing data and other issues common to time series data, very efficient to compute. In brief, it compares the similarity of subsequences of their time series in order to determine whether they share motifs or not and hence whether they are similar. The definition of MPdist makes use of a recently developed concept in time series data mining known as Matrix Profile.

2 Matrix Profile

Defining a matrix profile rigorously would require at least five prerequisite definitions. So, for interest of brevity, we define it informally. A matrix profile with respect to time series A and B of window size m, is a list representing the minimal euclidean distance of each m-length subsequence of A to any m-length subsequence of B. As a result, note that the matrix profile seen as an operation is non-commutative. The "matrix" in matrix profile refers to the distance matrix one would have to compute if they were to naively try to compute the matrix profile. A join similarity matrix is an extension of this concept that makes it more symmetric by concatenating the matrix profile of A with respect to B. Yeh et al. outlines the algorithm for computing the matrix profile for any given pair of time series [2].

MPdist between time series A and B can then be defined as the nth percentile of the join similarity matrix profile of time series A and B. n here is a parameter to MPdist, so more accurately, MPdist is a family of distance metrics parameterized by window size m and similarity percentile n. Given a matrix profile, it is quite trivial and fast to compute the MPdist. Thus, the bulk of the project will be spent in implementing the algorithm for computing the matrix profile.

3 STOMP and STAMP algorithm

The STAMP algorithm works by iterating through the indices of one of the time series to select as a starting index for the subsequence. Then, in each iteration, it computes a series of sliding dot products with the selected subsequence as the starting point so as to minimize computation. After each iteration, the each entry at the matrix profile if the newly computed distances are smaller. The novelty in this algorithm is the use of fast fourier transform and the sliding dot product trick in order to avoid recomputing. Overall, its complexity is $O(n^2 \log n)$ with n being the length of the time series. STOMP is even faster with a runtime of $O(n^2)$ by utilizing an additional small inner loop. At this time, I am not exactly sure how STOMP works.

Parallelizing the computation of both STAMP and STOMP involves simply doing the iterations in parallel instead of sequentially. [3]

4 Project Deliverable

In the project, I will attempt to do the following:

- Implement the STAMP algorithm as a sequential algorithm
- Parallelize the STAMP implementation
- Implement the algorithm for computing MPdist
- Implement the STOMP algorithm and parallelize it

I plan to at least finish the first three tasks and will leave the last one as a stretch goal if I am able to finish the rest effortlessly.

References

[1] Shaghayegh Gharghabi, Shima Imani, Anthony Bagnall, Amirali Darvishzadeh, and Eamonn Keogh. Matrix profile xii: Mpdist: a novel time series distance measure to allow data mining in more challenging scenarios. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 965–970. IEEE, 2018.

[2] Chin-Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Diego Furtado Silva, Abdullah Mueen, and Eamonn Keogh. 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. Ieee, 2016.

[3] Yan Zhu, Zachary Zimmerman, Nader Shakibay Senobari, Chin-Chia Michael Yeh, Gareth Funning, Abdullah Mueen, Philip Brisk, and Eamonn Keogh. Matrix profile ii: Exploiting a novel algorithm and gpus

to break the one hundred million barrier for time series motifs and joins. In *2016 IEEE 16th international conference on data mining (ICDM)*, pages 739–748. IEEE, 2016.