Matrix Profile – Motif discovery application

Name: Nguyen Quoc Hung

I. Executive summary

In recent years, the world has taken a big step in data analysis, especially time series data analysis. As one of the most popular applications, companies try to analyze the data sales throughout years to identify the purchase behavior of the customers. Although there are many factors that affect the number of sales in a company, finding out a trend in a specific time of a year based on the number of sales per day from years to years is really a good solution for the strategic team to propose a new strategy to increase the amount of the profits.

Besides, in the economics, data analysis is also considered as a solution for many different areas. For example, Musical Plagiarism has always been a problem for the development of music history. In fact, there are many famous songs that receive many rounds of applause from the listener before they realize that it is copied from another song. However, since there are thousands of music songs that are published every day, it is a challenge and causes a lot of time to reach out if it is a new song or a song that comes from another song.

All-pairs-similarity-search has been used to seek all pairs of records in a dataset that meet a similarity threshold, based on some definition of similarity. In this project, I used matrix profile as a definition of similarity to realize the repeating patterns in a real-valued time series data, or in other words, motifs. Motifs are very useful for explanatory and often used as inputs for clustering, classification, etc.

I chose 2 types of data in this project. First, I used a data of estimating Walmart sales published of Kaggle.com which includes the number of sales of various products all days from 2011 to 2016. With this data, I went far beyond the scope by summing the number of products that are sold per day at the exact day of time then using a matrix profile to find out the trends of the customers.

The second data I would like to represent is a combination of 2 songs as a train set and two other songs as a test set and still use the matrix profile to find out which song is offending the Music Plagiarism.

Since I believe that matrix profiles are only helpful to find out the similarity, I did not try to use any statistical learning or deep learning model method. In this project, I just want to represent the advantage when applying the Matrix profile in real data.

For future work, I believe it might be interesting to further investigate and understand how this algorithm can be helpful for model efficiency. Moreover, although using motif discovery is very exact and it is very easy to show the similarity of a similarity pair, in fact, it is computational expenses for using this algorithm. Therefore, in the future, I really want to put this algorithm with other larger data.

II. Introduction

These days, in the data era, every information can become a dataset and can be analyzed for future use. Companies try to analyze the data sales throughout years to identify the purchase behavior of the customers. Although there are many factors that affect the number of sales in a company, finding out a trend in a specific time of a year based on the number of sales per day from years to years is really a good solution for the strategic team to propose a new strategy to increase the amount of the profits.

On the other hand, Musical Plagiarism is also considered as a problem for the development of music history. Since there are thousands of music songs that are published every day, it is a challenge and causes a lot of time to reach out if it is a new song or a song that comes from another song.

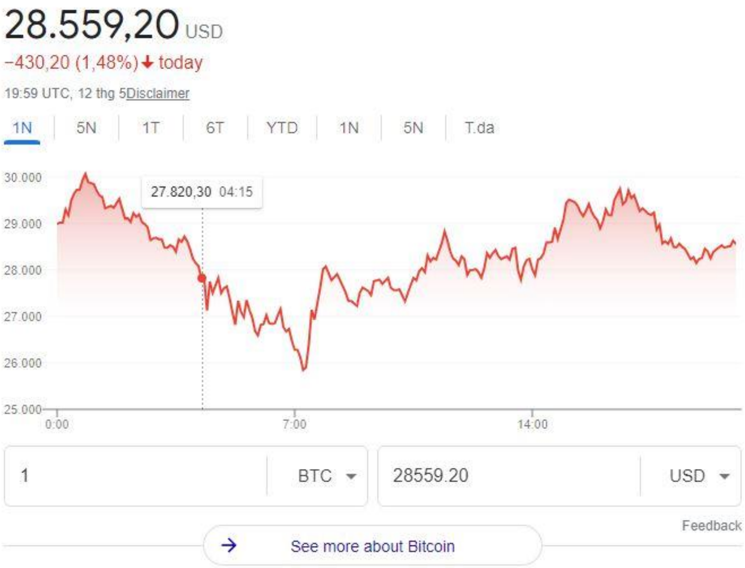
III. The data

The second one: The dataset is a combination of 3 audios, which is "Journey to the West", "Uot my" and "Tay Vuong Nu Quoc".

IV. Basic definition

To clearly understand about this project, I will clarify some basic definition that is used in this project paper:

1. Time series: In statistics, signal processing, econometric and financial mathematics, time series is a series of data points, which is measured at consecutive periods with a defined time frequency. Time series analysis includes methods for analyzing time series data to extract significant statistical properties and characteristics of the data. Time series prediction is the use of a model to predict time events based on the past events that are known in the past. The graph below shows the example of a time series, which is the price of bitcoin at a specific time in a day.



28.559,20 USD

−430,20 (1,48%)↓ today

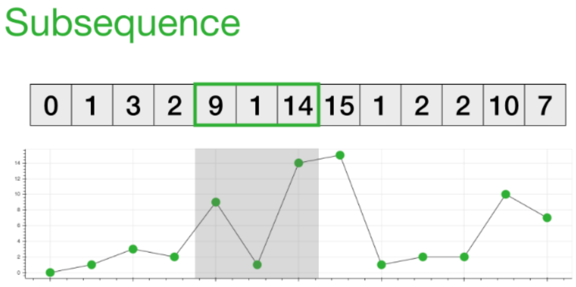
19:59 UTC, 12 thg 5Disclaimer

1 BTC

28559.20 USD

(Source: https://www.google.com/finance/?sa=X&sqi=2&ved=2ahUKEwjv74P44OX3AhXkHjQIHe-BDfwQ6M8CegQIAhAG)

2. Subsequence: A subsequence is a continuous subset of a specific length of values in a time series data.

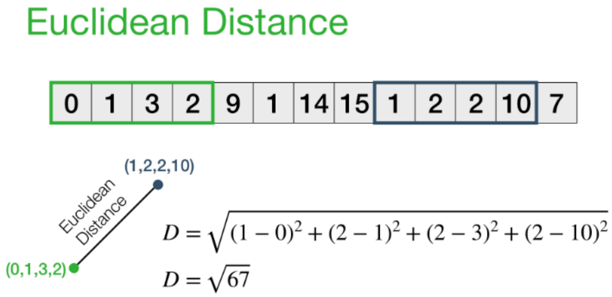


Subsequence

0	1	3	2	9	1	14	15	1	2	2	10	7

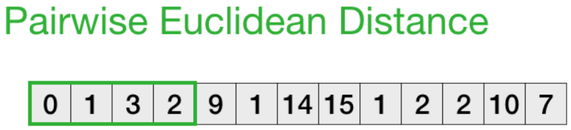
(Source: https://towardsdatascience.com/introduction-to-matrix-profiles)

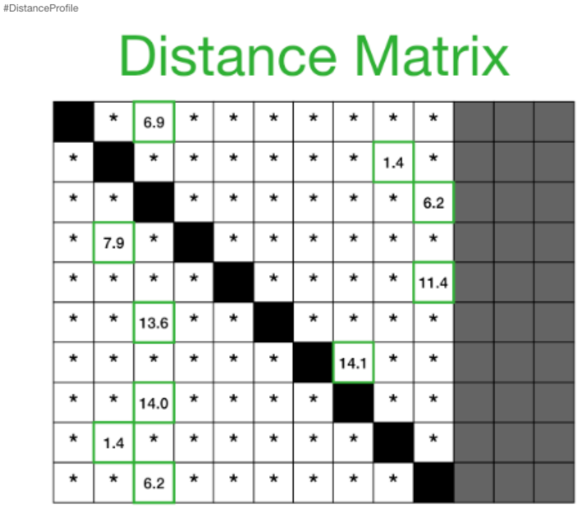
3. Distance Profile: A distance profile D is a matrix combines by all the vectors of the Euclidean distance given by each of the subsequence in a subsequence set in a time series data.



$$D = \sqrt{(1-0)^2 + (2-1)^2 + (2-3)^2 + (2-10)^2}$$

$$D = \sqrt{67}$$





■	*	6.9	*	*	*	*	*	*	*	▨	▨	▨
*	■	*	*	*	*	*	*	1.4	*	▨	▨	▨
*	*	■	*	*	*	*	*	*	6.2	▨	▨	▨
*	7.9	*	■	*	*	*	*	*	*	▨	▨	▨
*	*	*	*	■	*	*	*	*	11.4	▨	▨	▨
*	*	13.6	*	*	■	*	*	*	*	▨	▨	▨
*	*	*	*	*	*	■	14.1	*	*	▨	▨	▨
*	*	14.0	*	*	*	*	■	*	*	▨	▨	▨
*	1.4	*	*	*	*	*	*	■	*	▨	▨	▨
*	*	6.2	*	*	*	*	*	*	■	▨	▨	▨

(Source: https://towardsdatascience.com/introduction-to-matrix-profiles)

4. Matrix profile: A matrix profile of a time series data is a matrix that collects all the smallest Euclidean distance from each vector in a distance profile D
5. Similarity joins (motifs): is the most similar subsequence pair of a time series data.

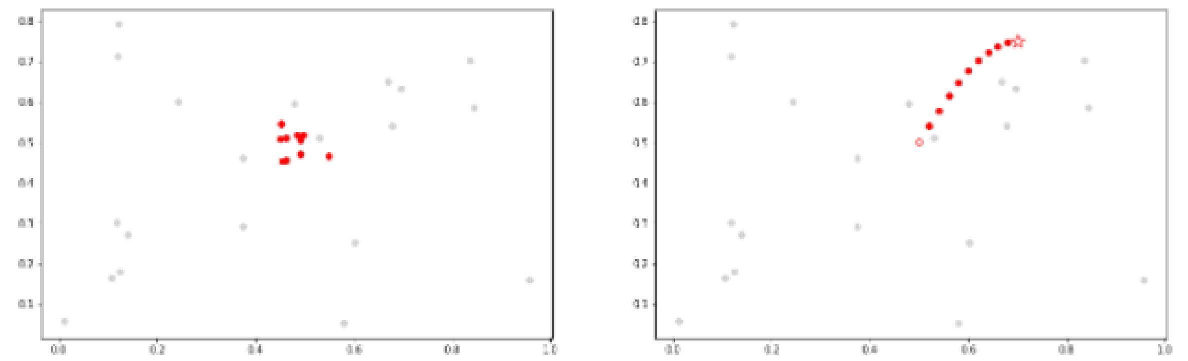
V. Analyzing the data using motifs in matrix profile

There are the best ideas in times series data mining in the last two decades and given the matrix profile, most time series data mining problems are trivial to solve in a few lines of code. In time series analysis, there are two things that always be considered: anomalies and trends. In fact, motifs analyzing methods have many applications such as seeking similarity pairs in a large time series data. Furthermore, using the matrix profile can easily reach out special point, which is low values point and high values point:

Low values mean that the subsequence in the original time series must have at least one relatively similar subsequence elsewhere in the data (motifs).

High values mean that the subsequence in the original time series must be unique in its shape (anomalies).

Another way to explore high-correlation point is using top K-motif method:



These two above graphs represent the top 10-motif method (k=10). The graph on the left shows a cluster of 10 points with the closest distance, when the graph on the right is a set of points in a curved series. This method holds great promise in data mining work. Instead of clustering, miners will directly classify the correlation points.

Since motif discovery is a good method with the advantage of being exact and simple, I try to apply this method to 2 types of data, which I will mention in the following paper.

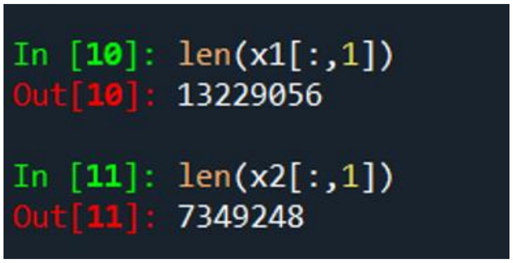
VI. Analyzing Data

Recently, copyright infringement is a common and difficult problem to solve. However, matrix profile motif should be considered as an appropriate method for this problem because the output of this method is to find out the similarity joins, so if we put a specific subsequence based on the copyright rule, it can easily reach out whether two songs have the same subset of music audio or not based on the exact characteristic of the method.

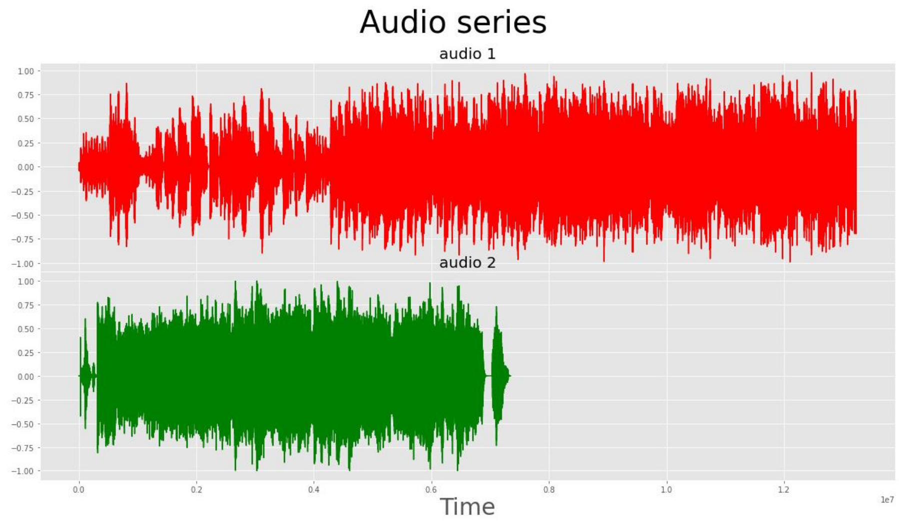
As I mentioned from above, in this project, I just choose two music audios that have nothing in common as a train set and test set.

Since both music audios have been downloaded independently from the internet with different frequencies and volumes, therefore, studentized the frequencies and volumes for these two audios should be considered. In this project, the frequency used is 44Khz. Audio1 file: 28Mb, audio2: 50Mb.

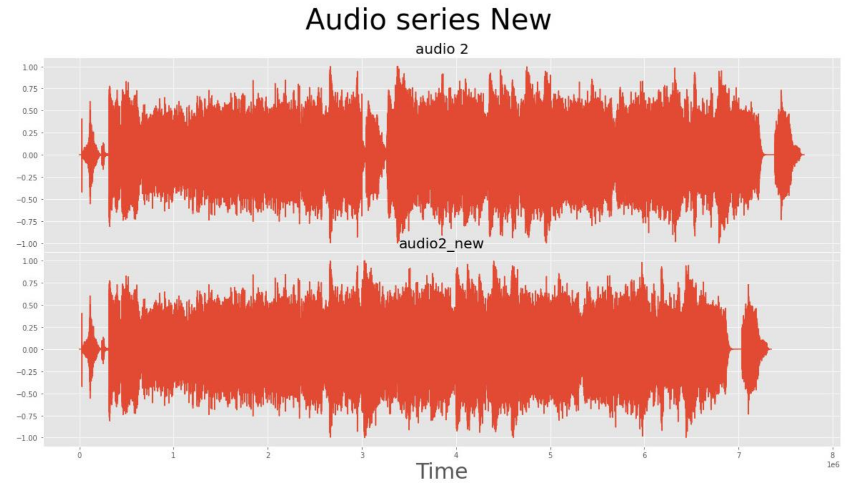
After converting the two audios into an array, we found the length of the first audio is 13,229,056, while the length of the second audio is 7,349,248



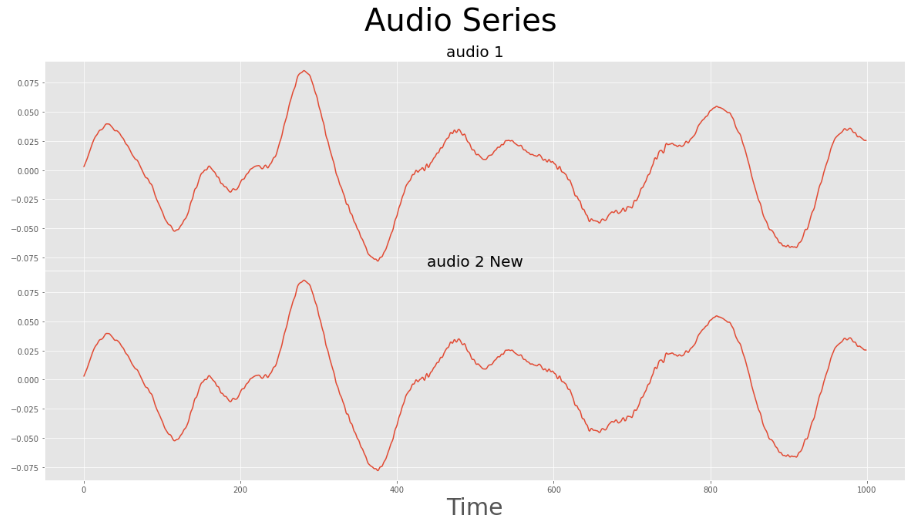
```
In [10]: len(x1[:,1])
Out[10]: 13229056

In [11]: len(x2[:,1])
Out[11]: 7349248
```



The graph above shows the two audios after converting the audio into an array with numbers. My idea is to cut a subset length of the first audio to combine with the second audio and then take the new audio as a test set to compare with the first audio.



The graph above shows a new audio that is fixed. It can be seen that the bottom string, which represents the new audio after combining, is longer than the bottom string from the previous graph, which represents the audio before combining. Moreover, from this graph, it is a challenge to conclude if there is any "copyright part" from the second audio to the first audio or not.

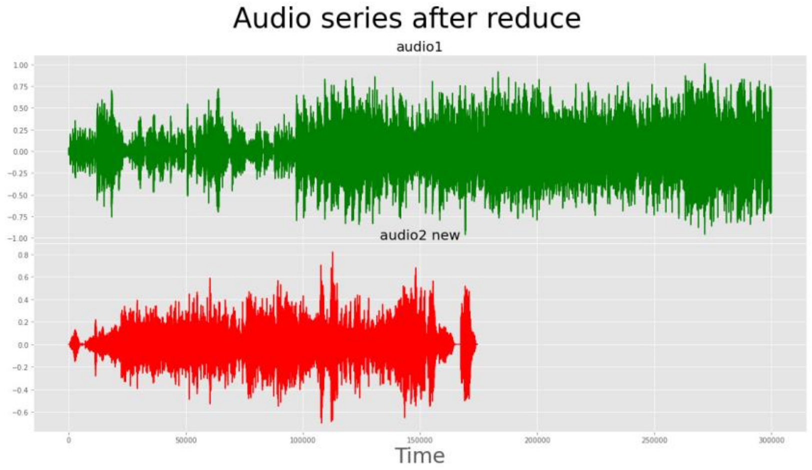


The above graph shows the two time series periods for two audios, which is T1[1650000:2000000] and T2[3000000:3550000]. These two time series periods show exactly the same graph.

However, finding the motifs based on the original data from the audio will cost so much money, since the length of these two songs are almost consist of 20,000,000 numbers, if there exist millions of songs, data storage for the data can reach to myriad Terabits, which lead to the fact that using this method is very slow, even though motif discovery is quite fast compared to other methods.

To solve this problem, we need to use some knowledge of audio sequence analysis, or in other words, we need to use a technique called MCFF, which is to reduce the loudness of the song and still keep the basic structure. After successfully using this algorithm, the audio sequence of each song is only about 150,000 in size, just enough to be used in motif similarity search. This is an important step with preprocessing, we can quickly determine the running time of the program and roughly determine where the motif is.

With 10% of the points used (after using MCFF algorithm to reduce), the running time for identifying the motif will reduce to 2 minutes.

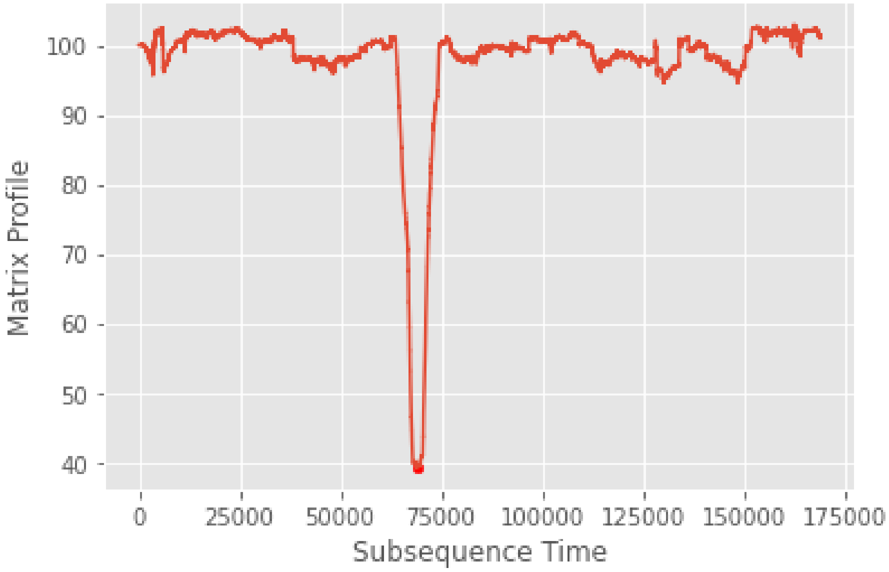


AB-Join motifs discovery:

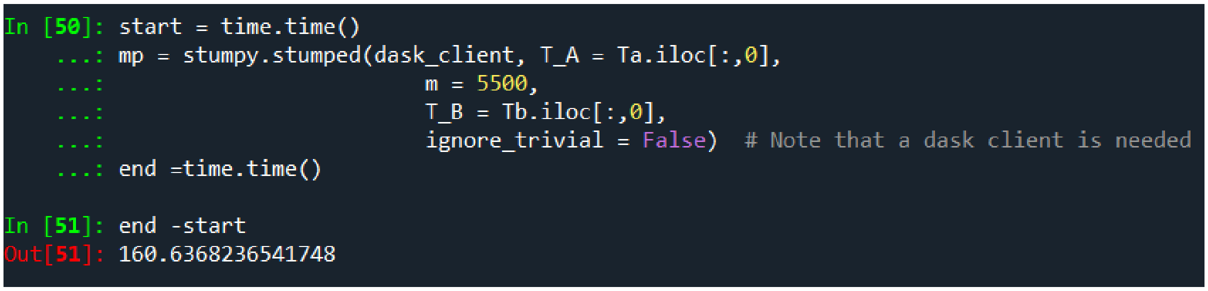
For the prior work, we have studied motif conservation in a single time sequence, when the process of comparison and calculation is simply just taking its own substring to compare with other substrings in the same set of time series T. However, if there are two time series data T_a and T_b we can compare them with each other. In fact, this method is called the AB – join motif method

Basically, AB-Joins compare each $T_{a(i)}$ with each $T_{b(j)}$. For example, assume that T_a has a length of 20 and T_b has a length of 5. If combining these two lengths into 1 and proceed to search for motifs, the number of iterations to calculate with subsequence (5) is 20 x 20 times. On the other hand, using the AB-joins motif method, the number of iterations will reduce to just 20 times. The difference for using this method is that we need to know where to define the motif, which will reduce the computational expenses.

The graph below shows a matrix profile for audio 2 (test set) with audio 1 (training set) by using AB-motifs method. From the graph, there exists an "anomaly" matrix profile point, which is much lower than other points. Based on the result from the graph, we can conclude that motifs may exist.

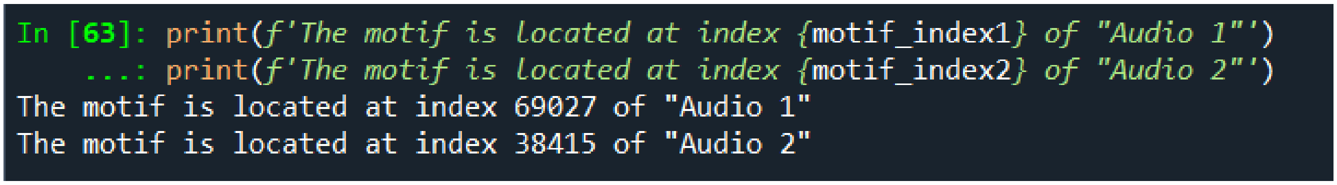


By setting m, which is the subsequence = 5500, which can be converted as 5.5 seconds for the length of the audio, using CPU Intel I3-9100F with 4 cores, the running time is 160.63 seconds, which means that it can be applied in real life for big dataset.

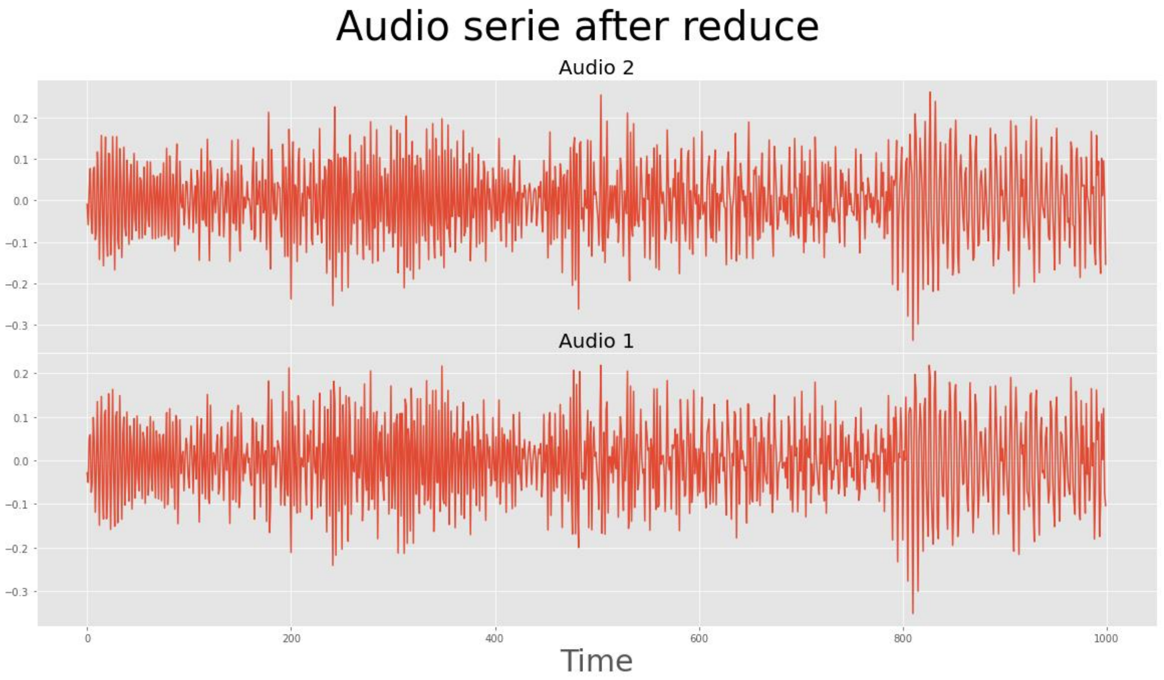


```
In [50]: start = time.time()
    ...: mp = stumpy.stumped(dask_client, T_A = Ta.iloc[:,0],
    ...:                                m = 5500,
    ...:                                T_B = Tb.iloc[:,0],
    ...:                                ignore_trivial = False)  # Note that a dask client is needed
    ...: end =time.time()

In [51]: end -start
Out[51]: 160.6368236541748
```

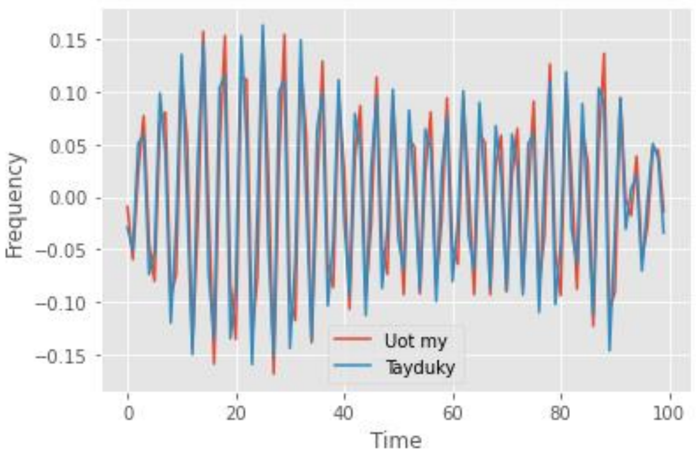
Identifying the motif occurs at position 69027 for audio 1 and 38415 for audio 2. And now I will check whether there is an overlap in these two positions.



```
In [63]: print(f'The motif is located at index {motif_index1} of "Audio 1"')
    ...: print(f'The motif is located at index {motif_index2} of "Audio 2"')
The motif is located at index 69027 of "Audio 1"
The motif is located at index 38415 of "Audio 2"
```

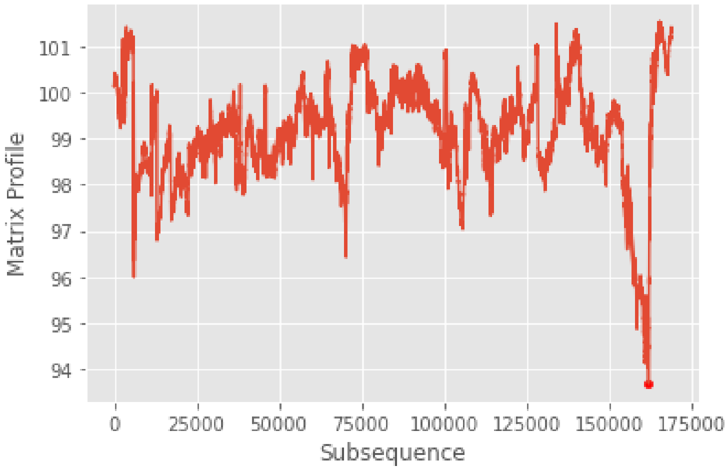


Examine about 1000 times for more details:



Looking at the graph of that point, it can be seen a huge similarity of these two audios (which the same amount of frequency with a little bit different from amplitude), which means that the Matrix Profile motifs have already found out a subset length of audio 2 that is similar with audio1.

Taking another audio, which is completely irrelevant to the existing audios, which is named as audio 3 and I continue using AB-joins motif to compare the similarity with audio 1.



From the graph above, the minimum for the matrix profile is 92, while the mean for others is about 98, while for the previous comparison, the minimum for the matrix profile is 36, which is much smaller than the mean (about 100), therefore, there is not sufficient evidence to conclude that there is any plagiarism here.

VII. Conclusion

Matrix profile motif discovery is simple, fast, and parameter-free, and can be incrementally updated for moderately fast data arrival rates. By using the matrix profile motifs method to these 2 above time series data, the main goal is to clarify the similarity pairs to do future work, which is to find out the trend of the sales for the first data and find out whether the audio copied other audio or not. However, there seems to be some disadvantages. Since the matrix profile is based on the calculation of the Euclidean distance from the subsequence, when facing a huge amount of data, the running time for this method is very long and the cost for the computational expense increases. This motif will only be fast until I find an algorithm to reduce the number of the data without affecting the basic structure. Furthermore, the algorithm is just for finding similarity pairs, therefore, we can just conclude a trend based on the motif graph, not directly predict the number of sales in the future.

Although there are some disadvantages when using this method, based on the advantage it brings, the model is still helpful. In the future, when facing other types of time series data, Matrix profile motifs discovery can still be considered as one of the first models to be used.

Reference:

The UCR Matrix Profile Page - https://www.cs.ucr.edu/~eamonn/MatrixProfile.html

Introduction to Matrix Profiles - https://towardsdatascience.com/introduction-to-matrix-profiles-5568f3375d90

Part 8: AB-Joins with STUMPY - https://towardsdatascience.com/part-8-ab-joins-with-stumpy-af985e12e391

Matrix Profile I: All Pairs Similarity Joins for Time Series: A Unifying View That Includes Motifs, Discords and Shapelets

Anomaly Detection using the Matrix Profile - https://andrewm4894.com/2021/02/16/anomaly-detection-using-matrix-profile/