Comparison of Subsequence Pattern Matching Methods for Financial Time Series

Xueyuan Gong
Department of Computer and Information Science
University of Macau
mb15506@umac.mo

Yain-Whar Si
Department of Computer and Information Science
University of Macau
fstasp@umac.mo

***Abstract*— In contrast to general time series analysis, only a few numbers of studies are devoted to subsequence pattern matching methods for financial time series. In this paper, we compare the processing time and accuracy of three well-known pattern matching methods from financial time series domain and two pattern matching methods from general time series area. Our experiment was conducted on the historical data of Hang Seng Index (HSI) from Hong Kong Stock Market. Our experiment reveals that segmentation step and time distortion issues can significantly affect the performance of these methods.**

Keywords- financial time series; subsequence pattern matching; segmentation; technical pattern;

I. INTRODUCTION

There has been a long-standing interest for subsequence pattern matching on time series. In contrast to general time series analysis, only a few reported works are available on subsequence pattern matching methods for financial time series. Many approaches choose to use Perceptually Important Points (PIP) [3] to do segmentation on subsequences as a preprocessing step followed by pattern matching on segmented subsequence with Template-based (TB) [2], Rule-based (RB) [2] and Hybrid [1] approaches. Zhang et al [8] developed a pattern modeling and recognition system based on kernel regression for financial time series prediction. To the best of authors' knowledge, there is no reported work on the comprehensive analysis and comparison of the pattern matching methods for general time series against the methods commonly used for financial time series.

Against this background, two most popular methods from general time series, Euclidean Distance (ED) and Dynamic Time Warping (DTW) are chosen for comparison with three other methods from financial time series. Altogether, five subsequence pattern matching methods; Template-based (TB), Rule-based (RB), Hybrid, ED and DTW are selected in this paper for comparison. We test the effectiveness of these methods in matching Head and Shoulders (H&S) technical pattern from a financial time series from Hong Kong Stock Exchange. Our experimental results reveal that RB has achieved the best accuracy while ED is the fastest among all five methods in subsequence pattern matching.

The remainder of the paper is organized as follows. In section II, definitions and notations from this paper are introduced. Related works is discussed in section III. In section IV and V, we describe the pattern matching methods on segmented subsequences and original subsequences respectively. Experimental results and their analysis are given in section VI. Finally, section VII gives conclusion and future work.

II. DEFINITIONS AND NOTATIONS

- Time Series T: A time series $T[1,n]=<(t_1,p_1), (t_2,p_2), \ldots, (t_n,p_n)>$ is a list of tuples, where t denotes its time value while p denotes its price value. Note that $T[i,i]=(t_i,p_i)$ represents the i^{th} point of T.
- Count(T): Function to get the number of points in T.
- Subsequence S: A subsequence $S[i,j]=<(t_i,p_i), (t_{i+1},p_{i+1}), \ldots, (t_j,p_j)>$ is a sub-list of $T[1,n]$, where $1 \le i \le j \le n$.
- Segmented Subsequence X: A segmented subsequence $X[i,j]$ is a proper subset of $S[i,j]$ and, both X and S begin and end at the same point.
- Pattern P: a pattern $P[1,n]=<(x_1,y_1), (x_2,y_2), \ldots, (x_n,y_n)>$ is a time series with a specific and meaningful shape, where x denotes its time value while y denotes its price value.
- SIM(S,P): Function to compare S and P to check whether their shapes are similar.

III. RELATED WORKS

Subsequence pattern matching has been extensively reported in related literature. One category is to change time series into frequency domain such as Discrete Fourier Transformation (DFT) [5] or Discrete Wavelet Transformation (DWT) [6]. However, for traders and analysts [2], keeping time series in original time domain is more popular in financial time series pattern matching due to its intuitive and understandable representation. In fact, one of the easiest ways is to calculate the Euclidean Distance (ED) of the subsequence and the pattern. But ED approach requires the subsequence and the pattern to have the same length, which is not applicable for most applications. Thus, Chung et al. [3] proposed Perceptually Important Points (PIP) approach to segment subsequence S, so that Count(S) = Count(P). As a result, the similarity of S and P can be calculated with relative ease.

Later on, Fu et al. [2] proposed Template-based (TB) and Rule-based (RB) approaches for financial time series subsequence pattern matching. For better accuracy, Zhang et

978-1-4799-2548-3/13 $31.00 © 2013 IEEE
DOI 10.1109/CIS.2013.39

al. [1] later proposed Hybrid approach which combines Spearman's Rank Correlation (Spearman) and Rule-based (RB) approaches. However, segmentation by PIP leads to information loss, which can lead errors in pattern matching at the later stages. Hence, it is interesting to know if DTW, a method proposed by Berndt et al. [7] and is widely used in time series pattern matching area, can have a good outcome. Unlike ED, DTW does not require subsequence and pattern to have the same length.

IV. PATEERN MATCHING ON SEGMENTED SUBSEQUENCE

The procedure for subsequence pattern matching on segmented time series can be described as follows. Suppose T is a given time series and P is the pattern to be matched:

1. Set a sliding window SW of length m to get subsequence S, where Count(S) = m.
2. Produce a segmented subsequence X from S based on PIP method. During the segmentation, maintain the length of X and P to be the same. (i.e. Count(X) = Count(P)).
3. Normalize both X and P.
4. Calculate SIM(X,P) by a specific pattern matching method.
5. SW is moved to the next point and repeat the procedure.

Perceptually Important Point (PIP) Method for Segmentation: For calculating PIP, there are three criteria, which are PIP-ED, PIP-VD and PIP-PD respectively. PIP-ED is based on Euclidean distance (ED), PIP-VD is based on vertical distance (VD) and PIP-PD is based on perpendicular distance (PD). As introduced in [2], PIP-VD outperforms the other two criteria, so it is adopted in this paper and a stopping criterion is defined (with length=7) for matching H&S technical patterns.

After segmentation by PIP, time distortion needs to be taken into account. For instance, for a subsequence S=<(1,2),(2,3),(3,4),(4,5)>, an input pattern P=<(1,2),(2,4),(3,5)>, and the segmentation result X=<(1,2),(3,4),(4,5)>, there is a need for time distortion to be addressed before the pattern matching step since the given input pattern and the segmented subsequence are in different dimensions. Interestingly, TB method considers time distortion while RB and Hybrid do not address this issue. In the following sections we outline three pattern matching methods which require prior segmentation of the subsequence.

A. Template-based Method

After the segmentation process, we can perform pattern matching based on X and P. First, AD(X,P) and TD(X,P) are calculated in advance for calculating SIM(X,P) based on Equation (1) and (2).

$$\mathrm{AD}(X,P) = \sqrt{\frac{1}{n}\sum_{k=1}^{n}(p_k - y_k)^2} \quad (1)$$

$$\mathrm{TD}(X,P) = \sqrt{\frac{1}{n-1}\sum_{k=2}^{n}(t_k - x_k)^2} \quad (2)$$

where AD represents amplitude distance and TD represents time distance. Variable p_k and y_k denote the price value of X and P, and t_k and x_k denote the time value of X and P. After AD and TD are calculated, SIM(X,P) for TB can be calculated based on Equation (3).

$$\mathrm{SIM}(X,P) = w \times \mathrm{AD}(X,P) + (1-w) \times \mathrm{TD}(X,P) \quad (3)$$

where w is a user defined weight between AD and TD. We set w = 0.5 for our experiment based on [2].

B. Rule-based Method

Rule-based (RB) Method is an intuitive but hard method since similarity calculation (SIM(X,P)) for RB requires satisfying all the predefined rules. Although RB method is intuitive and easy to understand, it is difficult to analyze the patterns and there are no specific strategies for designing correct rules. RB approach works by defining a pattern visually and measuring similarity directly. In RB approach, rules are defined to describe the feature of a pattern. Rules for Head and Shoulder (H&S) pattern defined by Fu et al. [2] are outlined in Table 1. These rules are designed according to the definition of technical patterns [4].

Table 1 Rules of H&S for RB

Rule 1: p_4>p_2 and p_6	Rule 5: p_5>p_7
Rule 2: p_2>p_1 and p_3	Rule 6: DIFF(p_2,p_6)<15%
Rule 3: p_6>p_5 and p_7	Rule 7: DIFF(p_3,p_5)<15%
Rule 4: p_3>p_1	

Rule 1 from Table 1 is used to capture the condition where p_4 is bigger than p_2 and p_6 simultaneously. DIFF(p_2,p_6)<15% from Rule 6 is used to describe the condition that the difference of p_2 and p_6 should be less than 15% (i.e. DIFF(p_2,p_6)<|p_2-p_6|<0.15).

C. Hybrid Method

Hybrid method [1] for pattern matching adopts two methods, Spearman's Rank Correlation (Spearman) and Rule-based (RB), to recognize the pattern. First, Spearman is adopted for eliminating those subsequences which are obviously not similar to pattern. Then RB is used as a filter for removing false positives from the remaining subsequences. For a segmented subsequence X and a given pattern P, the detailed steps of Hybrid Method are given as follows:

1. Compare the value of every point in X and P to get the rank (relative position of the points in descending order) of each point, and store the rank into RX and RP. For example, if X = <(1,12), (2,37), (4,53), (6,41), (7,25)>, then RX = [5, 3, 1, 2, 4].
2. Calculate the Spearman Coefficient (SC) between RX and RP. The equation is given in Equation (4).

$$SC = 1 - \frac{6\times\sum_{i=1}^{n}(RX[i]-RP[i])^2}{n(n^2-1)} \quad (4)$$

where n is the size of RX and RP. $RX[i]$ and $RP[i]$ represent the i^{th} element of RX and RP.

3. Compare *SC* with a user defined threshold ε. If $SC>\varepsilon$, then go to next step. Otherwise, conclude that *X* is not similar to *P*.
4. Check whether *X* satisfies all predefined rules. If they are satisfied, conclude that original subsequence *S* is similar to *P*. Otherwise conclude that *S* is not similar to *P*. The rules adopted from [4] for checking H&S pattern by Hybrid method are shown in Table 2.

Table 2 Rules of H&S for Hybrid Method

Rule 1: $\lvert p_2-p_6\rvert<15\%$
Rule 2: $\lvert p_3-p_5\rvert<15\%$
Rule 3: *RX*[4]=1
Rule 4: *RX*[2] and *RX*[6] must be 2 or 3
Rule 5: *RX*[1] and *RX*[7] must be 5 or 6 or 7

V. Pattern Matching on Original Subsequence

Pattern matching in financial time series is usually applied on segmented time series [1], [2]. One of the reasons is that segmented time series have fewer points compared to original time series and therefore results in simpler calculation. However, after comparing the processing time (the wall clock time) of two approaches (pattern matching based on segmented or original time series) in a series of experiments, we find that the former's running time is not necessarily faster than the latter one. In addition, we find that the latter approach has its own advantages since there is no need to consider time distortion and there is no information loss during the process. This observation leads us to further investigate the time series pattern matching based on original subsequence without performing any segmentation. For a time series *T* and a pattern *P,* the algorithm for pattern matching based on original subsequence can be described as follows:

1. Set a sliding window *SW* of length *m* to get subsequence *S*, where Count(*S*) = *m*.
2. Calculating SIM(*S,P*) by different pattern matching methods.
3. *SW* is moved to the next point and repeat the procedure.

A. Euclidean Distance

ED is the most naïve way to measure the similarity of a subsequence and the pattern. For a subsequence *S* and a pattern *P* where Count(*S*) = Count(*P*), Euclidean Distance (ED) can be calculated as follows:

$$\mathrm{SIM}(S,P)=\mathrm{ED}(S,P)=\sqrt{\sum_{i=0}^{n}(p_i-y_i)^2} \qquad (5)$$

where p_i and y_i denotes the price value of *S* and *P* respectively. If ED is smaller than a user specified threshold ε, namely ED < ε, we can conclude that *S* is similar to *P*.

B. Dynamic Time Warping

Since ED can be sensitive to time distortion, an alternative method called Dynamic Time Warping (DTW) [7] is proposed for pattern matching. Suppose there are a subsequence *S* and a pattern *P*, where Count(*S*)= Count(*P*)=*n*. Note that Count(*S*) and Count(*P*) can be different since DTW can support calculating of similarity measure between two time series with different length. But for the purpose of consistency in our experiment, we set Count(*S*)=Count(*P*). The procedure of DTW is outlined as follows:

1. Calculate a distance matrix which contains the distance of every point between *P* and *S*. Each entry of the matrix is denoted as $\mathrm{D}(i,j)=(p_i-y_j)^2$, where $i\in[1,n]$, $j\in[1,n]$.
2. Calculate DTW matrix (Equation (6)) based on the distance matrix from the previous step.

$$\gamma(i,j)=\mathrm{D}(i,j)+\min\{\gamma(i,j),\gamma(i,j-1),\gamma(i-1,j-1)\} \qquad (6)$$

where $i\in[1,n]$, $j\in[1,n]$. The $\gamma(i,j)$ is an entry of DTW matrix. If $i\notin[1,n]$ or $j\notin[1,n]$, $\gamma(i,j)$ is equal to positive infinity. The $\min\{\gamma(i\text{-}1,j),\gamma(i,j\text{-}1),\gamma(i\text{-}1,j\text{-}1)\}$ denote the minimum value from of the three entries.

3. Return the last entry of DTW matrix as the result, namely $\gamma(n,n)$.

VI. Experimental Results

Our experiments are conducted on a computer with Intel Core i7-2600 CPU, 4 GB RAM and Windows 7 32-bit Enterprise Version. The compiler for development is NetBeans IDE Version 7.2 while the language used is JAVA.

First of all, all the methods are tested on the historical data of HANG SENG INDEX (HSI) from Hong Kong Stock Market. The historical data was downloaded from http://finance.yahoo.com. The duration of the time series for our experiment is from Jan. 1 2003 to Dec. 31 2012 containing 2506 points in total. Based on the extracted data, the processing time and accuracy of the pattern matching methods are compared and analyzed.

A. Processing Time

As shown in Table 3, when window size (*WS*) is equal to 31, ED is the fastest, which is about 46ms, while other methods do not have obvious difference, which are about 150ms. The processing time of all methods increase quickly when *WS* becomes bigger, while ED does not have obvious change. It is because the complexity for calculating Euclidean Distance (ED) is O(n), which is the fastest among the five methods analyzed in this article. For the case of DTW, the most time consuming task is the calculation of DTW distance and overall complexity is $O(n^2)$.

The bottlenecks of TB, RB and Hybrid methods are in the segmentation process and the complexity of these methods are $O(n^2)$. After segmentation, there are fewer points left (seven in our case) for pattern matching step and thus, resulting similar processing time for similarity calculation regardless of the nature of the similar calculation approaches used at the later stage. That is why running times of TB, RB and Hybrid are similar in the experiment results.

TB, RB and Hybrid are faster than DTW even though their complexity is the same. One of the reasons is that our stop criterion of PIP is set to length=7 and therefore it inadvertently reduces many steps required for calculation. If the stop criterion is set to a longer length, then the processing time could be longer.

Table 3 Processing time of pattern matching methods

WS	Speed of Pattern Matching Methods (ms)				
	ED	DTW	TB	RB	Hybrid
31	46	146	140	148	152
61	50	437	374	390	382
91	63	920	749	752	755

B. *Accuracy of Methods*

We choose the results of *WS*=31 as the bottom line to analyze the characteristics of each method. The total unique patterns in Table 4 denote the patterns found after eliminating overlapped patterns with +1 and -1 difference in position which are found within the window size. Thus, we consider them as redundant and select only one from the overlapped patterns found. For TB, we select one pattern among overlapped patterns which has the smallest similarity value (SIM). For RB method, we select a random one whereas in Hybrid method, we select the pattern which has the smallest value of *SC*. For ED and DTW methods, we select the pattern which has smallest ED distance or DTW distance.

Table 4 Result of five pattern matching methods

Start Date	End Date	Number of patterns found				
		ED	DTW	TB	RB	Hybrid
2003/10/13	2003/11/24	1	0	0	0	0
2004/2/2	2004/3/15	0	1	0	0	0
2004/9/9	2004/10/26	0	1	1	0	0
2005/2/4	2005/3/23	0	0	1	0	0
2006/4/5	2006/5/23	0	1	0	0	0
2007/1/16	2007/2/28	1	0	1	0	0
2007/6/25	2007/8/7	0	1	0	0	0
2007/10/9	2007/11/20	1	1	1	1	1
2008/4/16	2008/5/28	1	1	0	0	0
2008/7/8	2008/8/20	0	1	1	0	0
2009/5/25	2009/7/8	0	0	1	0	0
2009/7/23	2009/9/3	1	1	1	1	1
2010/3/22	2010/5/4	1	1	0	0	0
2010/7/20	2010/8/31	1	1	0	0	0
2010/12/28	2011/2/10	1	1	1	1	0
2011/3/29	2011/5/12	0	1	0	0	0
2011/8/9	2011/9/21	0	0	1	1	1
2012/2/10	2012/3/23	0	1	1	0	0
2012/6/13	2012/7/25	1	1	1	1	0
Total unique patterns		9	14	11	5	3

From the experiment results, we can observe that TB, ED and DTW find more patterns than RB and Hybrid method. That is because RB and Hybrid apply rules while other methods use different approaches for matching. Specifically, the methods which do not rely on rules result significantly higher number of false positives than the methods which are based on rules. Examples of false positive found by TB, ED and DTW are shown in Fig. 1.

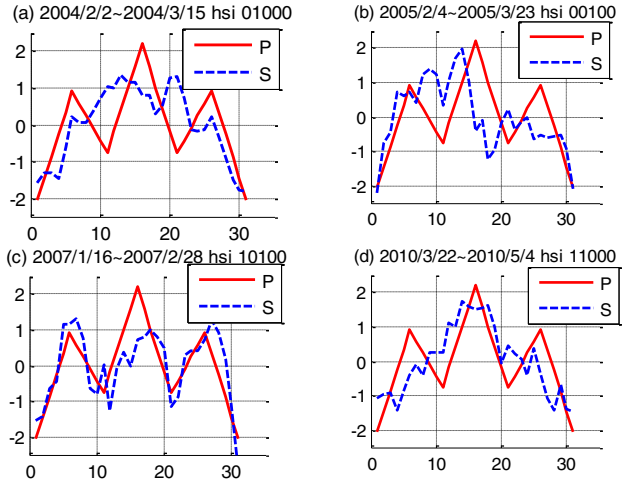

Fig. 1 False positive patterns found by TB, ED and DTW

In Fig. 1, *P* represents standard pattern while *S* represents subsequence. In these diagrams, the date denotes the duration of subsequence while 'HSI' is the abbreviation of Hang Seng Index. The five binary bits binary at the upper right corner of the figures is used to denote the type of pattern matching methods which can recognize the input. These five bits represent ED, DTW, TB, RB and Hybrid methods respectively. If a method recognizes the input as a pattern, then corresponding bit is set to one, otherwise the bit is set to zero. For instance, in Fig. 1(c), the bits 10100 are used to denote that only ED and TB methods can recognize the pattern.

From Fig. 1 (a), we can observe that DTW recognizes the subsequence as a pattern. This is caused by the lack of sensitivity to time distortion by the DTW method. In Fig. 1 (b), TB recognizes the subsequence as a pattern. The false positive is caused by the information loss during the PIP step. For illustration purpose, we show the segmented time series *X* of Fig. 1 (b) after PIP step in Fig. 2. We can see that although the left shoulder of *X* is higher than *P* and the right shoulder of *X* is lower than *P*, it is similar to *P* because of the information loss during the PIP step. Therefore, Hybrid method recognizes it as a pattern.

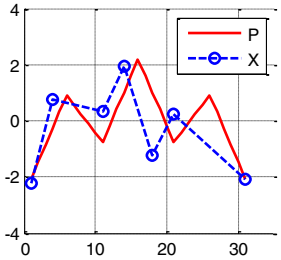

Fig. 2 Segmented subsequence during 2005/2/4~2005/3/23 in HSI

Due to the low positions of the head (in Fig. 1 (c)) and two shoulders (in Fig. 1 (d)), the subsequences are visually

not similar to *P*. Instead, the subsequence in Fig. 1 (c) is more similar to Triple Top pattern and Fig. 1 (d) is more similar to Rounded Top pattern. However, ED, DTW and TB recognize the subsequences in Fig. 1 (c) and Fig. 1 (d) as patterns. Therefore, we can conclude that the methods for calculating distance are not sensitive to small features. ED and DTW calculate the distance of every two points between *P* and *S*. TB is somewhat different from ED and DTW since it calculates the distance based on segmented subsequence *X* but not on the original subsequence *S*. However, TB is not sensitive to small features.

In contrast to ED, DTW and TB, the remaining two methods (Hybrid and RB) find fewer patterns. It shows that these methods are stricter in pattern matching compared to remaining three methods. Hybrid method uses two methods together (Spearman's Rank Correlation and Rule-based) and therefore it can be considered as the strictest among all methods under comparison. As expected, Hybrid method produces far more false negatives compared to other approaches. Fig. 3 shows two such examples.

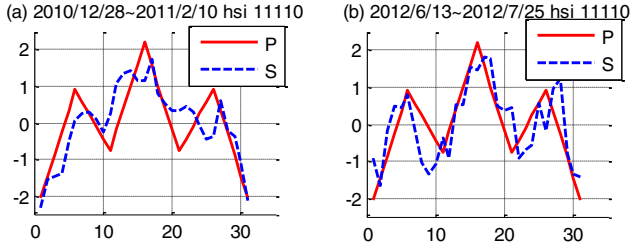

Fig. 3 Correct patterns found by ED, DTW, TB, and RB (false negatives found by Hybrid)

Obviously, subsequences in Fig. 3 (a) and (b) looks similar to the input pattern and except Hybrid, all other methods (ED, DTW, TB and RB) recognize it as a pattern. This illustrates that Hybrid is stricter than other methods and can generate higher number of false negatives.

VII. Conclusion and Future Work

In this study, the processing time and accuracy of five subsequence pattern matching methods are compared. From our experiment results, we find that the fastest method is ED while slowest method is DTW. There is no obvious difference in other remaining methods. From the perspective of accuracy, RB is the best among five methods although it does not consider time distortion after the segmentation. We also find that Hybrid causes more false negatives than other methods. Besides, information loss during segmentation process affects its accuracy as well. ED can be considered as the simplest way to do subsequence pattern matching. However it can produce higher number of false positives since ED is not sensitive to small features. In addition, ED is sensitive to time distortion compared to DTW. Compared to ED, DTW is not sensitive to time distortion; however, it also finds many false positives since it is not sensitive to small features. For TB, it only finds many false positives, but is also affected by information loss during the segmentation process. Moreover, it also finds many false positives like ED and DTW. As for the future work, we are planning to improve the processing time and accuracy of DTW method.

Acknowledgment

This research is funded by the Research Committee, University of Macau under grant MYRG041(Y1-L1)-FST13-SYW.

References

[1] Z. Zhang, J. Jiang, X. Liu, R. Lau, H. Wang, and R. Zhang, "A real time hybrid pattern matching scheme for stock time series", in proceedings of the 21st Australasian Conference on Database Technologies, vol. 103, pp. 161-170, 2010.

[2] T.C. Fu, F.L. Chung, R. Luk, and C.M. Ng, "Stock Time Series Pattern Matching, Template-based vs. Rule-based Approaches", Journal Engineering Applications of Artificial Intelligence, vol.20, No.3, pp.347-364, 2007.

[3] F.L. Chung, T.C. Fu, R. Luk, V. Ng, "Flexible time series pattern matching based on perceptually important points", in International Joint Conference on Artificial Intelligence (IJCAI) Workshop on Learning from Temporal and Spatial Data, pp. 1-7, 2001.

[4] A.W. Lo, H. Mamaysky, J. Wang, "Foundations of technical analysis: computational algorithms, statistical inference, and empirical implementation", Journal of Finance, vol. 55, pp. 1705–1765, 2000.

[5] R. Agrawal, C. Faloutsos and S. Arun, "Efficient similarity search in sequence databases" in proceedings of the Fourth International Conference on Foundations of Data Organization and Algorithms, pp. 69-84, 1993.

[6] K.P. Chan, A.C. Fu, "Efficient time series matching by wavelets", in proceedings of the 15th IEEE International Conference on Data Engineering, pp. 126-133, 1999.

[7] D.J. Berndt, J. Clifford, "Using Dynamic Time Warping to find patterns in time series", in KDD workshop, vol. 10, pp. 359-370, 1994.

[8] D. Zhang, Y. Liu, Y. Jiang, "Financial forecasting using pattern modeling and recognition system based on Kernel Regression". WSEAS Transactions on Computer 4(6), pp. 656-659, 2007.