Expert Systems with Applications 37 (2010) 1113–1123

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

# A polygon description based similarity measurement of stock market behavior ☆

Por-Shen Lai *, Hsin-Chia Fu

*Department of Computer Science, National Chiao-Tung University, Hsin-Chu 300, Taiwan*

ARTICLE INFO

*Keywords:*
Similarity measurement
Data distribution
Polygon deformation
Stock market behavior
Data mining

ABSTRACT

In this paper, we propose and implement a data modeling system to measure the similarity of stock market behavior in different time periods. To implement this system, we also propose a data distribution approximation model, *Polygon descriptor*, and a shape difference measurement, called *Deforming distance* method. The *Polygon descriptor* can model a data distribution by characterizing the data dependency relationship among the data variables, and the Deforming distance is designed to be translation, scale, and rotation invariant. The *Polygon descriptor* and *Deforming distance* method are combined to measure the similarity between data sets based on the shape of data distribution. Stock price and EPS (earn per share) data during different time periods were selected to verify the modeling and measuring power of the proposed model and method. The Polygon descriptor is used to model the collected Stock price and EPS data for their dependency relationship. Then, difference between Polygon descriptors can be measured by using the *Deforming distance* measurement method for their *Deforming distance* values. Based on the Deforming difference values, the similarity of stock market behaviors in two time periods can be assessed, and the similar stock market behavior of different time periods can be identified. To demonstrate the capabilities of the proposed method and system, real-world data of Taiwan stock market during the period from 1986 to 2006 were collected and used. Experimental results show that the *Polygon descriptor* successfully captures the data dependencies between stock price and EPS, and the *Deforming difference* based similarity measurement estimates the changes of market behavior better than some commonly used methods. A web-based prototype system is available at http://www.csie.cntu.edu.tw/~pslai/TWStockSIdx/ for public trial, evaluation, comments and suggestions.



## 1. Introduction

Finding investment reference information from historical data is a common used strategy for investment guidance. Based on the investment reference information, similarity measurement on market behaviors from different time periods are often conducted. Then, these similarity measurements can be used to retrieve useful investment information. Accordingly, an investor may decide when will be the right time to buy or to sell a targeted stock.

Also, investment experts look for stocks of similar behavior patterns in specific time periods, to make investment suggestions to help people making their investment decisions. However, there are too many data records in financial archives to be used as investment reference information, automated search mechanisms are needed to mining meaningful and useful stock behavior patterns from financial archives.

In the past, numerous researchers (Deco & Schurmann, 2001) proposed to model the dynamic behaviors from a single financial series. Often, they estimated the possible dependencies between past and future data according to various assumptions and data characteristics, such as linearity, mutual dependency, and so on. However, instead of using the estimated *instantaneous* dynamics as the descriptor of stock market behaviors, the *estimated dynamics* are used to perform short term predictions. That is, people mainly focus on using the time dependency of financial series to model system dynamics.

Recently, Terasvirta, van Dijk, and Medeiros (2005) present an excellent survey of linear model, auto-regressions, and neural networks for forecasting macroeconomics time series. By learning the behavior of economy systems from historical data, these models are used to predict the value changes in the near future. For instance, Pao (2007) proposed an artificial neural network(ANN) to predict the electricity price using direct forecasting approach based on historical data. Lin and Chen (2008) proposed a method which integrates the best features of several classification approaches by genetic-based hybrid approach, to predict the possibility of corporate failure.

On the other hand, several researchers proposed to estimate the probability of a business-cycle. Gregoir and Lenglart (2000) used

☆ This research was supported in part by the National Science Council under Grant NSC 95-2221-E-009-218.

* Corresponding author.
*E-mail addresses:* porshenlai@yahoo.com (P.-S. Lai), hcfu@csie.nctu.edu.tw (H.-C. Fu).

0957-4174/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2009.06.084

business survey data and HMM model to find turning points of business cycles. They also found that multivariate data can be used to detect turning points of business-cycles. Bellone and Gautier (2004) also used Gregoir's model in a set of four financial series to achieve an unrevised and reliable advanced qualitative probabilistic indicators. However, these models are proposed to capture the dependencies between input and output data as detailed as possible and are not intended to achieve reasoning from the learned parameters. In other words, one may have probabilistic information of certain stock market activities, but acquires little or nothing about what is real happening.

More reasoning information is always desired, so that basic knowledge and the estimated results from financial archive can be combined to conclude useful investment suggestions. A similarity measurement method is proposed to compute the similarities among records sets in financial archives. The measured similarity values will reflect the deforming degree of the shape of data distribution.

Since the shape of data distribution reflects the relations among data variables, the estimated similarities are capable to represent the change of relations among data variables. Based on the similarity measurement, data records can be searched, grouped, and analyzed to provide information for investment strategic validation, investment references data mining, etc.

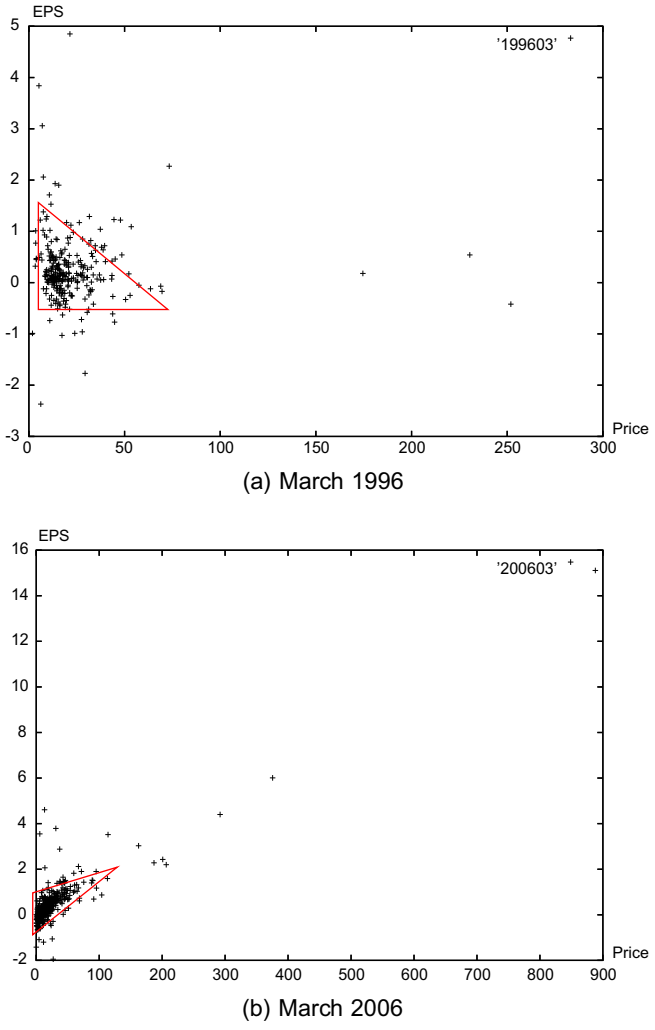


**Fig. 1.** Data distribution of stock price V.S EPS (earn per share) in 1996 (a) and 2006 (b). From the shape of these two distributions, the data distribution between stock price and EPS data in 1996 and 2006 are quite different.

The proposed similarity measurement contains two parts: (1) A Polygon descriptor, and (2) A deforming degree measurement method for polygon shape descriptors. By using the Polygon descriptor to depict the shape of a data distribution, the dependencies among stock market quantities, such as stock price and EPS (earn per share), are extracted and characterized. Based on a set of deforming operators (to be explained in Section 3), the similarity between two Polygon descriptors can be measured.

Fig. 1 shows the data distribution between stock price and EPS during 1996 and 2006 respectively. From the different shape of these two data distributions, we can see that the stock market behaviors for 1996 and 2006 are different, and from the location of shape changes, we can also notice the turning points of market behavior.

Based on the proposed methods, a prototype system is designed to achieve the following goals. (1) Unlike most financial archives which simply list data series, the proposed method provides query by example to help users to find investment reference information. That is, by grouping data records into data sets, desired data sets can be queried by specific data set. For example, user can use all data records in a month to query for monthly similar data dependencies. (2) Unlike most model-based prediction systems which forecast the future values based on a *blackbox*, the shape of data distribution of related data sets can be used by users to find reasons that induce the results.

The paper is organized as follows: the proposed data model, Polygon descriptor is introduced and formally defined in Section 2. Then Deforming distance for measuring the difference between Polygon descriptors are then defined and depicted in Section 3. Section 4 show the experimental results of the proposed methods and the web-based prototype system for real-world applications. Finally, concluding remarks are given in Section 5.

## 2. Polygon descriptor

Among various applications, measuring the shape difference between data distributions is often needed. A common approach for shape difference measurement, is first to represent a data distribution by a mathematical model. In general, linear or non-linear mathematical model is very difficult to represent the randomness of data distribution. Thus, modeling data by complex mixture models are proposed to preserve some characteristics of a data distribution. However, mixture models often lack of operational flexibilities, such as translation, scale, and rotation invariant operations, which are useful in shape difference computation. In the past, shape analysis (Loncaric, 1998; Zhang & Rosin, 2003) methods were proposed for translation, scale, and rotation invariant similarity measurement among images. However, most of these methods require shape features, such as edges or boundaries to represent an object or a data distribution. In this paper, the Polygon descriptor is proposed to represent a random and noisy data distribution in a translation, scale, and rotation invariant manner. In the following, we first present the Polygon descriptor in a formal mathematical form.

### 2.1. Model definition

Basically, the Polygon descriptor is proposed to formulate a data distribution. Given a set of data points, a Polygon descriptor can be used as feature objects to characterize the dependencies among data variables. Besides, the features represented by a Polygon descriptor can be invariant to translation, scaling, and rotation.

Since the stock market behavior is usually fluctuant and the market data values are in a random and/or noisy manner, thus invariant to translation, scale, and rotation can be essential for modeling stock market data.

In order to extend the Polygon descriptor concepts to any dimensionality, a *generalized Polygon* is first defined as follows:

1. A generalized Polygon is an union of several *convex units*.
2. A convex unit is a hyperspace surrounded by several hyperplanes.
3. A Polygon descriptor is the mathematical formulation of a convex unit.

A *Polygon descriptor* contains a reference center and $N$ normal vectors. A normal vector represents: (1) the normal direction of a hyperplane which encloses the convex unit, and (2) the distance from the reference center to each hyperplane. Fig. 2(a) shows an exemplar Polygon descriptor for the representation of the 2D data distribution in a convex unit. The reference center is located at (20,20) and five normal vectors are $\begin{pmatrix}10\\0\end{pmatrix}, \begin{pmatrix}5\\5\end{pmatrix}, \begin{pmatrix}-5\\5\end{pmatrix}, \begin{pmatrix}-10\\0\end{pmatrix}$, and $\begin{pmatrix}0\\-10\end{pmatrix}$. By applying a 1-D probability function to each normal vectors, a polygon shaped probability model is created as shown in Fig. 2(b). Suppose the 1-D probability function is a Gaussian function, then the polygon shaped probability model (distribution) can be centered at (20,20) and the variance corresponds to each normal vector can be the length of each normal vectors respectively.

By combining a few Polygon descriptors, a generalized Polygon can be represented by the proposed mathematical form. By using a generalized Polygon to model a set of data points, the data points may often need to be clustered into several groups (convex units) according to the requirements of various applications. In this paper, the Polygon descriptor is used to represent the data dependencies among data variables. Since a single convex unit is enough to describe the data dependencies among data variables for the applications addressed in this paper, thereafter, we will only discuss the Polygon descriptor for one convex unit.

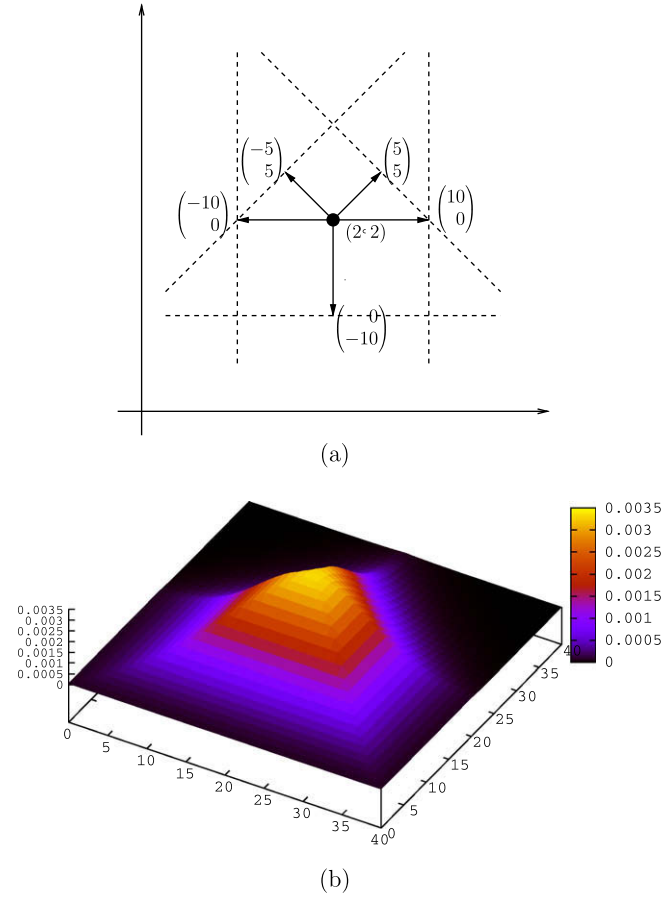


**Fig. 2.** (a) An example of a proposed Polygon descriptor, where its center is at (20,20) and normal vectors (drawn as solid arrow) to each surrounding hyperplane are $(10,0)^T, (5,5)^T, (-5,5)^T, (-10,0)^T$, and $(0,-10)^T$. (b) The pentagon shaped probability distribution for the Polygon descriptor of (a).

### 2.2. The learning algorithm of a Polygon descriptor

Based on the statistical characteristics of a data distribution, a *Polygon descriptor* can be learned from the data distribution through an iterative process. The flow chart of a Polygon descriptor learning process is shown in Fig. 3. First, the reference center is estimated, and a normal vector set $A = \{a_1, \ldots, a_j, \ldots, a_N\}$ is initialized in a random manner. Then, training data points are clustered into groups corresponding to each normal vectors in $A$. Each group of data points is used to estimate the orientation of the corresponding normal vector. Then, data point grouping and normal vector estimating processes are recursively executed until the orientation of normal vectors converges, then the length of normal vectors are estimated. The number of normal vectors can also be determined by gradually increasing the number of normal vectors until the number of different normal vectors are unchanged. In the followings, the Polygon descriptor learning algorithm is presented in four parts: (1) reference center estimation, (2) data point clustering, (3) normal vector orientation estimation, and (4) normal vector length estimation.

(1) *Reference center estimation*

The computation of reference point $c$ defined in Eq. (1) is the medoid of data set $P$.

$$c = \underset{p_i \in P}{\mathbf{argmin}} \left( \sum_{p_j \in P} norm(p_i, p_j) \right), \tag{1}$$

where $p_i$ and $p_j$ are two data points in $P$ and $norm(p_i, p_j)$ is the norm from $p_i$ to $p_j$.

(2) *Data point clustering*

Data point clustering process partitions training data points into groups, to which each normal vector corresponds. That is, the data point clustering process associates each point $p_i$ in the training data set to a normal vector $\vec{a}_w$, in the normal vector set $A = \{\vec{a}_1, \ldots, \vec{a}_j, \ldots, \vec{a}_N\}$, as shown in Eq. (2).

$$\vec{a}_w = \underset{\vec{a}_j}{\mathbf{argmax}} \in A \frac{\vec{a}_j \cdot \vec{cp}_i}{\|\vec{a}_j\|^2}, \tag{2}$$

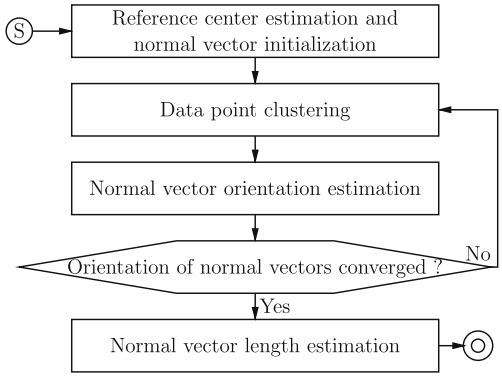


**Fig. 3.** The flow chart of Polygon descriptor learning process. Given a set of training data points, the learning process finds a reference center and normal vectors which represents the Polygon descriptor that can be best fitted to the data distribution of training data set.

where $c\vec{p}_i$ is the vector from reference center to data point $p_i$. In Eq. (2), $\vec{a}_w$ is the normal vector that associates the largest projection length $\frac{\vec{a}_j \cdot c\vec{p}_i}{\|\vec{a}_j\|^2}$. As shown in Fig. 4, the projection ratio is the ratio of the projection length $\frac{\vec{a}_j \cdot c\vec{p}_i}{\|\vec{a}_j\|}$ and the length of the normal vector $\|a_j\|$. Given two normal vectors $\vec{a}_1$ and $\vec{a}_2$, a data point $p_i$, with the identical projection ratio to $\vec{a}_1$ and $\vec{a}_2$, should satisfy the following Equation:

$$\frac{\vec{a}_1 \cdot c\vec{p}_i}{\|\vec{a}_1\|^2} - \frac{\vec{a}_2 \cdot c\vec{p}_i}{\|\vec{a}_2\|^2} = 0$$
$$\Rightarrow \left(\frac{\vec{a}_1}{\|\vec{a}_1\|^2} - \frac{\vec{a}_2}{\|\vec{a}_2\|^2}\right) \cdot c\vec{p}_i = 0$$

That is, data points having the sample projection ratio on $\vec{a}_1$ and $\vec{a}_2$ are on a hyperplane passing through the reference center and orthogonal to the vector $(\frac{\vec{a}_1}{\|\vec{a}_1\|^2} - \frac{\vec{a}_2}{\|\vec{a}_2\|^2})$. Therefore, the clustering process divide the convex unit into $N$ pyramid shaped partitions, where $N$ is the number of normal vectors.

(3) *Normal vector orientation estimation*

The data points in a cluster can be used to estimate the orientation of the associated normal vector. As shown in Fig. 5, the data points associated to a normal vector $\vec{a}$ are displayed in a gray-scale region. A hyperplane $H$, which passes through the reference center $c$, partitions the data points to two segments, $I$ and $II$. Let $\mu_1$ and $\mu_2$ be the Mean points of segments $I$ and $II$ respectively. According to the similar triangle properties, the line that connects $\mu_1$ and $\mu_2$ is orthogonal to the normal vector $\vec{a}$. Therefore, for 2-D data distribution, the orientation of a normal vector can be estimated as follows:

1. Partition the cluster into two segments by a hyperplane passing through reference center;
2. Computing the mean points, $\mu_1$ and $\mu_2$ in each segments;
3. Find a vector which is orthogonal to $\rightarrow \mu_1\mu_2$.

The generalized normal vector estimation for higher dimension data distribution can be found at http://www.csie.nctu.edu.tw/~pslai/PDGuide.

(4) *Mean length of a normal vector estimation*

To draw a boundary or outlines of a data distribution is often not feasible or in fact is not very meaningful. Thus to estimate the length of a normal vector is also meaningless. Hence, we propose to estimate the mean length of a normal vector, so that the approximated shape of a data distribution can be visualized. The mean length of a normal vector are adjusted as follows,

$$\|\vec{a}_j\| := \frac{\vec{a}_j \cdot \vec{\mu}_j}{\sqrt{\vec{a}_j \cdot \vec{a}_j}},$$

where $\vec{\mu}_j$ is the mean vector estimated using all data points associated to $\vec{a}_j$. That is, the mean length of normal vector is adjusted to be equal to the projection length of the mean vector along the normal vector.

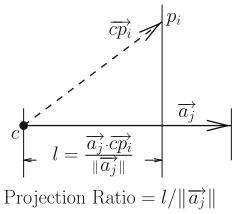


**Fig. 4.** The projection ratio for a data point $p_i$ on the normal vector $\vec{a}_j$ can be calculated by dividing the projection length $l$ by the length of the normal vector $a_j$.

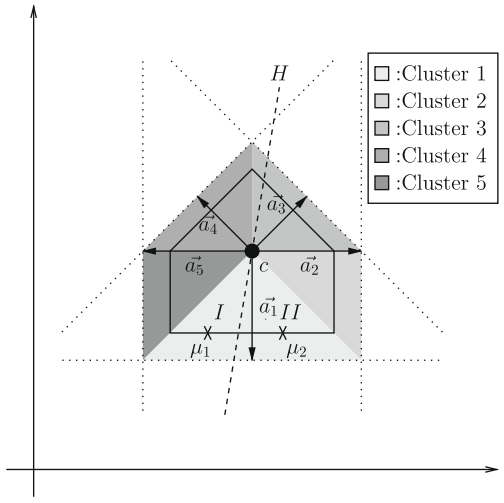


**Fig. 5.** An example to show the partitions of data cluster and the normal vector orientation estimation.

As shown in Fig. 5, the pentagon drawn in solid line illustrate the Polygon Descriptor which holds the approximated shape of a given data distribution. In fact, the data points can not be enclosed by any distinct boundary hyperplanes. Due to the excellent shape approximation on data distribution, the proposed Polygon descriptor has been applied to image (Pao, Chung, Xu, & Fu, 2008) and video (Pao, Chen, Lai, Xu, & Fu, 2008) data modeling and analysis successfully.

## 3. Deforming distance

Deforming distance is proposed to represent the difference between Polygon descriptors. Given a Polygon descriptor with $N$ normal vectors $\{n_0, \ldots, n_{N-1}\}$, a number sequence, $\{a_0, a_1, \ldots, a_{N-1}\}$, can be used to represent the Polygon Descriptor. The number sequence $\{a_0, a_1, \ldots, a_{N-1}\}$ can be enumerated as follows:

$$a_i = \|a'_i\| = \left\| cos^{-1}\left(\frac{n_i \cdot n_{(i+1)\%N}}{\|n_i\|\|n_{(i+1)\%N}\|}\right)\right\|, \quad \text{for } i = 0, \ldots, N-1,$$

where $a'_i$ is the angle between two adjacent normal vectors $n_i$ and $n_{i+1}$, $a_i$ is the magnitude of $a'_i$ and "%$N$" is the modular $N$ operation. Given a Polygon descriptor, the shape of corresponding polygon is encoded by the number sequence $\{a_0, a_1, \ldots, a_{N-1}\}$, since the angles between any two adjacent boundary edges of the corresponding polygon is equal to $\pi - a'_i$ where $n_i$ and $n_{i+1}$ are the corresponding normal vectors of the boundary edges.

The problem of measuring the difference between two Polygon descriptors are reduced to the problem of measuring the difference between two number sequences. The Deforming distance defined in Section 3.1 is used to measure the difference between two number sequences. The methods that estimating the Deforming distance are then introduced in Section 3.2.

### 3.1. The definition of Deforming distance

Let $\{a_0, \ldots, a_{N-1}\}$ be a number sequence, three deforming operations are defined as follows:

1. *Deletion*: *delete*$(i)$ – if $a_i = 0$, remove $a_i$ from the number sequence.

2. *Insertion*: *insert*$(i)$ – insert a zero in the number sequence next to $a_i$.
3. *Chang*: *change*$(i,\delta)$ – add $2\delta$ to $a_i$ and subtract one $\delta$ from both $a_{(i-1)\%N}$ and $a_{(i+1)\%N}$.

Given two number sequences $S=\{s_0,\ldots,s_n\}$ and $T=\{t_0,\ldots,t_M\}$, a sequence of deforming operations can change $S$ to $T$. The cost of the sequence of deforming operations is defined as the sum of change amount $\delta$ for all the needed *Change* operators. *Insertion* and *Deletion* operations require zero cost of the sequence of deforming operations, because these two operations just insert or remove a zero to or from number sequence. The minimal cost to convert a number sequence $S$ to $T$ using deforming operations is defined as the Deforming distance. And, the corresponding sequence of deforming operations for the Deforming distance is called Deforming path.

### 3.2. The estimation of Deforming distance

Many well-known algorithms, such as shortest-path (Manber, 1989a, chap. 7.5; Ahn & Ramakrishna, 2002), traveling salesman problem (Manber, 1989b, chap. 11.5.2), string-to-string correction problems (Manber, 1989c, chap. 6.8; Wagner & Fischer, 1974), are proposed to find the optimal path for *path-finding* problems. Among these algorithms, string-to-string correction problem is very similar to the problem of finding the Deforming path and Deforming distance described in Section 3.1. By referencing the correction path of the string-to-string correction problem, match assignments are proposed to divide the problem of Deforming path finding to several smaller sub-problems. Then, the Deforming distance for the sub-problems can be estimated.

A match assignment describes the relationship of elements between source and target number sequence. Let $S=(s_0,\ldots,s_{n-1})$ and $T=(t_0,\ldots,t_{m-1})$ be source and target sequences. Let a match assignment between $S$ and $T$ be $M=\{\left(l_0^s,l_0^t\right),\ldots,\left(l_{k-1}^s,l_{k-1}^t\right)\}$, where $k$ is the number of entries in $M, 0\leqslant l_i^s\leqslant n-1, 0\leqslant l_i^t\leqslant m-1, 0\leqslant i<k-1$, and $l_i^s$ and $l_i^t$ are the labels of the $i$-th number in $S$ and $T$ respectively. An element $\left(l_i^s,l_i^t\right)$ in a match assignment means that $t_{l_i^t}$ in $T$ is associated with $s_{l_i^s}$ in $S$. For example, the match assignment {(0,1),(1,1),(1,2),(2,0)} means that $s_0$ in $S$ is associated with $t_1$ in $T$, $s_1$ in $S$ is associated to $t_2$ and $t_3$, in $T$, and $s_2$ in $S$ is associated to $t_0$ in $T$.

When the relationship between the elements of source sequence and the elements of target sequence is fixed, the minimal cost of deforming operations to convert the source sequence to the target sequence can be measured by the methods presented in Section 3.2.1. By computing the minimal cost of deforming operations for every possible relationship between the elements of source sequence and the elements of target sequences, the Deforming distance which is the minimal cost of deforming operations without constraining the relationship of the elements between source sequence and the elements of target sequence, is available. The method that generates match assignments for every possible relationship between the elements of source sequence and the elements of target sequence is illustrated in Section 3.2.2.

#### 3.2.1. Local estimating of Deforming distance

Given a match assignment $M$ between two number sequences $S$ and $T$, a number of zeros are inserted into $S$ and $T$ to equalize the number of elements in these two sequences. Two modified source and target sequences $S'$ and $T'$ of equal number of elements are generated as follows:

- For two entries $\left(l_i^s,l_i^t\right)$ and $\left(l_{i+1}^s,l_{i+1}^t\right)$ in $M$, if $l_i^t=l_{i+1}^t$, then a zero is inserted right after $t_{l_i^t}$.
- For two entries $\left(l_i^s,l_i^t\right)$ and $\left(l_{i+1}^s,l_{i+1}^t\right)$ in $M$, if $l_i^s=l_{i+1}^s$, then a zero is inserted right after $s_{l_i^s}$.

Both of the modified sequences $S'=\{s'_0,\ldots,s'_{N'-1}\}$ and $T'=\{t'_0,\ldots,t'_{N'-1}\}$ contain $N'$ elements. By transferring a partial amount of value from elements in $S'$ to the elements next to them, $S'$ can be converted into $T\prime$, where transferring a partial amount of value $d_i$ from $s'_i$ to $s'_{i+1}$ means that $d_i$ are subtracted from $s'_i$ and added to $s'_{i+1}$.

A sequence $D=\{d_0,\ldots,d_{N'-1}\}$, called *Shift amount*, indicates the minimal transferring values $d_i$ needed to convert $S'$ to $T'$. Using the match assignment $M$ between $S'=\{s'_0,\ldots,s'_{N'-1}\}$ and $T'=\{t'_0,\ldots,t'_{N'-1}\}$, *shift amount* can be evaluated as follows:

1. First, let us map $S'$ and $T'$ to a coordinate system $(x,y)$, as shown in Fig. 6.
2. Then, plot $N'$ points $\{m_0,\ldots,m_{N'-1}\}$ where the point $m_i$ is located at $\left(\sum_{j=0}^{i}t'_j,\sum_{j=0}^{i}s'_j\right)$, for $i=0,\ldots,N'-1$. In Fig. 6, the points $\{m_0,\ldots,m_{N'-1}\}$ are marked in solid black dots.
3. Draw a line $y=x+\beta$, and plot another $N'$ points $\{p_0,\ldots,p_{N'-1}\}$ which are one-to-one corresponds to $\{m_0,\ldots,m_{N'-1}\}$ and each point $p_i$ is located at $\left(\sum_{j=0}^{i}t'_j,\beta+\sum_{j=0}^{i}t'_j\right)$, for $i=0,\ldots,N'-1$. In Fig. 6, the points $\{p_0,\ldots,p_{N'-1}\}$ are drawn in solid squares.
4. Change the value of $\beta$, such that $\sum_{i=0}^{N'-1}(x_i^m-x_i^p)$ equals to zero where $(x_i^m,y_i^m)$ and $(x_i^p,y_i^p)$ are the coordinate of $m_i$ and $p_i$ respectively.
5. Calculate shift amount $d_i$ in $D$. For each point $m_i$, the shift amount $d_i$ is equal to $y_i^m-y_i^p$. Thus, the Shift amount $d_i$ can be rewritten as follows:

$$d_i=\left(\sum_{j=0}^{i}s'_j\right)-\left(\beta+\sum_{j=0}^{i}t'_j\right),\quad \text{for } i=0,\ldots,N'-1.$$

The shift amounts $d_i$ can be solved by finding $\beta$ which minimizes $\sum_{i=0}^{N'-1}\|d_i\|$.

Since the Deforming distance is defined as the sum of deforming operations, the shift amounts $d_i$ have to be converted to $\delta_i$ which is used by the *changing operations*. An example shown in Fig. 7 illustrates the relation between $\delta_i$ and $d_i$. In general, the relation between $\delta_i$ and $d_i$ are summarized as follows,

$$d_i=\delta_i-\delta_{(i+1)\%N'},\quad \text{for } i=0,1,\ldots,N'-1.$$

Then, the change amount $\delta_i$ for the *changing operations* can be estimated by solving the following linear equations.

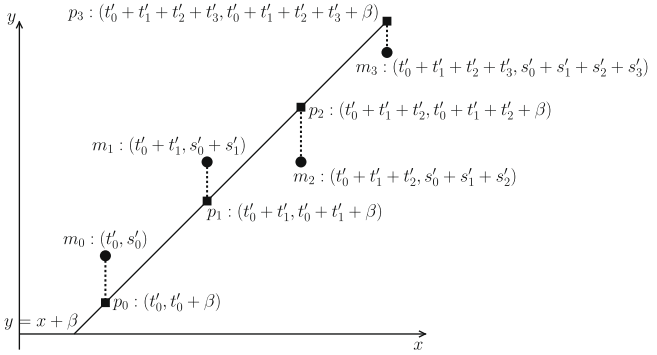


**Fig. 6.** An example which depicts how the Shift amounts are estimated. The points $\{m_0,\ldots,m_{N'-1}\}$ that correspond to the match assignment are marked in solid black dots and the match assignment corresponding points $\{p_0,\ldots,p_{N'-1}\}$ at $y=x+\beta$ are drawn in solid squares. The Shift amount $d_i$ is measured by subtracting the y coordinate value of $p_i$ from the $y$ coordinate value of $m_i$.

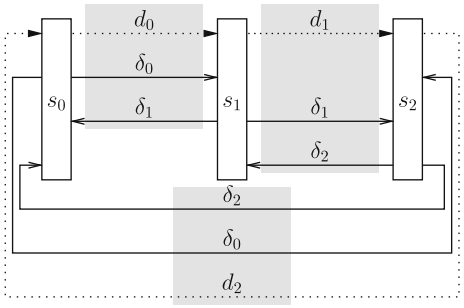


**Fig. 7.** An exemplar sequence $S'$ is used to show the relation between shift amounts $d_i$ and the change amounts $\delta_i$ for the *Chang operation*. The sequence $S\prime$ contains three elements $s_0, s_1$, and $s_2$. Three shift amounts $d_0, d_1$, and $d_2$ are moved from $s_0, s_1$, and $s_2$ respectively. The movement of change amounts $\delta_0, \delta_1$, and $\delta_2$ indicate how to use change operations to achieve the proper shift amounts among $s_0, s_1$, and $s_2$. The dotted lines indicate how the partial amount of values $d_i$ are transferred between elements. The solid lines indicate how the *changing operations* distributed partial amount of values $\delta_i$ to the adjacent elements. The light gray blocks show the related transfer between adjacent elements in number sequence. Since the amount of values transferred between elements should be the same for $d_i$ and $\delta_i$, thus $d_i = \delta_i - \delta_{i+1}$.

$$\begin{bmatrix} 1 & -1 & 0 & \dots & 0 & 0 \\ 0 & 1 & -1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & -1 \\ -1 & 0 & 0 & \dots & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta_0 \\ \delta_1 \\ \vdots \\ \delta_{N'-1} \end{bmatrix} = \begin{bmatrix} d_0 \\ d_1 \\ \vdots \\ d_{N'-2} \\ d_{N'-1} \end{bmatrix}.$$

Let $\delta_0 = \alpha$, then

$$\begin{cases} \begin{bmatrix} -1 & 0 & \dots & 0 & 0 \\ 1 & -1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & -1 \end{bmatrix} \begin{bmatrix} \delta_1 \\ \vdots \\ \delta_{N'-1} \end{bmatrix} = \begin{bmatrix} d_0 - \alpha \\ d_1 \\ \vdots \\ d_{N'-2} \end{bmatrix}. \\ \delta_{N'-1} = d_{N'-1} + \alpha = \alpha - \left( \sum_{i=0}^{N'-2} d_i \right) \end{cases}$$

Thus,

$$\begin{bmatrix} \delta_0 \\ \delta_1 \\ \delta_2 \\ \vdots \\ \delta_{N'-2} \\ \delta_{N'-1} \end{bmatrix} = \begin{bmatrix} \alpha \\ \alpha - d_0 \\ \alpha - d_0 - d_1 \\ \vdots \\ \alpha - \left( \sum_{i=0}^{N\prime-3} d_i \right) \\ \alpha - \left( \sum_{i=0}^{N\prime-2} d_i \right) \end{bmatrix}.$$

In other words,

$$\delta_0 = \alpha, \text{ and } \delta_j = \alpha - \left( \sum_{i=0}^{j-1} d_i \right) \quad \text{for } j = 1, \dots, N' - 1. \tag{3}$$

Since the Deforming distance is defined as the sum of change amount $\delta_i$ for all the needed *change operations*. That is the Deforming distance is equal to $\sum i = 0^{N'-1} |\delta_i|$. Thus, the Deforming distance can be determined by finding $\alpha$, such that $\sum_{i=0}^{N'-1} |\delta_i|$ is minimized.

Given a value $\alpha$, a set of values can be partitioned into two subsets $A$ and $B$, where elements in $A$ is larger than $\alpha$ and elements in $B$ is smaller than $\alpha$. Let $n_A$ and $n_B$ are the number of elements in $A$ and $B$ respectively, and $\mathscr{S}$ is the sum of difference between $\alpha$ and elements in $A$ and $B$. By increasing or decreasing $\alpha$ by $\sigma$, the change amount of $\mathscr{S}, \Delta\mathscr{S}$, is equal to $|\sigma| \times |n_A - n_B|$. Since $\Delta\mathscr{S} \geqslant 0, \mathscr{S}$ is minimal when $n_A = n_B$. Therefore, $\alpha$ can be determined by assigning its value equal to the Median of $\{0, \delta_0, \dots, \sum_{i=0}^{N'-1} \delta_i, \sum_{i=0}^{N'-2} \delta_i\}$. By substituting $\alpha$ to Eq. (3), the Deforming distance under the constraint of a given match assignment can be estimated.

### 3.2.2. Global Deforming distance

To generate all possible match assignments of two given number sequences, the algorithms to solve the *string-to-string correction problem* are considered.

Let $A = \{a_1, a_2, \dots, a_n\}$ and $B = \{b_1, b_2, \dots, b_m\}$ be two strings of characters and let three edit operations: (1) insertion – insert a character into a string, (2) deletion – delete a character from a string, and (3) replacement – replace one character by a different character, be used to correct the string $A$ to string $B$.

Let $A(i)$ and $B(i)$ be two sub-strings which contain the first $i$ characters of $A$ and $B$, respectively. Since the minimal editing cost from $A(i+1)$ to $B(i+1)$ can be estimated based on the combined cost correcting $A(i)$ to $B(i), A(i+1)$ to $B(i)$, and $A(i)$ to $B(i+1)$. String-to-string correction problem is generally solved by dynamic programming (Manber, 1989d). As shown in Fig. 8, a diagram is created according to strings $A$ and $B$. $C(i,j)$ which represents the minimal cost from $A(i)$ to $B(j)$, can be formulated as follows:

$$\begin{aligned} &C(i,0) = i, \quad \text{for } i = 0, \dots, n; \\ &C(0,j) = j, \quad \text{for } j = 0, \dots, m; \\ &C(i,j) = min \left\{ \begin{array}{l} C(i-1,j) + 1, \\ C(i,j-1) + 1, \\ C(i-1,j-1) + \begin{cases} 1 & , a_i \neq b_j \\ 0 & , a_i = b_j \end{cases} \end{array} \right\}, \\ &\qquad \text{for } i = 1, .., n \text{ and } j = 1, \dots, m. \end{aligned}$$

If $C(i,j)$ is estimated based on $C(i-1,j)$, that means a *deletion* is applied. If $C(i,j)$ is estimated based on $C(i,j-1)$, that means a *insertion* is applied. If $C(i,j)$ is estimated based to $C(i-1,j-1)$, that means a *replacement* is applied, or no operations is performed.

In Section 3.1, three deforming operations – insertion, deletion, and changing are defined to convert a number sequence to another. When considering an *insertion* of deforming operation to be an *insertion* of string-editing operations, a *deletion* of deforming operation to be a *deletion* of string-editing operations, and a *changing* of deforming operation to be a *replacement* of string-editing operations, a match assignment can be represented as a path in a dynamic programming diagram for string-to-string correction problem. Fig. 9 shows an example of the graphical representation of a match assignment between two number sequence $S$ and $T$. In this example, there are one *insertion* next to $s_0$, and two *deletion* at $s_2$ and $s_{m-1}$, respectively.

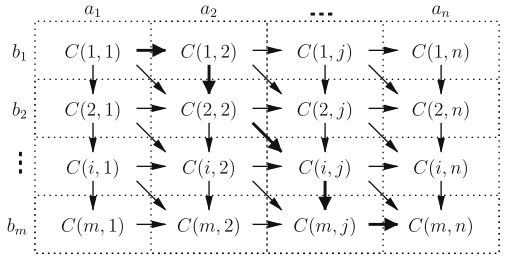


**Fig. 8.** Solving the string correction problem by dynamic programming. Let $a_i$ be the $i$-th character of string $A$ and $b_j$ is the $j$-th character of string $B$, and $C(i,j)$ represents the editing cost of to correct the first $i$ characters in $A$ to the first $j$ characters in B. Each arrow represents a dynamic programming operation. The operation starts from the left-top corner and terminates at the right-bottom corner. A minimal cost correction path from the left-top corner to right-bottom corner is searched by the dynamic programming method.

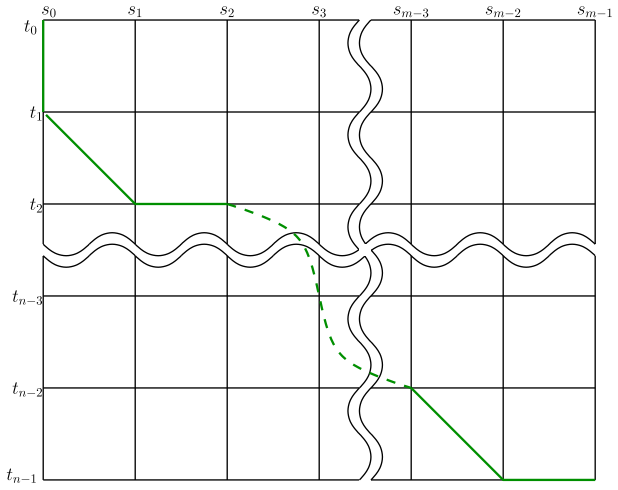


**Fig. 9.** An example of the graphical representation of a match assignment between $\{s_0,\ldots,s_{m-1}\}$ and $\{t_0,\ldots,t_{n-1}\}$. The graphical representation is similar to a correction-path on the dynamic programming diagram for string-to-string correction problem. In this example, there are one *insertion* next to $s_0$ and two *deletions* at $s_2$ and $s_{m-1}$.

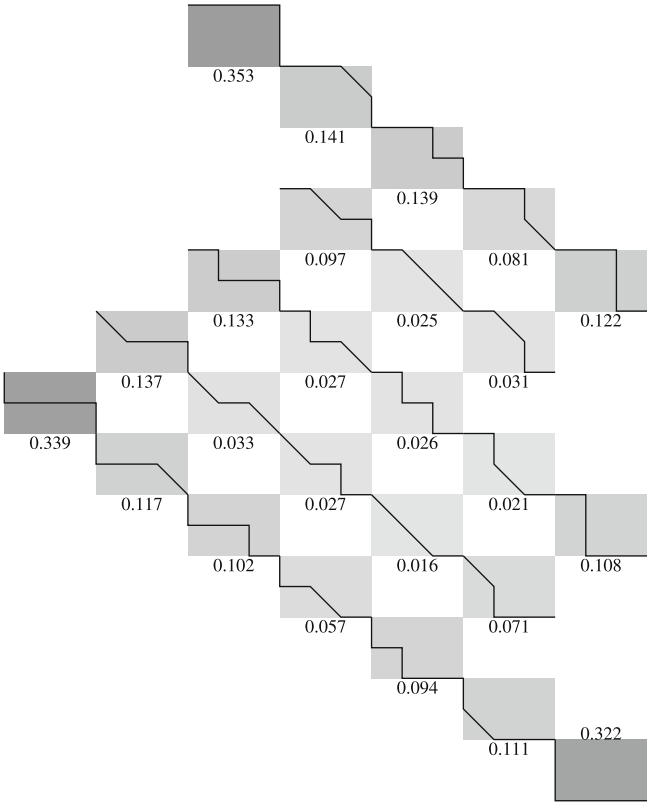


**Fig. 10.** The Deforming distance of possible match assignments to convert a sequence containing 3 elements to a sequence containing 2 elements. A block represents the match assignment corresponding a correction path of string-to-string correction problem. The solid line in each block is the graphical representation (see. Fig. 9) of a match assignment. And the darkness of blocks or the value under the block represent the minimal magnitude among the Deforming distance of the match assignments.

By generating all the possible correction path for the corresponding string-to-string correction problem, possible match assignments of two given number sequences are available. However, dynamic programming is not capable of estimating the Deforming distance of two number sequences, due to the Deforming distance of two number sequences may not relate to the Deforming distance of two sub-sequences of the two original number sequences. For example, given two number sequence $S=\{1,6,1\}$ and $T=\{2,4,2\}$, the Deforming distance is 1. Although, $S'=\{1,6\}$ and $T'=\{2,4\}$ are sub-sequences of $S$ and $T$ respectively, the Deforming distance between $S'$ and $T'$ is 4/3, which is larger than the Deforming distance between $S$ and $T$. That is, the deforming operations that are used to convert $S$ to $T$ may not include the deforming operations to convert $S'$ to $T'$.

Let $S=\{s_1,\ldots,s_n\}$ and $T=\{t_1,\ldots,t_m\}$ be two number sequences, and $S_i=\{s_i,\ldots,s_n,s_1,\ldots,s_{i-1}\}$ be the number sequence created by rotating $i-1$ elements in $S$. For each $S_i$, a Deforming distance between $S_i$ and $T$ is measured using the methods proposed in Section 3.2.1. The minimum among all the $n$ Deforming distances are selected as the global Deforming distance.

Fig. 10 shows the Deforming distance of all the possible match assignments which convert a number sequence with three elements to a number sequence with two elements. The graphical representation of each match assignment is shown as a solid line in each block. The brightness of blocks or numerical values under the block represent the minimal magnitude among the local Deforming distance under the constraint of the corresponding match assignment.

As the example shown in Fig. 10, the global *Deforming distance* can be found by searching every possible match assignments if the number of normal vectors in a Polygon descriptor is small. If the number of normal vectors in a Polygon descriptor becomes larger, the global *Deforming distance* can be approximated by using Evolutionary Computation (Lai & Fu, 2007).

Polygon descriptor in Section 2 models the shape of input data distribution by a reference center and normal vectors. For each Polygon descriptor, a number sequence can be measured based on the angle between adjacent normal vectors. By measuring the similarity between two number sequences using the proposed Deforming distance, the similarities between the shape of input data distributions are available. Therefore, the proposed methods can be used to build the similarity index among data point sets.

## 4. Evaluation and web demonstration

In this section, performance evaluation and Web demonstration of the proposed Polygon descriptor and minimal Deforming distance are conducted to exercise a prototype system by using real-world data.

Section 4.1 uses the stock price and earn per share (EPS) collected from Taiwan Stock Market during the period from 1986 to 2006 (Taiwan economic journal, 2006) to show that the idea of using Polygon descriptor and minimal Deforming distance to measure the behavior change of stock market is appropriate. Then, the proposed methods are compared to general coverage-area based methods in Section 4.2. In Section 4.3, A web-based system (Taiwan stock market, 2009) is presented to show the functionality of searching similar data sets in financial archive for historical financial data analysis.

### 4.1. Evaluation on Taiwan stock market data

To evaluate the proposed similarity measurement method, stock price and EPS (earn per share) data are used to see if the proposed method generates reliable and useful results.

Although Polygon descriptors are proposed for the representation of arbitrary dimension data, 2-D data are applied in the evaluation to ease the plotting of Polygon descriptors. First, 20 years of monthly stock price and EPS data are used to train a total of 243

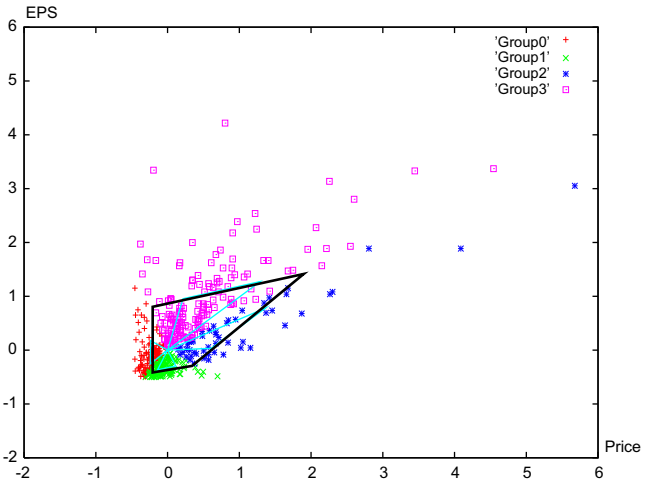


**Fig. 11.** An example of a resulted Polygon descriptor for EPS-price data distribution in February 2003. The Polygon descriptor clusters input data points into four groups. Four normal vectors are determined from the four group of data points. According to the four normal vectors, a contour of the Polygon descriptor is plotted in thick lines.

Polygon descriptors. Thereafter, the 243 Polygon descriptors are called 243PD set. The estimated Polygon descriptor for 2003/February is illustrated in Fig. 11. The input data points are clustered into four groups based on the estimated four normal vectors of the estimated Polygon descriptor. As shown in this example, a Polygon contour in thick lines and four normal vectors are used to represent the input data distribution.

Then, the similarities between every two Polygon descriptors in the 243PD Set, are measured by using the proposed minimal Deforming distance method. Fig. 12 depicts the measured similarities between every two month data. The grey intensity of each pixel $(x, y)$ in Fig. 12 represents the similarity between the $x$th month and the $y$th month during the period from January 1986 to March 2006.

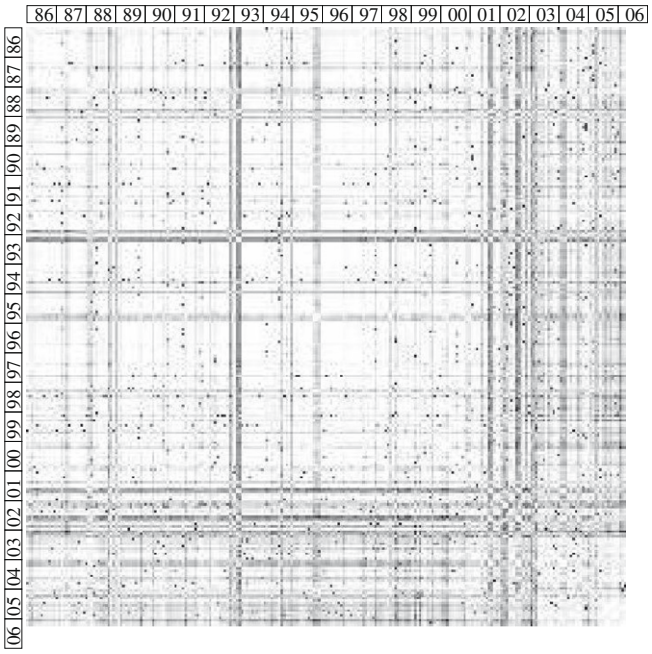


**Fig. 12.** The similarities between every 2 months of TW stock market data between 1986 and 2006. The brightness at $i$th row and $j$th column represents the similarity degree between the $i$th month and $j$th month since January 1986. Brighter intensity means higher similarity between data sets.

As shown in Fig. 12, the monthly data are similar to each other during the period between 2003 and 2006. Also, the monthly data in 199X are quiet similar to each others too. However, data in these two periods are not similar to each other. By observing the shape of stock data distribution, we noticed that the data points in the period from 2003 to 2006 are distributed more or less along a line. That means the stock price in the period from 2003 to 2006 is likely dependent on EPS, and the stock price in 90s is unlikely dependent on EPS. Fig. 11 illustrates the data distribution of the period in February 2003. That is, the change of data distribution reflects that the investors of TW stock market start to pay attention to the *intrinsic value* of stock after the crash at 2002. By using the proposed method, the estimated results match the real-world situation during that period.

### 4.2. Comparison to the other methods

According to the news paper and some popular stock investment magazines, Taiwan stock market experienced a clear turning point at the fourth quarter of 2002. In this section, we will use the information as ground truth to evaluate the proposed methods and three other methods, which are: (1) *Hausdorff distance* based, (2) *histogram* based, and (3) *mixture Gaussian* model-based methods. The price-EPS data during the period between 1998 and 2002 are used as source data. Four similarity measurement results, which are estimated by using the proposed methods and three other methods, are shown in Fig. 13. In the following, *PD + MDP* repre-

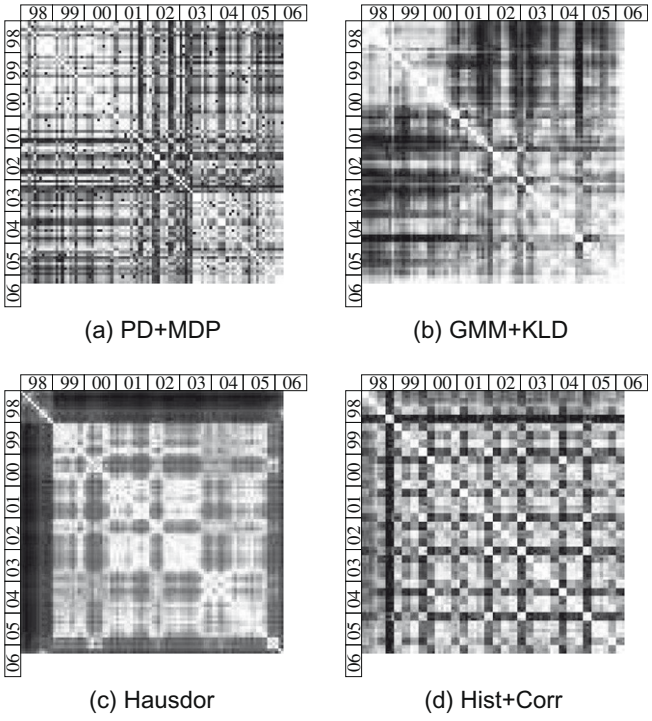


**Fig. 13.** The resulted similarity pattern of the proposed method and three common used methods. For each similarity pattern, there are 96 columns and 96 rows corresponding to 96 months price-EPS data from January 1998 to December 2005. (a) Pattern of *PD + MDP* shows the similarity results created by using Polygon descriptor and minimal Deforming distance; (b) pattern of *GMM + KLD* shows the similarity results created by using Gaussian mixture model and K–L distance; (c) pattern of *Hausdorff* shows the similarity results created by using Hausdorff distance and (d) pattern of *Hist + Corr* shows the similarity results created by using histogram and correlation measurement.

sents the proposed *Polygon descriptor* and *Deforming distance*. *GMM + KLD* represents the method that use *Gaussian mixture model* to describe the distribution of data points and *K–L distance* to measure the distance between two Gaussian mixture models. *Hausdorff* represents the method that measures the similarity based on the *Hausdorff distance*. And, *Hist + Corr* represents the method that describes the distribution of data points by *histogram* and uses the *correlation coefficient* to represent the similarity between two data sets.

As shown in Fig. 13, the *Hausdorff distance* and *histogram* based methods show that 1999 is the end of the first stage, then the market went into another stage immediately. The result pattern of *Gaussian mixture* based method shows that the market finished its first stage after the middle of 2000 (i.e. 30th months after 1998) and started the second stage at the beginning of 2002. The pattern associated with proposed method, *PD + MDP*, shows that the market finished its first stage at July 2001 (i.e 43th months after 1998) and started the second stage at the first quarter of 2003. Apparently, the end of the first stage and the start of the second stage reported by the proposed method is quiet match with the ground truth.

Among the 4 methods under evaluation, only the proposed method can show the stock market's turning point in a precise manner. We believe this measurement accuracy comes from the inherent invariant naturals, i.e., translation, scale and rotation invariant in the Polygon descriptor. These invariant operations make the proposed method to concentrate on core stock market information and to ignore surface and/or human made inferences while performing similarity measurement. However, the other three methods use the coverage-area techniques for data modeling, which usually are sensitive to some surface disturbances and/or human inference on stock price information.

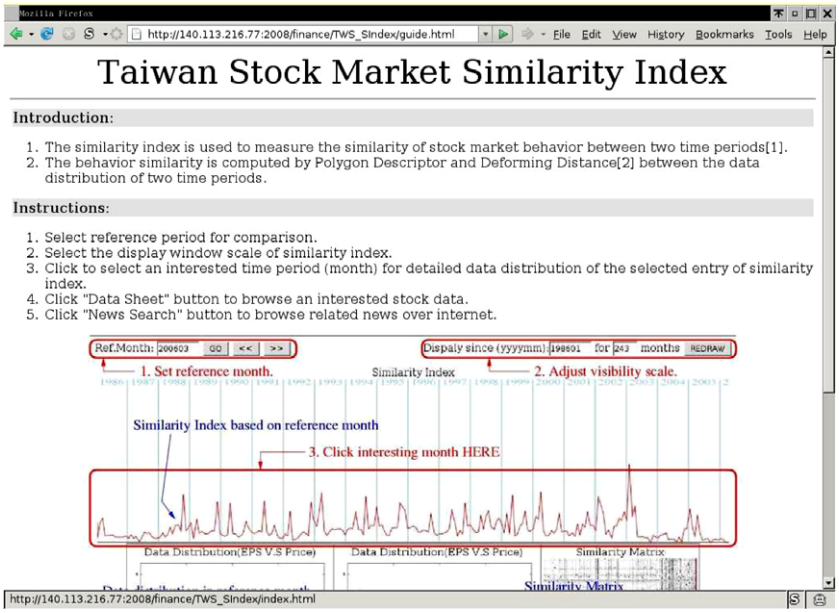


**Fig. 14.** The home page of the proposed prototype system. The contents of the home page are the system introduction and user instructions.

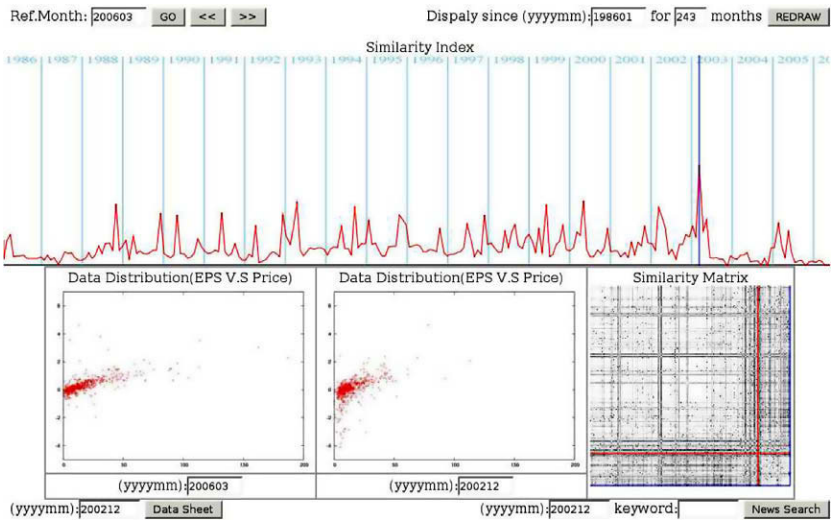


**Fig. 15.** The main user interface for the web prototype system. The user interface shows the similarities between reference month and the rest months. By clicking on the region for display of similarities, the data distribution of corresponding month are shown. Then the user can browse the data items by click the button at the left-bottom or browse the related news by click the button at the right-bottom.

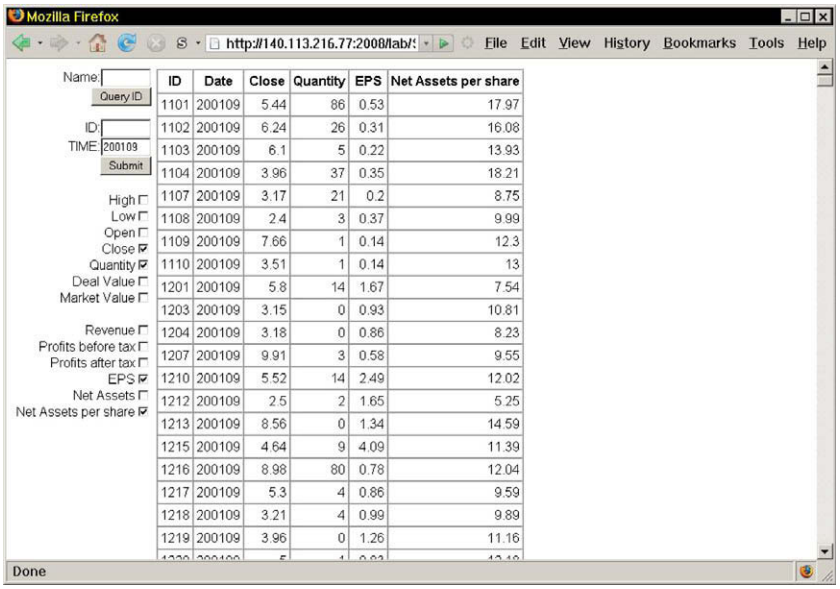


**Fig. 16.** The user interface for browsing data items at a specific time period. The time period of interests is assigned automatically. User can tick on the radio boxes to select their desired data items.

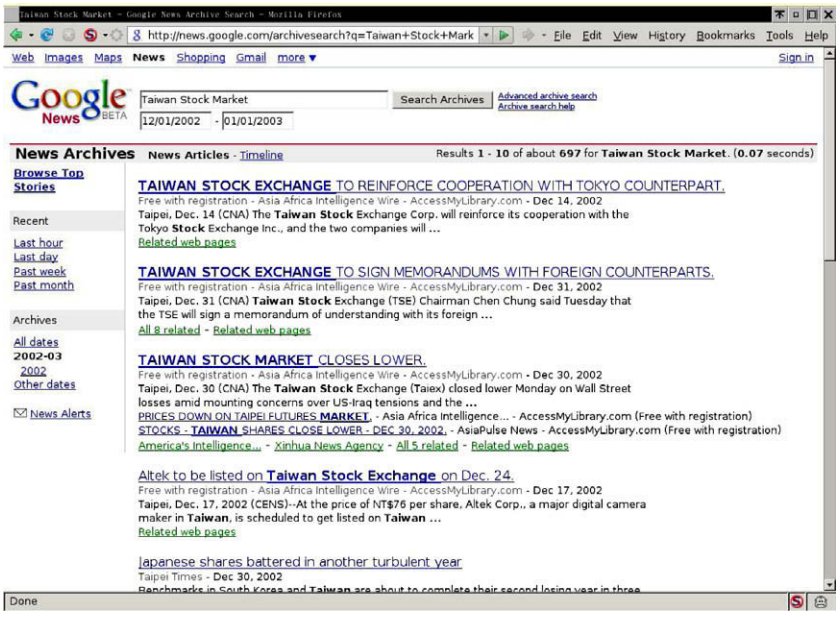


**Fig. 17.** The user interface to query related news at specific time period. The request is automatically redirected to Google News Archive Search.

### 4.3. The prototype system

Based on the similarity measurement of the proposed method, a prototype system is established for public trial, evaluation, comments and suggestions. The home page of the prototype system is shown in Fig. 14. The contents of the home page are introduction and user instructions of the prototype system. The main user interface is shown in Fig. 15. By selecting a reference month at the left-top corner, the similarity value between the reference month and the months during the period from 1986 to 2006 are drawn. The scale of the similarity drawing can be adjusted from the right-top panel. Then a user can select their interested month by clicking the similarity drawing. The data distribution of reference month and selected month are also shown below. Besides, the data sheet of selected month can be browsed by click the button at left-bottom corner.

Fig. 16 shows a data display window. A user selects data items to be browsed first, then click the *Query* button to retrieve the associated data items.

As shown in Fig. 17, clicking the button at the right-bottom corner of the main user interface sends a query to Google News Archive to retrieve the related news of the selected month.

## 5. Conclusion

Financial analysts and stock trading experts often claim that the stock price will go up when they observed the current market looks

like a bull market in the past. However, discovering the market behavior or comparing the similarity between two periods of market by plain historical values are very difficult. Therefore, this paper proposes methods to describe the market behavior by numerical expressions and measuring the behavior similarity between markets in different time periods. Base on the proposed methods, analyzing tools and data mining systems can be developed to help people study stock market data for referencing of similar behavior. In general, the system behavior can be determined by the dependencies among its data variables. Thus, the Polygon descriptor which describes the shape of sample data distribution is proposed to model the system behavior in mathematical forms. Then a number sequence is extracted from the mathematical model, Polygon descriptor, to represent a system behavior in numerical values. Since a Polygon descriptor is estimated from the statistical characteristics of sample data points, the modeling power of a Polygon descriptor is more robust to noisy data than most geometry based methods such as convex hull.

Then, Deforming distance is proposed to measure the similarity between two number sequences, which correspond to the market behavior in two time periods. Since, the proposed method is designed to capture the change of data dependences among variables, thus it can model the core features of stock market behavior better than the methods which are based on the coverage area of data distribution, such as *histogram*, *GMM*, and so on.

As the experimental results shown in Fig. 13, the proposed method actually captures the behavior changes of stock market better than the general coverage-area based methods, such as *Hausdorff distance*, *histogram* based, and *Gaussian mixture model* based methods. We also constructed a web demonstration site at http://www.csie.nctu.edu.tw/~pslai/TWStockSIdx/. The site is open for public trial, evaluation, comments and suggestions. Also, the users of this site can be benefit from the resulted similarity pattern to achieve some investment suggestions. We also plan to improve the proposed model and the prototype system according to the public comments and suggestions.

## References


Ahn, C. W., & Ramakrishna, R. S. (2002). A genetic algorithm for shortest path routing problem and the sizing of populations. *IEEE Transaction on Evolutionary Computation, 6*(6), 566–579.

Bellone, B., & Gautier, E. (2004). Predicting economic downturns through a financial qualitative hidden Markov model. Working Paper, available at http://bellone.ensae.net/bellonepaper.html.

Deco, G., & Schurmann, B. (2001). *Information dynamics*. Springer.

Gregoir, S., & Lenglart, F. (2000). Measuring the probability of a business cycle turning point by using a multivariate qualitative hidden Markov model. *Journal of forecasting, 19*(2), 81–102.

Lai, P.-S., & Fu, H.-C. (2007). A polygon description based similarity measurement of stock market behavior. In *CEC 2007. IEEE congress on evolutionary computation* (pp. 806–812). Singapore.

Lin, P.-C., & Chen, J.-S. (2008). A genetic-based hybrid approach to corporate failure prediction. *International Journal of Electronic Finance, 2*(2), 241–255.

Loncaric, S. (1998). A survey of shape analysis techniques. *Pattern Recognition, 31*(8), 983–1001.

Manber, U. (1989a). *Introduction to algorithms – A creative approach* (pp. 201–208). Addison Wesley.

Manber, U. (1989b). *Introduction to algorithms – A creative approach* (pp. 365–368). Addison Wesley.

Manber, U. (1989c). *Introduction to algorithms – A creative approach* (pp. 155–158). Addison Wesley.

Manber, U. (1989d). *Introduction to algorithms – A creative approach*. Addison Wesley.

Pao, H.-T. (2007). Forecasting electricity market pricing using artificial neural networks. *Energy Conversion and Management, 48*(3), 907–912.

Pao, H., Chen, Y., Lai, P., Xu, Y., & Fu, H.-C. (2008). Constructing and application of multimedia tv news archives. *Expert Systems with Applications: An International Journal, 35*, 1468–1472 (0957-4174).

Pao, H., Chung, S., Xu, Y., & Fu, H.-C. (2008). An em based multiple instance learning method for image classification. *Expert Systems with Applications: An International Journal, 35*, 1468–1472 (0957-4174).

Taiwan economic journal data bank (2006). URL http://www.tej.com.tw/.

Taiwan stock market similarity index (2009). URL http://www.csie.nctu.edu.tw/pslai/TWStockSIdx/.

Terasvirta, T., van Dijk, D., & Medeiros, M. C. (2005). Linear models smooth transition autoregressions and neural networks for forecasting macroeconomic time series: A re-examination. *International Journal of Forecasting, 21*, 755–774.

Wagner, R. A., & Fischer, M. J. (1974). The string to string correction problem. *Journal of the ACM, 21*(1), 168–173.

Zhang, X., & Rosin, P. L. (2003). Superellipse fitting to partial data. *Pattern Recognition, 36*(3), 743–752.