Received June 21, 2018, accepted July 23, 2018, date of publication August 3, 2018, date of current version August 28, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2862908

Modeling Similarities Among Multi-Dimensional Financial Time Series

DAWEI CHENG[1], YE LIU[1], ZHIBIN NIU[2], AND LIQING ZHANG[1], (Member, IEEE)

[1]Key Laboratory of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering, Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

[2]School of Computer Software, Tianjin University, Tianjin 300354, China

Corresponding author: Liqing Zhang (lqzhang@sjtu.edu.cn)

This work was supported in part by the National Basic Research Program of China under Grant 2015CB856004 and in part by the Key Basic Research Program of Shanghai Science and Technology Commission, China, under Grant 15JC1400103 and Grant 16JC1402800.

ABSTRACT Pairs trading is one of the most successful strategies for stock investment. The performance of pairs trading heavily depends on modeling how similarity of two paired financial signals. Conventional methods measure similarity based on one-way or two-way signal, ignoring multiple information sources. In this paper, we propose a tensor-based framework to capture the intrinsic relations among multiple factors. Equities data is represented by tensors in firm-time-trading modes, on which tensor decomposition method is applied to seek a set of multilinear patterns for each mode. In this process, structural information is preserved which provides supplementary information for pairs trading. Experiments on stocks data of all constituent firms of $S\&P500$ demonstrate the superior performance of the proposed framework when compared with some state-of-the-art methods.

INDEX TERMS Data mining, time series, tensor theory.

I. INTRODUCTION

It is widely known that equity movements are affected by various sources that cover a wide range of topics including economics, politics, and psychology. Portfolio design is a tradeoff between desired profit and potential risks. In recent years, many researchers have been devoted to studying the relationship among financial signals in investment strategy [1], [2].

One of the most successful approach is termed pairs trading [3]. Pairs trading enables traders to profit from almost any market condition by building an investment portfolio that includes a set of related stocks whose relative pricing deviates from their equilibrium state. When the correlation between the two equities weakens, i.e. one stock moves down while the other moves up, the pairs trading would be to go long on the undervalued one and to short on the overvalued stock, making a profit by unwinding the position upon convergence of the spread, or the measure of relative mispricing.

Pairs trading usually take the form of statistical arbitrage or risk arbitrage. Statistical arbitrage is a stock trading strategy, which uses signal processing methods to identify the relative mispricing between stocks [4], [5], while risk arbitrage refers to strategies involving stocks of merging companies [6]. Recently, several studies try to find pairs by modeling how similarity of two paired equities, such as correlation, partial correlation, cosine distance and matrix decomposition based distance measures [1], [7], [8].

Generally, previous studies usually measure the similarity between equities based on vector or matrix methods by vectorization or matricization of financial signals, such as correlation mentioned above, and factor model based methods. However, these approaches inevitably diminish the interrelations among the various information sources.

To address problems mentioned above, we introduce a tensor based similarity measure method for paired high-related financial signals, aiming at capturing the intrinsic relations among different modes. Tensor provides an effective and faithful representation of the structural properties of data [9], in particular, when multidimensional data or a data ensemble affected by multiple factors are involved [10], [11]. We first represent multiway stocks with a tensor form in three modes: time, firms and publicly available trade information. After that, we try to capture the inherent structure of stocks by tensor decomposition, yielding low-dimensional factors representing each mode. Finally, we calculate the similarity between equities in firms mode using Kullback-Leibler divergence and propose a tensor cluster algorithm for selecting pairs candidates.

The main contributions in this paper includes:

1) To the best of our knowledge, this is the first tensorial framework for similarity measures between paired correlative financial signals which attempts to capture the intrinsic relations among multiple trading sources.
2) The latent patterns in firm-time-trading modes are obtained by tensor decomposition, by which the inherent relations and interactions between each mode can be studied, providing additional information for investors.
3) In order to pick the most significant behavior patterns of firm, a significant similarity measures method, which combine multiple-modality information based on multilinear interactions in tensor, is proposed, yielding a superior performance of pairs trading.

In the rest part of the paper, Section 2 introduces the notations and basic multilinear algebra operations. Details of tensorial framework is elaborated in Section 3. Finally, we give a brief discussion and conclusion about our work in Section 4 and Section 5, respectively.

II. NOTATIONS AND TENSOR ALGEBRA

Nth-order tensors (multi-way arrays) are denoted by calligraphy letters, matrices (two-way arrays) by boldface capital letters, and vectors by boldface lower-case letters, e.g., $\mathcal{X}$, $\mathbf{P}$ and $\mathbf{t}$ are examples of a tensor, a matrix and a vector, respectively. The ith entry of a vector $\mathbf{x}$ is denoted by x_i, element (i, j) of a matrix $\mathbf{X}$ is denoted by x_{ij} , and element $(i_1, i_2, \ldots, i_N)$ of an Nth-order tensor $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ is denoted by $x_{i_1,i_2,\ldots,i_N}$. The order of a tensor is the number of dimensions, also knows as ways or modes. The nth element in a sequence is denoted by a superscript in parentheses, e.g., $\mathbf{U}^{(m)}$. Matricization, also known as unfolding, is the process of recording the elements of a tensor into a matrix. More, specifically, the mode-m matricization of a tensor $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ is denoted by $\mathbf{X}_{(m)} \in \mathbb{R}^{I_n\times I_1\times\cdots\times I_{m-1}\times I_{m+1}\cdots\times I_M}$, while the vectorization of a tensor is denoted as $\text{vec}(\mathcal{X})$.

Definition 2.1 (Inner Product): The inner product of two same-sized tensors $\mathcal{X}$,$\mathcal{X}'$ is defined by:

$$\langle \mathcal{X}, \mathcal{X}' \rangle = \sum_{i_1 i_2\cdots i_M} x_{i_1 i_2\cdots i_M} x'_{i_1 i_2\cdots i_M} \tag{1}$$

The Frobenius norm by $\|\mathcal{X}\|_F = \sqrt{\langle \mathcal{X}, \mathcal{X}' \rangle}$.

Definition 2.2 (Outer Product): The outer product of the tensors $\mathcal{X}$ and $\mathcal{Y}$ is given by

$$(\mathcal{X} \circ \mathcal{Y})_{i_1 i_2\ldots i_M j_1 j_2\ldots j_N} = x_{i_1 i_2\ldots i_M} y_{j_1 j_2\ldots j_N} \tag{2}$$

Definition 2.3 (Mode-n Product): The *mode-n product* of a tensor $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ and vector $\mathbf{v} \in \mathbb{R}^{I_n}$ is denoted by $(\mathcal{X}\overline{\times}_n\mathbf{v}) \in \mathbb{R}^{I_1\times\cdots I_{n-1}\times I_{n+1}\times\cdots\times I_N}$ and is defined as:

$$(\mathcal{X}\overline{\times}_n\mathbf{v}) = \sum_{i_n=1}^{I_n} x_{i_1 i_2\ldots i_N}\mathbf{v}_{i_n} \tag{3}$$

The *mode-n product* of a tensor $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ and matrix $\mathbf{A} \in \mathbb{R}^{J_n\times I_n}$ is denoted by $\mathcal{Y} = \mathcal{X} \times_n \mathbf{A} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_{n-1}\times J_n\times I_{n+1}\times\cdots\times I_N}$ and is defined as:

$$y_{i_1\ldots i_{n-1} j_n i_{n+1}\ldots i_N} = \sum_{i_n} x_{i_1\ldots i_N}\mathbf{a}_{j_n i_n} \tag{4}$$

The *mode-n product* of a tensor $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ and multiple matrices $\{\mathbf{A}^{(n)} \in \mathbb{R}^{J_n\times I_n}, n = 1, \ldots, N\}$ is denoted by $\mathcal{Y} = \mathcal{X}\prod_{n=1}^{N}\times_n\mathbf{A}^{(n)} \in \mathbb{R}^{J_1\times J_2\times\cdots\times J_N}$. Especially, the product of $\mathcal{X}$ and multiple matrices $\{\mathbf{A}^{(n)}\}_{n=1}^{N}$ except the k-th one is denoted as

$$\mathcal{X}^{(\bar{k})} = \mathcal{X}\prod_{n=1,n\neq k}^{N}\times_n\mathbf{A}^{(n)} \in \mathbb{R}^{J_1\times\cdots\times J_k-1\times I_k\times J_{k+1}\ldots\times J_N} \tag{5}$$

Definition 2.4 (Tensor Contraction): The contraction of a tensor is obtained by equating two indices and summing over all values of the repeated indices. Contraction reduces the tensor order by 2. Given two tensors $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_M\times J_1\times J_2\times\cdots\times J_N}$ and $\mathcal{Y} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_M\times K_1\times K_2\times\cdots\times K_P}$, the contraction on the tensor product $\mathcal{X}\otimes\mathcal{Y}$ along the first M modes is

$$[\mathcal{X}\otimes\mathcal{Y}; (1:M)(1:M)]] = \sum_{i_1}^{I_1}\cdots\sum_{i_M}^{I_M} x_{i_1\ldots i_M j_1\ldots j_N} y_{i_1\ldots i_M k_1\ldots k_P} \tag{6}$$

Especially, contracted product of $\mathcal{X}$ and $\mathcal{Y}$ on all indices except the k-th index is denoted as $[\![\mathcal{X}\otimes\mathcal{Y}; (\bar{k})(\bar{k})]\!]$.

Definition 2.5 (Tensor Matricization): Matricization is a process of reordering the elements of an N-th order tensor into a matrix. The mode-n unfolding of tensor $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ is denoted by $\mathcal{X}_{(n)}$ arranges the mode-n fibers into columns of a matrix. More specifically, a tensor element $(i_1, i_2, \ldots, i_N)$ maps onto a matrix element (i_n, j), where

$$j = 1 + \sum_{\substack{k=1\\k\neq n}}^{N}(i_k - 1)J_k, \quad J_k = \prod_{\substack{m=1\\m\neq n}}^{k-1} I_m \tag{7}$$

III. SYSTEM FRAMEWORK

In this paper, we propose a tensor-based financial time series framework, called TeFTS, to systematically study the relations between multiple sources on stocks like time, firms and publicly available trade information. The framework of TeFTS is sketched in Figure 1. We first represent multiway stocks in tensor form. Afterwards, tensor decomposition is applied to remove noise and capture intrinsic relations of different modes in these tensors, generating low rank tensors containing the structural information of the original data. Finally, we map the low rank tensors back to the source space, which are feed into the analytical model for extracting significant paired firms.

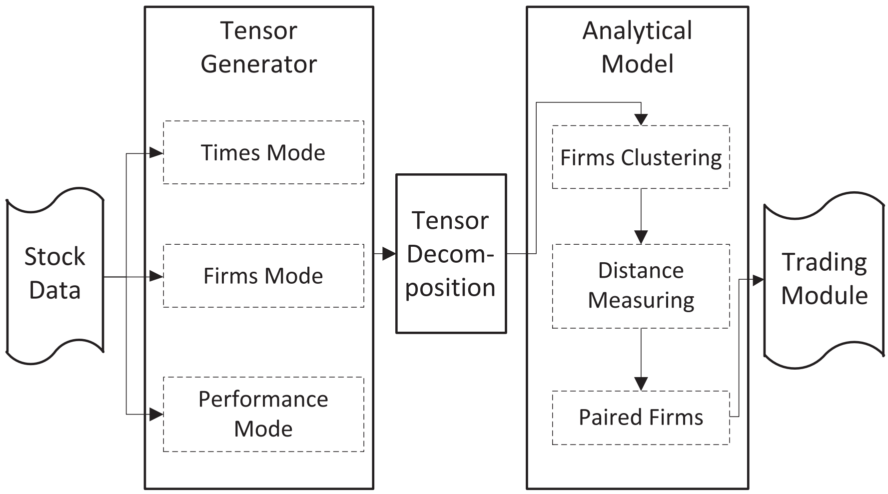


FIGURE 1. A brief view of the tensor-based framework. The framework is consisted of four components: tensor representation, tensor decomposition, analytical model and trading model.

A. MULTILINEAR TENSOR DECOMPOSITION

We utilize a tensor decomposition technique to derive latent relationships between different information modes on multiway stock tensor which are first constructed in multiway form. After that, higher-order tensor decompositions are nowadays frequently used in a variety of fields including psychometrics, chemometrics, image analysis, graph analysis, and signal processing. Two of the most commonly used decompositions are the Tucker decomposition and Canonical Decomposition /Parallel Factor Analysis (also known as CANDECOMP or simply CP) which are often considered as higher-order generalizations of the principal component analysis (PCA) or matrix singular value decomposition (SVD).

1) TUCKER DECOMPOSITION

Given a Nth-order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, its $(i_1, i_2, \cdots, i_N)$th entry is denoted by $\mathcal{X}_{i_1, i_2, \cdots, i_N}$, where $i_n = 1, \cdots, N$. The standard Tucker decomposition is defined by

$$\mathcal{X} = \mathcal{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times \cdots \times_N \mathbf{U}^{(N)} \tag{8}$$

$\{\mathbf{U}^{(n)} \in \mathbb{R}^{I_1 \times R_n}\}_{n=1}^N$ are a set of mode-n factor matrices, $\mathbf{G} \in \mathbb{R}^{R_1 \times R_2 \times \cdots \times R_N}$ denotes the core tensor and $(R_1, R_2, \cdots, R_N)$ denote the dimensions of mode-n latent space, respectively. The overall model complexity can be represented by $\prod_n R_n$ or $\sum_n R_n$, whose minimum associated values $\{R_n\}_{n=1}^N$ is termed as multilinear rank of tensor $\mathcal{X}$. For a specific $\mathbf{U}^{(n)}$, we denote its row vectors by $\{\mathbf{u}_{r_n}^{(n)} | r_n = 1, \cdots, R_n\}$

Definition 3.1 (Kronecker Products): Let $\{\mathbf{U}^{(n)} \in \mathbb{R}^{I_1 \times R_n}\}_{n=1}^N$ denote a set of matrices, the sequential Kronecker products in a reversed order is defined and denoted by

$$\bigotimes_n \mathbf{U}^{(n)} = \mathbf{U}^{(N)} \otimes \mathbf{U}^{(N-1)} \otimes \cdots \otimes \mathbf{U}^{(1)}.$$
$$\bigotimes_{k \neq n} \mathbf{U}^{(k)} = \mathbf{U}^{(N)} \otimes \cdots \otimes \mathbf{U}^{(n+1)} \otimes \mathbf{U}^{(n-1)} \otimes \cdots \otimes \mathbf{U}^{(1)} \tag{9}$$

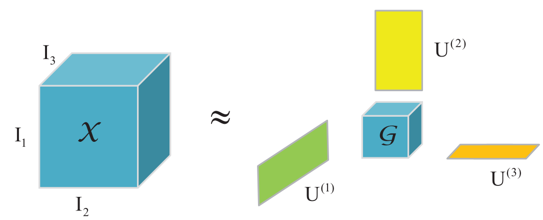


FIGURE 2. Tucker model is a weighted sum of the product of multiple component matrices representing each mode and a core tensor defining a linking structure between the set of components.

The symbol $\otimes$ denotes Kronecker product. $\bigotimes_n U^{(n)}$ is a matrix if size $(\prod_n I_n \times \prod_n R_n)$ The Tucker decomposition can be also represented by using matrix, vector or element-wise forms, given by

$$\mathbf{X}_{(n)} = \mathbf{U}^{(n)} \mathbf{G}_{(n)} \left(\bigotimes_{k \neq n} \mathbf{U}^{(k)T} \right),$$
$$vec(\mathcal{X}) = \left(\bigotimes_n \mathbf{U}^{(n)} \right) \text{vec}(\mathcal{G}),$$
$$\mathcal{X}_{i_1 \cdots i_N} = \left(\bigotimes_n \mathbf{u}_{in}^{(n)T} \right) \text{vec}(\mathcal{G}). \tag{10}$$

It should be noted that the multilinear operation is significantly efficient for computation. For example, if we compute $\bigotimes_n \mathbf{U}^{(n)}$ firstly and then multiply it with $\text{vec}(\mathcal{G})$, both the computation and memory complexity is $\mathcal{O}(\prod_n I_n R_n)$. In contrast, if we apply a sequence of multilinear operations $(\cdot) \times_n \mathbf{U}^{(n)}$ without explicitly computing $\bigotimes_n \mathbf{U}^{(n)}$, the computational complexity is $\mathcal{O}(\min_n (R_n) \prod_n I_n)$ while the memory cost is $\mathcal{O}(\prod_n I_n)$. In this paper, we use notation $\bigotimes_n(\cdot)$ frequently for clarity however, the implementation can be performed by using multilinear operations.

2) CANDECOMP-PARAFAC (CP) DECOMPOSITION

The CP Decomposition algorithm decomposes a given tensor into a sum of multi-linear terms, which can be formulated as follows. Given a three order tensor $\mathcal{X}^{I\times T\times Q}$ and the positive index J, find three-component matrices, also called loading matrices or factors, $A = [a_1, a_2, \ldots, a_J] \in R^{I\times J}$, $B = [b_1, b_2, \ldots, b_J] \in R^{T\times J}$ and $C = [c_1, c_2, \ldots, c_J] \in R^{Q\times J}$ which perform the following approximate factorization:

$$\mathcal{X} = \sum_{j=1}^{J} \mathbf{a}_j \circ \mathbf{b}_j \circ \mathbf{c}_j + \mathcal{E} \tag{11}$$

or equivalently in the element-wise form

$$x_{itq} = \sum_{j=1}^{J} a_{ij} \circ b_{tj} \circ c_{qj} + \mathcal{E} \tag{12}$$

Essentially, CP decomposition is a particular case of Tucker decomposition. Thus, in this study, we apply Tucker decomposition on stock tensors to derive latent relationships inherent in a tensor. Tucker decomposition is a form of higher-order PCA, which decomposes a tensor into a core tensor multiplied by a matrix along each mode.

B. ANALYTICAL MODULE

The core tensor $\mathcal{G}$ and projected matrices $\mathbf{U}^{(i)}$ (ith mode) learned in tensor decomposition will be fed into the analysis module shown in Fig. 3, which are consisted of the clustering methods and measuring methods based on each sector.

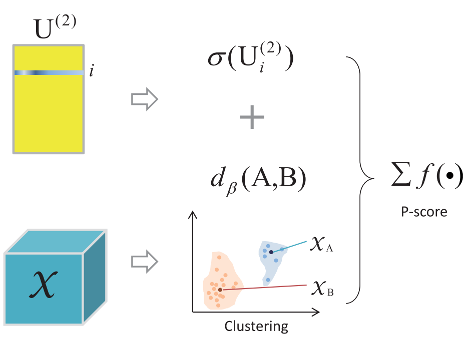


FIGURE 3. Analysis module: clustering method and measuring methods based on each sector. Both clustering method and measuring method aims to find the significant firms in each sector. Here U_2 represents the projection matrix of firm mode.

We aim to explore the connection and significant firms by historical data, thus we model each firm by $\mathbf{U}^{(i_n)}$ in the other modes, and reconstruct the original tensor by:

$$\widehat{\mathcal{X}} = \mathcal{G} \times_i \prod_{i\neq j} \mathbf{U}^{(i_n)} \tag{13}$$

Where j is firm mode of tensor $\mathcal{X}$, and $\mathbf{U}^{(i_n)}$ is n principle components of $\mathbf{U}^{(i)}$ since Tucker decomposition is a form of higher-order PCA. In this process, each firm is elite by dimensionality reduction. Then, we split them into a $N-1$-th order tensor in the firm mode.

1) DISTANCE DEFINITION

Once we obtain the reconstructed tensor $\widehat{\mathcal{X}}$, we introduce β-divergence to calculate the distance among firms.

Definition 3.2 (β-Divergence): The β-divergence between two tensors $\mathcal{A}$ and $\mathcal{B}$ with the same size is

$$d_\beta(\mathcal{A}, \mathcal{B}) = \frac{1}{\beta(\beta-1)} \sum_{I_N} \left(\mathcal{A}_{I_N}^{\beta} + (\beta-1)\mathcal{B}_{I_N}^{\beta} - \beta \mathcal{A}_{I_N} \mathcal{B}_{I_N}^{\beta-1} \right) \tag{14}$$

where $\beta \geqslant 0$ is a constant, and $I_N = i_1, \cdots, i_N$.

For completeness, by making use of the limit theory, we define $d_\beta(\mathbf{A}, \mathbf{B})$ for $\beta = 0$ and $\beta = 1$ between two matrices $\mathbf{A}$ and $\mathbf{B}$ as follows.

$$\begin{aligned} d_0(\mathbf{A}, \mathbf{B}) &= \lim_{\beta\to 0} \sum_{ij} \left(\mathbf{A}_{ij} \frac{\mathbf{B}_{ij}^{\beta-1}}{1-\beta} - \frac{\mathbf{A}_{ij}^{\beta} - \mathbf{B}_{ij}^{\beta}}{\beta} + \frac{\mathbf{A}_{ij}^{\beta}}{\beta-1} \right) \\ &= \sum_{ij} \mathbf{A}_{ij}(\log \mathbf{A}_{ij} - \log \mathbf{B}_{ij}) + (\mathbf{B}_{ij} - \mathbf{A}_{ij}) \end{aligned} \tag{15}$$

$$\begin{aligned} d_1(\mathbf{A}, \mathbf{B}) &= \lim_{\beta\to 1} \sum_{ij} \left(\mathbf{A}_{ij} \frac{\mathbf{A}_{ij}^{\beta-1} - \mathbf{B}_{ij}^{\beta-1}}{1-\beta} + \frac{\mathbf{B}_{ij}^{\beta} - \mathbf{A}_{ij}^{\beta}}{\beta} \right) \\ &= \sum_{ij} \mathbf{A}_{ij}(\log \mathbf{A}_{ij} - \log \mathbf{B}_{ij}) + (\mathbf{B}_{ij} - \mathbf{A}_{ij}) \end{aligned} \tag{16}$$

β-divergence is a very general divergence: $d_0(\mathbf{A}, \mathbf{B})$, $d_1(\mathbf{A}, \mathbf{B})$, $d_2(\mathbf{A}, \mathbf{B})$ correspond to the Itakura-Saito distance, generalized Kullback-Leighbler divergence and Euclidean distance [12].

2) CLUSTERING ALGORITHM

Given tensor datasets $S = \{\widehat{\mathcal{X}}_i\}_{i=1}^{N}$, which contain N elements and $n_c(\geqslant 0)$ clusters. $\{m_j\}_{j=1}^{n_c}$ denotes the data elements of each cluster center, e.g., $\widehat{\mathcal{X}}_{m_j}$ is the jth cluster center. $\{c_i\}_{i=1}^{N}$ represents element i belongs to cluster c_i. $\{b_i\}_{i=1}^{N}$ denotes the distance closest element among datasets S, in which the local density is larger than $\widehat{\mathcal{X}}_i$. It is defined as

$$n_{q_i} = \begin{cases} \arg\min_{q_j, j<i}\{d_\beta(q_i, q_j)\}, & i \geqslant 2, \\ 0, & i = 1. \end{cases} \tag{17}$$

Where q_i is a subscript sequence of descend ordered local density ρ_i. ρ_i is defined as

$$\rho_i = \sum_{j\in I_S\setminus\{i\}} \chi\left(d_\beta(i, j) - d_\beta(c)\right) \tag{18}$$

$\chi(x) = 1$ if $x < 0$ and $\chi(x) = 0$, otherwise. $d_\beta(c)$ is cut-off distance. δ_i is the distance of the closest data point of higher density, and defined as $\delta_i = \min_{j:\rho_j>\rho_i}(d_\beta(i, j))$. $\{h_i\}_{i=1}^{N}$ is a flag of whether the data element is a cluster core or cluster halo. If $h_i = 0$, it means $\widehat{\mathcal{X}}_i$ belongs to cluster core [13]. The main process of cluster analysis is showed in Algorithm 1.

Algorithm 1 The Clustering Analysis of Tensor Datasets

Input: Tensor dataset $S = \{\widehat{\mathcal{X}}_n^{I_1 \times I_2 \times \cdots \times I_{M-1}}\}_{n=1}^{N}$, $\widehat{\mathcal{X}}_n^{I_1 \times I_2 \times \cdots \times I_{M-1}}$ denotes the n-th element of tensor for clustering; the cut-off distance $d_\beta(c) \in (0, 1)$

Output: $\{c_i\}_{i=1}^{N}$ represents element i belongs to cluster c_i; $\{h_i\}_{i=1}^{N}$, flag of cluster core or cluster halo.

begin

- $d_{ij} = d_\beta(i, j)$, $i < j$, $i, j \in I_S$; $n_i = 0$;
- $\rho_i = \sum_{j \in I_S \setminus \{i\}} \chi\left(d_\beta(i, j) - d_\beta(c)\right)$
- $\{q_i\}_{i=1}^{N}$ = descend ordered index of ρ_i
- **for** $i = 2, 3, \cdots, N$ **do**
  - $\delta_i = \max_{i<j}(d_\beta(i, j))$
  - **for** $j = 1, 2, \cdots, i-1$ **do**
    - **if** $d_\beta(\widehat{\mathcal{X}}_{q_i}, \widehat{\mathcal{X}}_{q_j}) < \delta_{q_i}$ **then**
      - $\delta_{q_i} = d_\beta(\widehat{\mathcal{X}}_{q_i}, \widehat{\mathcal{X}}_{q_j})$
      - $n_{q_i} = q_j$
- $\delta_{q_1} = \max_{j \geqslant 2}\{\delta_j\}$
- Pick k points as centers from top k ρ,δ elements into $\{K\}$
- **for** $i = 1, 2, \cdots, N$ **do**
  - **if** $\widehat{\mathcal{X}}_i \notin \{K\}$ **then**
    - $c_i = -1$; $c_{q_i} = c_{n_{q_i}}$
  - **else**
    - $c_i = k$
- for each cluster, set $\{\rho_i^b\}_{i=0}^{n_c}$ as mean density
- **for** $i = 1, 2, \cdots, N$ **do**
  - **if** $\rho_i < \rho_{c_i}^b$ **then**
    - $h_i = 1$

C. PAIRS PICKING AND TRADING STRATEGY

In each sector, different firms are clustered into different classes, in which p-score is utilized to pick a seed representing the significant firm for each cluster. p-score is defined as

$$\{p\text{-score}\}_i = a\sigma\left(\mathbf{U}_i^{(i_n)}\right) + (1-a)d_\beta\left(\widehat{\mathcal{X}}_i, \widehat{\mathcal{X}}_c\right) \tag{19}$$

where $\sigma(\mathbf{x})$ is the variance of vector $\mathbf{x}$. $\widehat{\mathcal{X}}_c$ denotes the center element of a cluster. We choose the highest p-score element as the seed firm, and then calculated the closest β-divergence as paired firm in each cluster.

Here we define some basic notations. Let P_A^t represents price of equity A on day t. Spread value $SP_{A,B}^t$ as $SP_{A,B}^t = \log(P_B^t) - \gamma \log(P_A^t)$, where γ is the regression coefficient of $log(P_A^t)$ and $log(P_B^t)$ by OLS (ordinary least squares). Afterwards, we utilize z-score to evaluate the difference between prices, in which $z\text{-score}_{A,B}^t = \frac{SP_{A,B}^t - mean(SP_{A,B})}{var(SP_{A,B})}$. Finally, $SP_{A,B}$ needs to be statistical cointegration, which means we need to test the stability of time series, such as Augumented Dickey-Fuller(ADF) test, Elliott-Rothenberg-stock test, Schmidt-Phillips test, etc [14], [15], [16].

Pairs trading strategy is simplified as the below rules:

Rule 1: ($BASB$) when $z\text{-score}_{A,B}^t > C_a$, buy a setting quota of equity A, and sell B.

Rule 2: ($SABB$) when $z\text{-score}_{A,B}^t < C_b$, sell a setting quota of equity A, and buy B.

Rule 3: ($SASB$) when $z\text{-score}_{A,B}^t \in (\lambda_1 C_b, \lambda_2 C_a)$, balance A and B in order to withdrawn from the market.

Where C_a,C_b,λ_1,λ_2 are parameters adjusted by different market situation.

Once the pairs trading candidate is captured, we apply the trading strategy on all the constituent firms, as described in Algorithm. 2.

Algorithm 2 Pair Trading Strategy

Input: Paired Stocks A and B; Marketing information of paired stocks on day t P_A^t, P_B^t; Given investment amount M.

Output: Position Ratio of equity A and B in day t, $PO_A(t)$, $PO_B(t)$. market capitalisation of invest in day t, M_t

begin

- Set Initial Parameters: C_a,C_b,λ_1,λ_2;
- Set Initial Position Ratio: $PO_A(0)$, $PO_B(0)$;
- Set Trading Ratio: τ
- **for** $t = 1, 2, \cdots, T$ **do**
  - $SP_{A,B}^t = \log(P_B^t) - \gamma \log(P_A^t)$;
  - $z\text{-score}_{A,B}^t = \frac{SP_{A,B}^t - mean(SP_{A,B})}{var(SP_{A,B})}$;
  - **if** $z\text{-}score_{A,B}^t > C_a$ **then**
    - Buy A in ratio τ, Sell B the same;
    - Update Position Ratio Position Ratio;
  - **else if** $z\text{-}score_{A,B}^t < C_b$ **then**
    - Buy B in Ratio τ, Sell A the same;
    - Update Position Ratio Position Ratio;
  - **else if** $z\text{-}score_{A,B}^t \in (\lambda_1 C_b, \lambda_2 C_a)$ **then**
    - Trade A and B based on current $PO_A(t)$, $PO_B(t)$;
    - Adjust $PO_A(t)$, $PO_B(t)$ to $PO_A(0)$, $PO_B(0)$ in in Ratio τ;
  - Calculating M_t;

IV. EXPERIMENTS

In this section, we present the experimental results based on $S\&P$ 500 stock data. In detail, we show the low rank characteristic of tensor stock by tensor decomposition, followed by the illustration of significant firms by clustering method. Finally, pairs trading strategy is applied on the paired firms for investment.

A. DATASETS

We collect the dataset from $S\&P$ 500, which traded from Jan 1, 2004 to Dec 31, 2015, to verify the effectiveness of our proposed framework. 448 firms of them sectors are chosen listed in $S\&P$ 500. These sectors include: Industrials, Health Care, Information Technology, Financial, Utilities, Materials, Consumer Staples, Consumer Discretionary, Energy and Telecommunication Services. All publicly firms' trading information in a day are corporated including opening price, closing price, highest price, lowest price and trading volume.

In data cleaning process, we first fill the missing value. There are total 9 companies with missing values in last 12 years, e.g., AAL(American Airlines Group) was public traded since Feb 2, 2015, which means there is no data between Jan 1, 2004 and Feb 2, 2005. In this paper, we fill the missing data by means of historical value.

After data cleaning, we construct the three dimensional tensor data, consisted of time, firms and trading modes. In total, There are 3021 trading days, 448 firms and 5 trading values, which are split into training set and testing set by time mode. Data from the early 2769 of 3021 trading days was chosen as the training set represented as tensor $\mathcal{X} \in \mathbb{R}^{2769\times448\times5}$, while the latest one year data is reserved for testing, denoted by $\mathcal{Y} \in \mathbb{R}^{252\times448\times5}$.

B. TENSOR DECOMPOSITION RESULTS

In this paper, we utilize tucker decomposition to find low-rank multilinear patterns in high order data, reserving the structural information. Afterwards, by visualizing the firm patterns obtained during tucker decomposition, we try to find the significant firms in each sector.

1) LOW RANK PATTERNS

In order to display structural information of stock data, we illustrate the tensor rank contained in the core tensor and corresponding residual, as showed in Fig. 4. In Fig. 4, left part shows all the rank values in descending order while right part represents the residual when we reconstruct the data into source space under different number of principle components. It is obvious that the rank values declined and are close to 0 when the rank number reaches 30, revealing the sparsity characteristic and distinctive structural information of the stock data. Moreover, since Tucker decomposition is a form of higher-order PCA, we try to reconstruct the

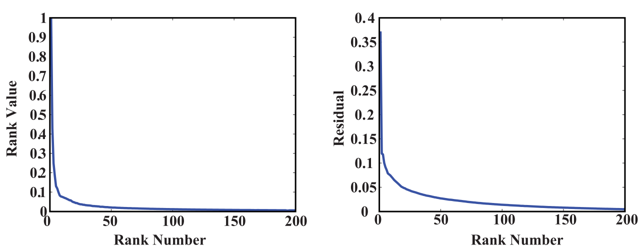


FIGURE 4. **Rank information underlied in core tensor. Left part shows all the rank values in descending order while right part represents the residual when we reconstruct the data into source space under different number of principle components.**

data into source space under different number of principle components, and then calculate the residual. Here, residual is calculated by $Res = \frac{\|(\mathcal{X}-\widehat{\mathcal{X}})\|_2^2}{\|\mathcal{X}\|_2^2}$, where $\mathcal{X}$ is the original data and $\widehat{\mathcal{X}}$ is the reconstruction data. It is obvious that the residual is relatively small when we only adopt top 30 principle components for reconstruction, which means we can get an acceptable trade-off between computational complexity and data representation performance.

2) FIRM PATTERNS

In this paper, we focus on capturing significant firms in each sector which were fed into pairs trading strategy for investment. By visualizing the firm patterns in each sector, we try to find the difference between significant firms and conventional ones, revealing the particular characteristic in each sector. Figure 5 presents the projection matrix obtained during tensor decomposition for firm mode in four sectors. Each row represents each firm and columns denote principle component number. Red color represents higher value while green one show lower values. The more diverse in a row, the more significant of the company. For most firms of the firms, we can see that the values changed slightly with the increase of principle component number. However, there exist some small part of firms where they are highly diversed historically in all sectors, which reminds us the behavior of significant firms change a lot in time series. Interestingly, there are still some difference among sectors. Majority of firms in sectors are gray, in particular, only one company in industry is diversed. However, there are more diverse firms in I.T. (information technology) sector, which may imply that there are more behavior significant firms in last decades in I.T. compared with conventional industry sector.

C. SEED FIRMS SELECTION BY CLUSTERING

In this section, we try to find the most significant firm as seed firm in each sector, assembling pairs of seeds for pairs trading strategy. Here we use cluster method mentioned in section III-B.2 to differentiate significant firms from conventional ones. Consequently, we need to correctly find firms, which are far away from cluster centers, as the significant firms. Thus, in each sector, we set the cluster number to one and then find the cluster center of all firms. Afterwards, by calculating the β-divergence between each firm and the center firm, where β is set to 0.5 [12], we evaluate the scatter divergence of firms in each sector. Finally, top-K firms whose distance are away from the center firm are selected as seed candidates.

Figure 7 shows the clustering degree for the four chosen sectors of Financials, Consumer Discretionary, I.T., and Industry. x-axis shows the firm ID while y-axis are the distance. It is obvious that most of the firms are close to the center firm, demonstrating firms clustered well around the center firm. A few firms are relatively far away from the the center firm, which means that our cluster algorithm could differentiate significant firms. The distance of each firms with its

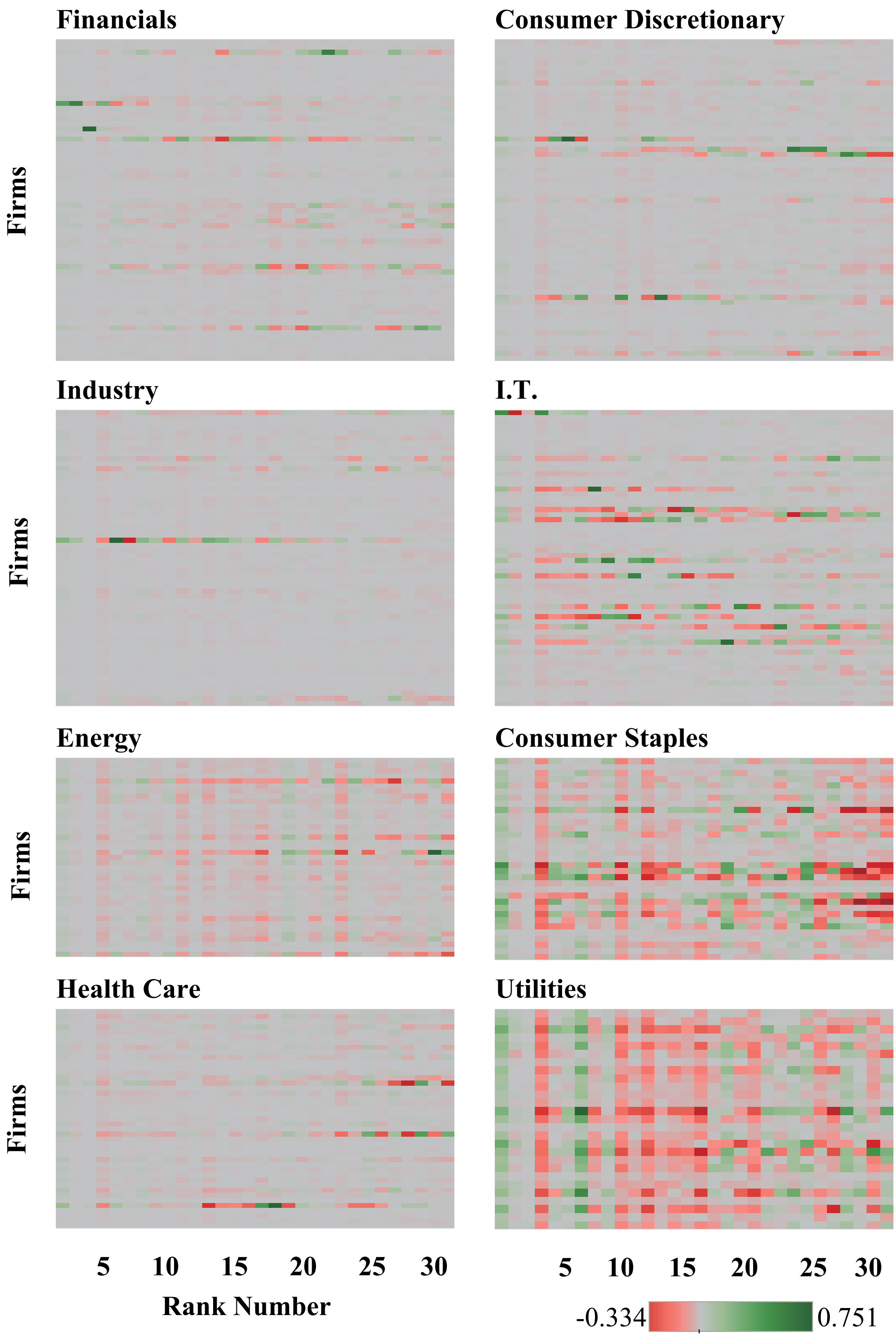


FIGURE 5. Illustration of firm patterns in eight chosen sectors. Each row represents each firm and columns denote principle component number. Red color represents higher value while green one show lower values.

cluster center is ordered descending, where top-K candidates are picked as seed firms for the next step of pairs trading. In this paper, we choose top-100 firms as candidates, as the rest part are not significant at all.

D. PAIRS TRADING

Once we get the seed firms in each sector, we utilize a greedy strategy to pick the potentially pairs firms. In detail, we pick the most significant seed in each iteration, and paired it with

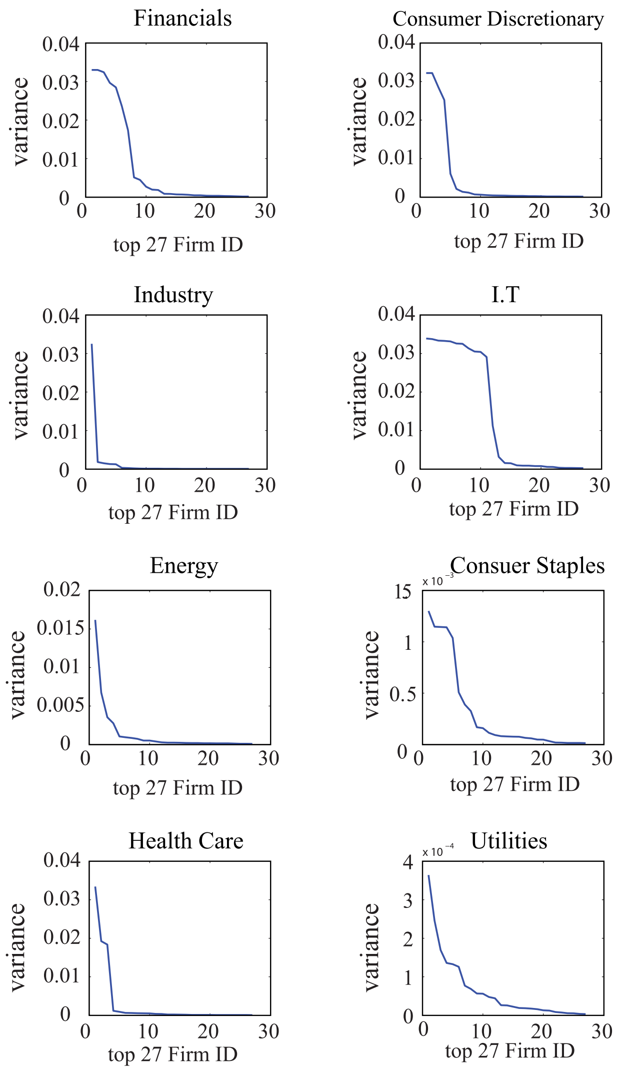


FIGURE 6. Clustering degree for the four chosen sectors of Financials, Consumer Discretionary, I.T., and Industry. *x*-axis shows the firm ID while *y*-axis are the distance.

the next significant once, where the Augumented Dickey-Fuller(ADF) test [17] is applied on the paired firms to test if they are co-integrated. If so, these paired firms were treated as candidates. If not, pick next onces. In this paper, ADF testing is applied on stock data collected from Jan, 2014 to Jan 2015, as we only need to verify the cointegration in the latest time. The trading strategy is employed on data from Jan, 2015 to Dec, 2015.

For comparison, we test our proposed method with four classic approaches in financial literature: correlation, cosine, top variance with correlation (marked as VarCor), top variance with cosine (marked as VarCos). They are also calculated based on historical data from Jan 2004 to Jan 2015. Afterwards, for correlation and cosine methods, we pick them one by one in the order of distance descending to pass the ADF test (from Jan 2014 to Jan 2015). Once passed, the paired equities are fed to pairs trading strategy (on Jan 2015 to Dec 2015). For VarCor and VarCos, we first choose the company with top variance as seed, measure other companies distance with seed and order them descending as a candidate list, pick them one by one to pass the ADF test.

We pick the first 5 successfully paired equities obtained by all the methods. We measure the performance of pairs trading strategy in five popular benchmarks: Largest Loss in A Day, The Number of Event Days [2], Maximum Drawdown [18], [19], Expected Shortfall on 5% [20], Sharpe Ratio and Total Cumulative Return [21]. Here largest loss in a day in the minimal daily return, denoted as $min(x_t)$, where x_t is the return of equity x on day t. Expected Shortfall is the average result obtained when the result is worse than the Value at Risk for the α fraction of best results, for a given distribution $P(x)$, Expected Shortfall is denoted as $\rho_{ES}(\mathrm{P}) = -\frac{1}{1-\alpha}\int_{-\infty}^{P^{-1}(1-\alpha)} x \cdot dP(x)$. Maximum drawdown is the largest single drop from peak to bottom in the value of a portfolio. The Sharpe Ratio measures a portfolio's excess return relative to the total variability of the portfolio [21]. It is formulated as $S = \frac{E(x-x_{\hat{t}})}{\sqrt{var(x-x_{\hat{t}})}}$. Here, $x_{\hat{t}}$ is the index reference return. Cumulative Return $X_t(i)$ from dat 1 to day t is given by $X_t(i) = \prod_{k=1}^{t}(x_t(i)+1) - 1$. For example, the total cumulative return $X_t(i)$ is the overall return of equity i from the beginning to day t.

Table 1 gives the experimental results of the aforementioned five popular benchmarks when using pairs trading strategy when combined with our methods and the other four competing methods. We chose the top five suggested pairs by each method, and apply them with same pairs trading policy. Results are the average of five runs. The results show that our method outperformed all the other algorithms in total cumulative return, which is the key indicator for an investment, while our method also get comparable methods in sharp ratio, expected shortfall and max drawdown.

TABLE 1. Pairs Trading Returns.

Methods	Correlation	Cosine	VarCor	VarCos	TeFTS
◊	-0.04	-0.04	-0.11	-0.10	-0.08
□	-0.03	-0.03	-0.06	-0.05	-0.05
△	-0.12	-0.12	-0.33	-0.28	-0.23
▽	0.08	0.13	0.05	0.13	0.08
◯	0.09	0.08	0.23	0.45	0.91

◊ Largest loss in a day □ Expected Shortfall (5%) △ Max Drawdown
▽ Sharp Ratio ◯ Total Cumulative Return

E. LIMITATIONS

However, there are still some limitations in this experiment. Firstly, since we trade paired equities based on market condition and renews the position ratio daily, which incurs transaction costs consequently. But these costs are ignored in our methods. Secondly, because the main purpose of this paper is to find the paired companies by TeFTS, in order to clarify the effectiveness of our method, we only apply a basic trading strategy as described in section 3.3, which does not include

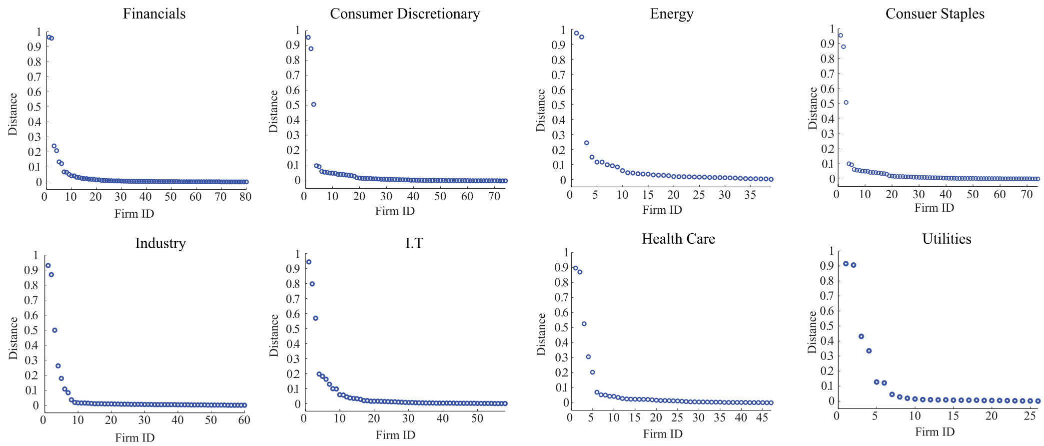


FIGURE 7. **Clustering degree for the four chosen sectors of Financials, Consumer Discretionary, I.T., and Industry. *x*-axis shows the firm ID while *y*-axis are the distance.**

stop-loss strategy (cut the lose and let the winning run) [22]. However, there are still some advanced trading strategy could be applied in future work.

V. DISCUSSION

This study presented a tensor-based framework for similarity measures between paired correlated stocks which attempted to capture the inherent relations among multiple trading sources like firms, time, and the publicly available trade information. First, stocks are represented by high order tensors (multiway arrays) i.e., multiple-modality patterns in the firm-time-trading mode. Second, tensor decomposition method is applied on high order tensors to obtain low-ranked multilinear patterns for each mode, and thus high-dimensional tensors are mapped to low-dimensional representative tensors. Next, the most significant firms are selected using cluster method by combining the captured relations in the other modes. Finally, pairs trading strategy is applied on the picked firms for investment strategy.

We test our farmworker on all constituent firms of $S\&P500$, which are public traded in 12 years, from Jan-01-2004 to Dec-31-2015. The result shows our model consistently compete conventional methods, including correlation, partial correlation, SVD, etc. In particular, when considering the structure of stocks, one interesting phenomenon was observed: not only stocks in all sectors but also stock in each sector are with low-rank information, implying that there are some leading firms influencing the whole sectors.

To explore the reasons why the tensor-based framework can achieve high performance, firstly, it could be the fact that our framework, which are different from the vector or matrix based methods, takes multiple trading sources into consideration. This makes sense, since equity movements are affected by various factors and each factor contains useful information for investment. Thus, the previous studies do not work well since they only used one or two factors. Secondly, the intrinsic relations among different sources can be captured, providing interrelated information for similarity measures. This is interpretable, e.g., relations between upstream and downstream firms can be obtained by tensor decomposition, leading to a low-rank stock data. Thirdly, dimensional reduction in the time and trading modes extract minimum redundant information for similarity measures in the firm mode, they complement each other, both contributing to the superior performance.

Finally, we would like to remark that our tensor-based framework is, although very well suited to stock analysis, a general framework that can be readily applied to other multivariate financial time-series analysis, and can be transferred to some other practical applications.

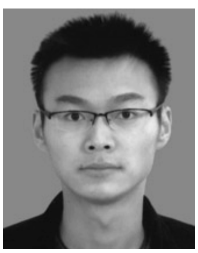
VI. CONCLUSION

The financial time series is strongly affected by various types of highly interrelated information that interact in a complex fashion. Previous studies, which are based on one or two order methods, ignored the interrelations among the various sources. In this paper, a tensor-based framework is proposed for similarity measuring among equities. Multiway financial signals are constructed in tensor form, on which tensor decomposition method is applied. In this process, structural information is preserved which provides supplementary information for pairs selecting and trading algorithms. Experiments on $S\&P500$ datasets demonstrate the superior performance of the proposed algorithm when compared with some state-of-the-art methods.

REFERENCES

[1] Y. Feng *et al.*, "A signal processing perspective on financial engineering," *Found. Trends Signal Process.*, vol. 9, nos. 1–2, pp. 1–231, 2016.
[2] G. Ganeshapillai, J. V. Guttag, and A. Lo, "Learning connections in financial time series," in *Proc. 30th Int. Conf. Mach. Learn. (ICML)*, 2013, pp. 109–117.
[3] G. Vidyamurthy, *Pairs Trading: Quantitative Methods and Analysis*, vol. 217. Hoboken, NJ, USA: Wiley, 2004.
[4] L. François and S. Bruno, "Correlation structure of international equity markets during extremely volatile periods," *J. Finance*, pp. 649–676, 1999.

[5] B. Do, R. Faff, and K. Hamza, "A new approach to modeling and estimation for pairs trading," in *Proc. Financial Manage. Assoc. Eur. Conf.*, 2006, pp. 87–99.
[6] C. Stărică, "Multivariate extremes for models with constant conditional correlations," *J. Empirical Finance*, vol. 6, no. 5, pp. 515–553, 1999.
[7] E. Gatev, W. N. Goetzmann, and K. G. Rouwenhorst, "Pairs trading: Performance of a relative-value arbitrage rule," *Rev. Financial Stud.*, vol. 19, no. 3, pp. 797–827, 2006.
[8] R. J. Elliott, J. Van Der Hoek, and W. P. Malcolm, "Pairs trading," *Quant. Finance*, vol. 5, no. 3, pp. 271–276, 2005.
[9] Q. Zhao, G. Zhou, T. Adali, L. Zhang, and A. Cichocki, "Kernelization of tensor-based models for multiway data analysis: Processing of multidimensional structured data," *IEEE Signal Process. Mag.*, vol. 30, no. 4, pp. 137–148, Jul. 2013.
[10] Q. Zhao *et al.*, "Higher order partial least squares (HOPLS): A generalized multilinear regression method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1660–1673, Jul. 2013.
[11] Q. Zhao, L. Zhang, and A. Cichocki, "A tensor-variate Gaussian process for classification of multidimensional structured data," in *Proc. AAAI*, 2013, pp. 1041–1047.
[12] F. Wang, N. Lee, J. Hu, J. Sun, and S. Ebadollahi, "Towards heterogeneous temporal clinical event pattern discovery: A convolutional approach," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2012, pp. 453–461.
[13] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, Jun. 2014.
[14] Y.-W. Cheung and K. S. Lai, "Lag order and critical values of the augmented Dickey–Fuller test," *J. Bus. Econ. Statist.*, vol. 13, no. 3, pp. 277–280, 1995.
[15] C. F. Baum, "Tests for stationarity of a time series," *Stata Tech. Bull.*, vol. 10, no. 57, pp. 36–39, 2001.
[16] P. Schmidt and J. Lee, "A modification of the Schmidt–Phillips unit root test," *Econ. Lett.*, vol. 36, no. 3, pp. 285–289, 1991.
[17] M. Bock and R. Mestel, "A regime-switching relative value arbitrage rule," in *Proc. Oper. Res.* Berlin, Germany: Springer, 2009, pp. 9–14.
[18] M. Magdon-Ismail, A. Atiya, A. Pratap, and Y. Abu-Mostafa, "The maximum drawdown of the Brownian motion," in *Proc. IEEE Int. Conf. Comput. Intell. Financial Eng.*, Mar. 2003, pp. 243–247.
[19] M. Magdon-Ismail and A. F. Atiya, "Maximum drawdown," *Risk Mag.*, vol. 17, no. 10, pp. 99–102, 2004.
[20] C. Acerbi and D. Tasche, "On the coherence of expected shortfall," *J. Banking Finance*, vol. 26, no. 7, pp. 1487–1503, 2002.
[21] W. F. Sharpe, "The sharpe ratio," *J. Portfolio Manage.*, vol. 21, no. 1, pp. 49–58, 1994.
[22] S.-Y. Shen and A. M. Wang, "On stop-loss strategies for stock investments," *Appl. Math. Comput.*, vol. 119, nos. 2–3, pp. 317–337, 2001.



DAWEI CHENG is currently pursuing the Ph.D. degree with the Key Laboratory of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering, Department of Computer Science and Engineering, Shanghai Jiao Tong University. He was a Senior Software Engineer with Intel Asia Pacific Research and Development Ltd. His research fields include machine learning, financial data mining, knowledge discovery, and time series pattern recognition.

YE LIU received the Ph.D. degree from the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China. His research interests are in tensor theory, data mining, brain signal processing and system modeling, algorithms and computer systems, pattern recognition, and functional brain imaging and modeling.

ZHIBIN NIU received the master's degree from Tianjin University and Shanghai Jiao Tong University, China, and the Ph.D. degree in computer sciences from Cardiff University, U.K. He is currently a Lecturer with the School of Computer Software, Tianjin University, China. His research interests include reverse engineering, data mining, and visual analytics for the financial data. He was a recipient of the Marie SkÅĆodowska-Curie Fellow.

LIQING ZHANG received the Ph.D. degree from Zhongshan University, Guangzhou, China, in 1988. He is currently a Professor with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China. He has authored over 210 papers in international journals and conferences. His current research interests cover computational theory for cortical networks, visual perception and computational cognition, and statistical learning and inference.