Knowledge-Based Systems 89 (2015) 332–345

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Multi-aspect-streaming tensor analysis

Hadi Fanaee-T [a,*], João Gama [b]

[a] Laboratory of Artificial Intelligence and Decision Support, FCUP/University of Porto, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal
[b] Laboratory of Artificial Intelligence and Decision Support, FEP/University of Porto, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal

ARTICLE INFO

Article history:
Received 12 January 2015
Received in revised form 14 July 2015
Accepted 15 July 2015
Available online 21 July 2015

Keywords:
Tensor analysis
Data streams
Online histogram
Tensor decomposition
Streaming tensor analysis

ABSTRACT

Tensor analysis is a powerful tool for multiway problems in data mining, signal processing, pattern recognition and many other areas. Nowadays, the most important challenges in tensor analysis are efficiency and adaptability. Still, the majority of techniques are not scalable or not applicable in streaming settings. One of the promising frameworks that simultaneously addresses these two issues is Incremental Tensor Analysis (ITA) that includes three variants called Dynamic Tensor Analysis (DTA), Streaming Tensor Analysis (STA) and Window-based Tensor Analysis (WTA). However, ITA restricts the tensor's growth only in time, which is a huge constraint in scalability and adaptability of other modes. We propose a new approach called multi-aspect-streaming tensor analysis (MASTA) that relaxes this constraint and allows the tensor to concurrently evolve through all modes. The new approach, which is developed for analysis-only purposes, instead of relying on expensive linear algebra techniques is founded on the histogram approximation concept. This consequently brought simplicity, adaptability, efficiency and flexibility to the tensor analysis task. The empirical evaluation on various data sets from several domains reveals that MASTA is a potential technique with a competitive value against ITA algorithms.

1. Introduction

Tensor decomposition is a powerful technique for the analysis of multiway data in psychometrics, chemometrics, network information systems, pattern recognition and data mining [1]. The growing interest in tensors is due to their capability of discovering complicated patterns in multiway settings that is impossible via other methods. Many techniques are developed for tensor decomposition, but two of the most popular ones are Tucker [2] and PARAFAC [3]. Both of these models suffer from two major issues. Firstly, they are not scalable to large size data sets due to their time/space complexity; and secondly, are not updatable when a new stream of data is retrieved.

The scalability issue is already addressed in three major groups of solutions, including sparse-optimized methods, parallel and distributed techniques and GPU-based solutions. For instance, in [4] a new extension of Tucker decomposition is proposed, called Memory-efficient Tucker (MET) that its space complexity scales up to the non-zero elements in tensor (i.e. O(nz)). In [5] a distributed version of PARAFAC is implemented in MapReduce [6] scaling PARAFAC decomposition up to 100 times for sparse tensors. A different distributed framework is proposed in [7,8] for PARAFAC that divides the tensors into some small sub-tensors and solve sub-tensors problems in different machines. Similar to these works, [9] proposes a parallelized version of PARAFAC called ParCube which is optimized for sparse tensors and provides 14 times acceleration in runtime. In [10] a new method is proposed based on general-purpose computing, on the GPU that operates 360 times faster than the regular PARAFAC decomposition.

Although the above techniques are great tools for dealing with large tensors, they suffer from the non-adaptability problem. This means that when new data is received we have to rebuild the model from scratch. In addition, sparsity-optimized techniques such as MET also do not have any added value for dense tensors, because they only scale up when there is a considerable amount of zero elements in the tensor. Furthermore, the parallelization of tensor decompositions is not as straightforward and it requires extra hardware and software infrastructures.

The pioneer research studies on this problem are those performed by [11–13] who propose some streaming approximation solutions for tensor decomposition in an unified framework called Incremental Tensor Analysis (ITA). The ITA solution, opposed to other scalable decomposition techniques does not need any special infrastructure. It also does not make any restrictive assumption like sparsity. It performs tensor decomposition on each tensor in each time instant, maintains some statistics and then incorporates that for the processing of the next tensor. Therefore, it does not require keeping historical data in the memory. This solution has

* Corresponding author.
E-mail address: hadi.fanaee@fe.up.pt (H. Fanaee-T).

http://dx.doi.org/10.1016/j.knosys.2015.07.013
0950-7051/

two advantages. First, tensor model is easily updatable when new data arrives, and second, the space required for decomposition of the tensor becomes independent of stream length.

The merits of ITA and its usefulness to the analysis of time-evolving tensors are investigated in many studies, so that nowadays, ITA is recognized as the state-of-the-art solution for streaming tensor analysis. However, although ITA allows the tensor to evolve infinitely in time, it makes a restrictive assumption that the dimension of the tensor remains constant during the process. We may not find this limitation annoying for only-time-evolving tensors like network traffic or video streams, when the number of nodes or image frames remain constant during the analysis. But, we may deeply feel this constraint in dealing with multi-aspect-evolving tensors such as social networks, where the number of nodes grows during the evolution of the network. Or in recommendation systems when new users are joined to the system, and size of *user* × *profile* matrix consistently changes. Aside from that, ITA encounters the *intermediate data explosion problem* [5,14] as well as its offline counterparts when the size of the tensor is large.

The intermediate data explosion problem corresponds to the heart of the these techniques, i.e. space-inefficient linear algebra computations that operate directly on the input data. Therefore, in these methods, space efficiency is more influenced by the size of input data rather than the method per se. However, we know that a large portion of tensor decomposition applications is related to *analysis-only* tasks such as anomaly detection (e.g. [15–18]) or simple data analysis (e.g. [19–22]). In such applications, computing the exact subspace of the tensor may not seem mandatory, as opposed to other applications such as compression where the reconstruction of tensor is inevitable. Can we find an alternative adaptive solution for tensor analysis that on one hand avoids space-inefficient computations and on the other hand provides the basic analytical power of tensor decomposition?

We know that histograms are central tools for summarization in data mining. They are also the key technique in image retrieval for measuring similarity between images. Is it possible to extend these ideas to tensor analysis problem? We may find a positive answer for this question, but two more questions will be raised in the following: (a) how do we deal with the huge space/time complexity of histograms while we actually require an efficient method?; (b) is it conceivable to utilize a non-adaptive tool like histogram for solving a streaming problem?

In this research, we tackle these problems by recommending a histogram-based solution that allows the tensor to simultaneously evolve through all modes. We initiate with the description of fundamental concepts such as histograms, tensor segmentation and distribution matching and proceed to develop the first basic approach for histogram-based tensor analysis. Furthermore, we extend the baseline solution to the multi-aspect-streaming scheme (see Fig. 1) by replacing the conventional histogram with a recent incremental approach. To the best of our knowledge the application of histograms in tensor analysis is not reported elsewhere. This is also the first work that addresses the multi-aspect-streaming tensor analysis problem.

The rest of the paper is organized as follows: Section 2 outlines the preliminary concepts. In Section 3 we describe the proposed method. We introduce a new evaluation methodology in Section 4 and later employ it for assessment of the proposal in Section 5. Next, in Section 6 we illustrate the application of the proposed approach on two real case studies. The last section concludes the exposition, presenting the final remarks.

2. Preliminary concepts

Following [23], throughout the paper, scalars are denoted by non-bold lowercase letters (e.g. i), vectors are denoted by boldface lowercase letters (e.g. $\boldsymbol{a}$), matrices are denoted by boldface capital letters (e.g. $\boldsymbol{A}$) and tensors are denoted by Calligraphic letters (e.g. $\boldsymbol{\mathcal{X}}$). In the following we define the necessary concepts required for further description of the proposed methodology. More comprehensive discussion about tensors and their application can be found in survey papers [23,1].

2.1. Tensor

A tensor is a multi-dimensional array and the order of a tensor is the number of dimensions, also known as ways or modes. Vectors, matrices and tensors respectively, are equivalent to first, second and dth order tensor where $d \geqslant 3$.

2.2. Slice

A slice is a *(d-1)*-dimension partition of tensor when an index is fixed in one mode and the indices vary in the other modes. The horizontal, lateral, and frontal slides of a third-order tensor $\boldsymbol{\mathcal{X}}$, are denoted by $X_{i::}, X_{:j:}$, and $X_{::k}$, respectively. Each slice in each mode corresponds to an entity (or feature). For instance, in a three-order tensor of *country* × *year* × *measurement*, the country "Portugal" is a feature in the first mode. The year 2014 is an entity in the second mode and "population" or "GDP" are the features in the third mode.

3. Histogram-based tensor analysis

Histograms are simple statistical tools that have been applied in a wide range of applications [24]. They are simple, non-parametric

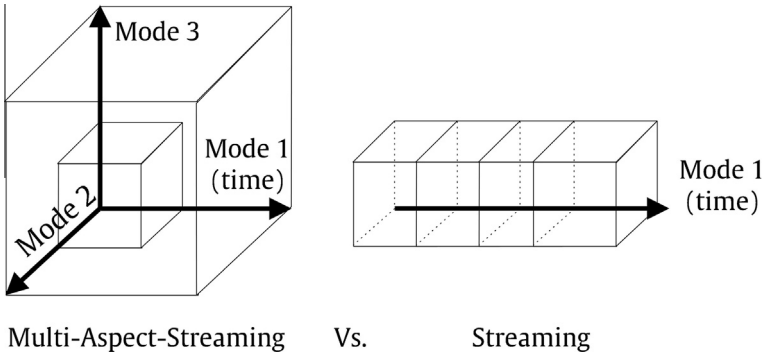

Fig. 1. Comparison of multi-aspect-streaming tensor analysis (proposed) versus streaming tensor analysis (state-of-the-art).