An Adaptive Latent Semantic Analysis for Text mining*

Hong T. Tu, Tuoi T. Phan and Khu P. Nguyen

Abstract—**Latent Semantic Analysis or LSA uses a method of singular value decomposition of co-occurrence document-term matrix to derive a latent class model. Despite its success, there are some shortcomings in this technique. Recent works have improved the standard LSA using method of probability distribution, regularization, sparseness constraint. But there are still some other deficiencies. It is dealt with this paper, an adapted technique called hk-LSA based on reducing dimension of vector space and like-probabilistic relationships between document and latent-topic space is proposed. The adaptive technique overcomes some weak points of LSA such as processing density of orthogonal matrices, complexity in matrix decomposition, facing with alternative iteration algorithms, etc. The experiments show consistent and substantial improvements of the hk-LSA over LSA.**

Keywords—**Latent semantic analysis, convex optimization, regularization, coordinate descent, matrix decomposition**

I. INTRODUCTION

Recently, the topic modeling has seen significant progress in text mining, information retrieval, natural language processing. For mining text from a given collection of n documents, each document must be commonly represented by m unique identifiers such as words, keywords, phrases, etc. called terms. All stop words, e.g. pronouns, verbs, prepositions, auxiliaries are excluded from this set of terms.

Thus, a document is one-to-one corresponding to a tube of m terms. Each term is digitalized by weighting technique, e.g. by a Boolean value specifying whether the term appears in the document, a term frequency, or term frequency-inverse document frequency, etc. So, each tube can be considered as a document vector in the m-dimensional vector space $\Re^m$.

Let D_{i*} be the i^{th} document vector of m weights d_{ij} , j = 1,2,…, m or $D_{i*} = (d_{i1}, d_{i2},\ldots,d_{im})^T \in \Re^m$, i = 1,2,…, n. Hence, the collection of n documents is represented by a matrix **D** of n rows D_{i*}. If the j^{th} columns of **D** is denoted by $D_{*j} = (d_{1j}, d_{2j},\ldots,d_{nj})^T \in \Re^n$, j = 1,2,…, m, then $\mathbf{D} = [D_{1*},D_{2*},\ldots, D_{n*}]^T = [D_{*1},D_{*2},\ldots,D_{*m}] \in \Re^{n\times m}$, the so-called document-term matrix.

Topic modeling refers to algorithms whose aim is to discover a hidden semantic structure in a set of documents. Latent semantic analysis or LSA [1] is one of the most popular applications in information retrieval. In LSA, a high dimensional vector space of documents is transformed to a lower dimensional space or latent topic space for mining text.

By the SVD theorem [2], there exist column orthogonal matrices $\mathbf{U}\in\Re^{n\times n}$, $\mathbf{V}\in\Re^{m\times m}$, and diagonal matrix $\mathbf{S}\in\Re^{n\times m}$ of $r \le \min(n, m)$ positive singular values $\sigma_1 \ge \sigma_2 \ge \ldots \ge \sigma_r > 0$, so that $\mathbf{D} = \mathbf{USV}^T$. If the number of chosen latent topics is k, $k \le r$. LSA applies this matrix factorization to conduct a low rank-k matrix $\mathbf{D}_k$ approximate to **D** as follows

$$\mathbf{D}_k = \mathbf{U}_k\mathbf{S}_k\mathbf{V}_k^T \doteq \mathbf{D} \quad (1)$$

Where $\mathbf{U}_k\in\Re^{n\times k}$ and $\mathbf{V}_k\in\Re^{m\times k}$ are column orthogonal matrices obtained by removing n-k and m-k rightmost columns of **U** and **V**, respectively; $\mathbf{S}_k = \mathrm{Diag}[\sigma_1,\sigma_2,\ldots,\sigma_k]$.

If $\mathbf{T} = \mathbf{S}_k\mathbf{V}_k^T\in\Re^{k\times m}$ then the pseudo-inverse of $\mathbf{T}^T$ is the so-called projection matrix $\Re^m \to \Re^k$ that approximately maps documents in m-dimensional input document space onto corresponding column vectors in k-dimensional latent topic space, k << m. It is obtained

$$\mathbf{T}^{T\dagger} = \mathbf{S}_k^{-1}\mathbf{V}_k^T \quad (2)$$

The approximate equality (1) implies $\mathbf{D}^T \doteq \mathbf{T}^T\mathbf{U}_k^T$, or

$$\mathbf{U}_k^T \doteq \mathbf{T}^{T\dagger}\mathbf{D}^T \quad (3)$$

If $Q = (q_1,q_2,\ldots,q_m)^T\in\Re^m$ is any document vector of m terms, (3) gives its projection $Q' = (q'_1,q'_2,\ldots,q'_k)^T\in\Re^k$ of rank-k in the latent space, where

$$Q' = \mathbf{T}^{T\dagger}Q \quad (4)$$

So, each latent topic component q'_i is described by a linear combination of terms q_j in Q. But this form is not to precisely evaluate topic-term relations. Moreover, to coverage most of topics it is necessary to enlarge k value.

Although there have been lots of efficient applications in information retrieval, LSA is in the face of challenges due to using SVD with orthogonal matrices; time complexity of SVD requires computational cost and storage; a large corpus with millions of documents implies a large number of topics; sometimes not fit for finding relevance document, [3].

It is dealt with this paper a proposed model to the problem of latent semantic analysis, called hk-LSA based on methods of reducing dimension of vector space and nonnegative convex optimization to establish latent-topic space. The hk-LSA overcomes some weak points of traditional LSA such as processing dense and orthogonal matrices of large size, high time complexity with SVD, difficult to processing data collections. The experiments show a well prospect of proposed technique over standard LSA and related others.

*Research supported by ABC Foundation.

Hong T.Tu is with the HCMC University of Technology and Education, a PhD student at the HCMC University of Technology -VNU HCMC; 268, LyThuong Kiet, HCMC, Vietnam (corresponding author to provide phone: +84 908 379 610, e-mail: hongtt@hcmute.edu.vn).

Tuoi T. Phan is with the HCMC University of Technology, VNU HCMC; 268, LyThuongKiet, HCMC, Vietnam (tuoi@cse.hcmut.edu.vn).

Khu P. Nguyen is with the UIT - VNU-HCMC; Ward no. 6, LinhTrung, ThuDuc Dist., HCMC, Vietnam (e-mail: khunp@uit.edu.vn).

The rest of this paper is as follows, the Section II the basic hk-LSA model; Section III illustrates the related works; Section IV the empirical results of the model; and the paper conclusion presents in the Section V.

II. PROPOSED MODEL

A. Approximate to Document Input Space

Using row reduction algorithm to transform **D** into a row echelon form, it is highlighted which subset of h column vectors in **D** is linear independent. This subset forms a h-basis denoted by $\mathbf{B}_h = \{B_1, B_2, \ldots, B_h\}$ of the document space.

In reality, m and n may be very large so h is. By clustering vectors of $\mathbf{B}_h$ into a number of k clusters, k much less than h. Then, for each cluster, a representative vector is chosen. These k representatives are among vectors of $\mathbf{B}_h$, so linear independent and making up a k-basis of a subspace, named the latent topic space. For simplicity, this set is also denoted by the same notation $\mathbf{B}_k = \{B_1, B_2, \ldots, B_k\}$.

Some methods of clustering, e.g. [8] or [9] can be applied. This paper deals with a proposed clustering technique called MSG or Maximum Similarity sub-Graph. This technique is based on MST or Maximum Spanning Tree algorithm with a weighted undirected graph whose vertices are term vectors in $\mathbf{B}_h$ and weights are cosine similarities between term vectors.

The procedure for MSG clustering is as follows. Firstly, find MST of the graph, for each connected triple which is a single vertex with edges running to an unordered pair of other two unconnected neighbor vertices [10], add an edge between the ending vertices of connected triple if similarity between them is significant. Finally, the obtained sub-graph is traversed to detect clusters of vertex-triangles. A demonstration of MSG algorithm is in Fig. 1 below.

After MSG clustering, there are some ways for choosing representative vector or vertex of a cluster. In this paper, as representative of a cluster is chosen so that it is a longest length vector in a couple of vectors having highest similarity.

B. Sparse Optimization Approximate Space

For a k-basis $\mathbf{B}_k$ of the latent topic space, the j^{th} column vector $D_{*j} = (d_{1j}, d_{2j}, \ldots, d_{nj})^T \in \Re^n$ of **D** can be represented as,

$$D_{*j} = \sum_{i=1:k} a_{ij} B_i + \varepsilon_j \quad j = 1,2,\ldots,m \tag{5}$$

Where all ε_j-s are zero-mean noises, $a_{1j}, a_{2j}, \ldots, a_{kj}$ real numbers that need be found so that ε_j-s are minimized for j = 1,2… ,m. Let $A_{*j} = (a_{1j}, a_{2j}, \ldots, a_{kj})^T$, $\mathbf{A} = [A_{*1}, A_{*2}, \ldots, A_{*m}] = (a_{ij}) \in \Re^{k \times m}$ and $B_j = (b_{1j}, b_{2j}, \ldots, b_{nj})^T$, $\mathbf{B} = [B_1, B_2, \ldots, B_k] = (b_{ij}) \in \Re^{n \times k}$, the following non-negative convex optimization problem is obtained

$$\min_{\mathbf{A}} \tfrac{1}{2} \| \mathbf{D} - \mathbf{BA} \|_F^2 \tag{6}$$

For convenience in one-side comparison, it is naturally assumed all columns of **A** non-negative, written briefly as $\mathbf{A} \geq \mathbf{0}$. On the other hand, if **A** is sparse, a great advantage of the interpretation in (5) is to clearly show the most relevant terms for each latent topic and more easily to describe topic-term relationship in a compact form. The sparseness of the projection matrix **A** also saves computational cost and storage requirements. This reason leads to add a regularization term to (6) with ℓ_1-norm as follows,

$$\min_{\mathbf{A} \geq \mathbf{0}} \tfrac{1}{2} \| \mathbf{D} - \mathbf{BA} \|_F^2 + \lambda \|\mathbf{A}\|_1 \tag{7}$$

Where $\|\mathbf{A}\|_1 = \sum_{i=1:k, j=1:m} |a_{ij}| = \sum_{i=1:k, j=1:m} a_{ij}$ and $\lambda \geq 0$ is a regularization parameter which controls the sparseness of **A**, a larger λ leads to a sparse **A**. The solution **A** of (7) implies $\mathbf{BA} \doteq \mathbf{D} \in \Re^{n \times m}$ or $\mathbf{D}^T \doteq \mathbf{A}^T \mathbf{B}^T$. Consequently, there exists a matrix $\mathbf{U}_k \in \Re^{n \times k}$ approximate to **B** so that

$$\mathbf{U}_k^T = \mathbf{A}^{T\dagger} \mathbf{D}^T \doteq \mathbf{B}^T \tag{8}$$

Here $\mathbf{A}^{T\dagger}$ is the pseudo-inverse of $\mathbf{A}^T$. As though the one in (3), **A** conducts the projection matrix $\mathbf{A}^{T\dagger}$ that maps any m-dimensional vector of the document space into a k-dimensional vector of the latent topic space. This projective vector is called latent topic vector.

To solve for **A**, it is noticed that the problem (7) can be decomposed by the vectors A_{*j} into the m following independent problems,

$$\min_{A, j \geq O} \tfrac{1}{2} \| D_{*j} - \mathbf{B} A_{*j} \|_2^2 + \lambda \|A_{*j}\|_1 \quad j = 1, 2, \ldots, m \tag{9}$$

The objective function in (9), denoted by f, is a non-negative convex function of the k non-negative variables a_{ij} :$f(a_{1j}, a_{2j}, \ldots, a_{kj}) = \tfrac{1}{2} \sum_{i=1:n} (d_{ij} - \sum_{\alpha=1:k} b_{i\alpha} a_{\alpha j})^2 + \lambda \sum_{\alpha=1:k} a_{\alpha j}$. It attained a minimum when all its partial derivatives vanish, or

$$\partial f / \partial a_{vj} = \sum_{i=1:n} (d_{ij} - \sum_{\alpha=1:k} b_{i\alpha} a_{\alpha j})(-b_{iv}) + \lambda =$$
$$-\sum_{i=1:n} b_{iv} (d_{ij} - \sum_{\alpha=1:k, \neq v} b_{i\alpha} a_{\alpha j}) + a_{vj} \sum_{i=1:n} b_{iv}^2 + \lambda = 0 \tag{10}$$

Let $\beta_v = \sum_{i=1:n} b_{iv} (d_{ij} - \sum_{\alpha=1:k, \neq v} b_{i\alpha} a_{\alpha j})$, $\chi_v = \sum_{i=1:n} b_{iv}^2 > 0$. Then, (10) gives $a_{vj} \chi_v - \beta_v + \lambda = 0$ or all components a_{vj} must simultaneously satisfy for $v = 1,2,\ldots,k$,

$$\text{if } \beta_v > \lambda \text{ then } a_{vj} = (\beta_v - \lambda)/\chi_v \text{ else } a_{vj} = 0$$

The presence of $a_{\alpha j}$ in β_v, $\alpha \neq v$ and $\alpha = 1,\ldots,k$, conducts solving (10) by the coordinate descent method for each A_{*j}, [11]. To do so, first fix a coordinate v and minimize (9) with respect to v assuming all of the other $a_{\alpha j}$ with $\alpha \neq v$ are given. Then, iteratively solve the problem until all of equations are satisfied simultaneously.

C. Like-probabilistic Relationships

To determine relationship of document D_{*j} by k topics B_i in the probabilistic meaning, all columns of **A** need be normalized or a_{ij} in (10) is replaced with $p_{ij} \stackrel{def}{=} a_{ij} / (\sum_{\alpha=1:k} a_{\alpha j})$. Thus, a projection matrix in this case is conducted by

$$\mathbf{P} = (p_{ij}) = (a_{ij} / (\textstyle\sum_{\alpha=1:k} a_{\alpha j})), \quad 0 \leq p_{ij} \leq 1 \tag{11}$$

By this replacing, D_{*j} in (5) turning to a nonnegative convex combination of $B_1, B_2, \ldots, B_k$. Each p_{ij} can be seen as a probabilistic measure of how frequent occurrence of the topics B_i in the D_{*j}, (8) becomes

$$\mathbf{U}_k^T = \mathbf{P}^{T\dagger} \mathbf{D}^T \tag{12}$$

Like (4), if $Q \in \Re^m$ is any document vector, (12) gives its projection $Q' \in \Re^k$ of rank-k in the latent space,

$$Q' = \mathbf{P}^{T\dagger} Q \tag{13}$$

This is a relation of Q, Q′ in the latent space. Moreover, if the j^{th} components of Q are very small, the corresponding entries of Q′ are near zeros. So, the sparse latent expression of Q′ clearly shows the topics in which Q belongs to.

D. Relevance Scores of Document

Traditional relevance models are almost based on term matching. For a given query vector $Q \in \Re^m$ with the same weighting technique as document vector $D \in \Re^m$, their matching score in high dimension space, named Sim_h, is

$$Sim_h(D,Q) = D^T \cdot Q/(\|D\|_2.\|Q\|_2) \quad (14)$$

Some problems arising from this score due to synonymy or polysemy of terms or words, e.g. 'read a book' and 'book a room'. This issue is not only to cause the term mismatch problem but also lead to low relevance scores.

Based on the transformation $\mathbf{P}^{T\dagger}$ from (11), projections of D and Q into the latent topic space are determined by $D' = \mathbf{P}^{T\dagger}D$ and $Q' = \mathbf{P}^{T\dagger}Q$, respectively. Then, relevance score of D and Q in the topic space is also defined by cosine similarity of D' and Q'. This score is named Sim_k and called topic matching score,

$$Sim_k(D',Q') = D'^T \cdot Q'/(\|D'\|_2.\|Q'\|_2) \quad (15)$$

In recent work [12], the authors improved the search relevance documents by applying probabilistic topic models, such as p-LSA or LDA. One of the advantages of using these model types is to reduce term mismatch. Moreover, if two terms are included in the same topic, then the use of matching score in the topic space can avoid the mismatch problem. That is the reason why these above scores are either incorporated in a form of non-negative convex combination score or in learning to rank score like [13].

III. RELATED WORKS

A. Latent Semantic Analysis

Latent Semantic Analysis or LSA [1] is has widely been used for learning the latent topics from text document to retrieve information. LSA uses the method of the dimension reduction by singular value decomposition or SVD and then projects document-term matrix into a k-dimensional topic space. This projection allows finding relevance documents in the low-dimensional topic space in place of original space. However, it is very difficult to make LSA efficient due to the following reasons:

- Each topic is represented by term features and lacks probabilistic interpretation, so it is difficult to precisely describe the topic-term relationships;
- It is unable to handle polysemy or synonymy of a word or term, that why some mistakes happen in finding appropriate documents, [3];
- Choosing a low rank-k is typically based on ad hoc heuristics, hard to show how small enough with it.

B. Principle Component Analysis

Principle Component Analysis or PCA is closely related to LSA in application for purpose of reducing dimension, [5]. In PCA, the document-term matrix $\mathbf{D}$ is centered by columns, the resulting a matrix is named as $\mathbf{D}_*$. Then, applying SVD on the covariance matrix $\mathbf{C} = (\mathbf{D}_*^T\mathbf{D}_*)/n$ and choosing the first k eigenvalues, $\mathbf{C}$ is approximately factorized as

$$\mathbf{C} \doteq \mathbf{O}\Lambda\mathbf{O}^T \quad (16)$$

Where, $\mathbf{O}$ is an orthogonal matrix, Λ diagonal matrix of k chosen eigenvalues. Hence, a projection matrix similar to (3) is defined so that each given centered document D is projected with image $\mathbf{O}^T D$.

Although in recent years, many variants of PCA [15] have been developed and widely used, however PCA still meet some weak points like standard LSA, in particular making matrices denser, very expensive to compute and store, hard to process with scalability.

C. Probabilistic Latent Semantic Analysis

In probabilistic Latent Semantic Analysis, or p-LSA follows the statistically-oriented point of view to conduct a method for learning the meaning of words in a data-driven approach and identifying different context of word usage, [4]. The p-LSA is based on a latent variable model for co-occurrence data. This dataset relates a set of latent classes $\{z_\alpha \mid \alpha = 1,2,..,k\}$ to observations being occurrence of a word w_j in a specific document d_i. Then, a model for co-occurrence of word and document is defined by a joint probability formulation, as:

$$P(d_i,w_j) = P(d_i)P(w_j|d_i) = P(d_i)\sum_{\alpha=1:k}P(w_j|z_\alpha)P(z_\alpha|d_i) \quad (17)$$

Where, $i = 1,2,\ldots,n$ and $j = 1,2,\ldots,m$. Using the probability multiplication formula, the above implies

$$\begin{aligned} P(d_i,w_j) &= \sum_{\alpha=1:k} P(d_i)P(z_\alpha|d_i)P(w_j|z_\alpha) \\ &= \sum_{\alpha=1:k}P(z_\alpha)P(d_i|z_\alpha)P(w_j|z_\alpha) \end{aligned}$$

Let $\mathbf{D} = (P(d_i,w_j)) \in \Re^{n\times m}$ and $\mathbf{U} = (P(d_i|z_\alpha)) \in \Re^{n\times k}$, $\mathbf{S} = Diag[P(z_1),P(z_2),\ldots,P(z_k)] \in \Re^{k\times k}$, $\mathbf{V} = (P(w_j|z_\alpha)) \in \Re^{m\times k}$. The rightmost sum in (8) leads to a matrix decomposition similar to (1) or (16),

$$\mathbf{D} = \mathbf{U}\mathbf{S}\mathbf{V}^T \quad (18)$$

Comparing with LSA, it is noticed that p-LSA

- performs matrix decomposition using information divergence and relies on the expectation maximization iterative algorithm, so its solution may converge to local optima instead of the global one;
- consumes a vast amount of computational resource and statistical sampling, so it is hard to train the model on a large or very large collection of documents;
- like LSA, it is difficult to determine how many latent classes need enough for a given problem, also the problem of how to make existing topic modeling methods is still open and challenging, so sometimes LSA solution can be used to find an initialization for p-LSA [4].

D. Sparse Latent Semantic Analysis

Sparse Latent Semantic Analysis or s-LSA has recently received a lot of attention in text mining, machine learning community, [6]. This model is motivated by assuming that there is a set of k uncorrelated latent vectors $U_1,U_2,\ldots,U_k$ in $\Re^n$ with $k \le \min(n, m)$ so that $\mathbf{U} = [U_1,U_2,\ldots,U_k] \in \Re^{n\times k}$ is an orthonormal matrix $\mathbf{U}^T\mathbf{U} = \mathbf{I}_k$, an identity matrix of order k. Due to an orthonormal basis of $\Re^n$, any of document vector D_{*j} of m terms is represented approximately as a linear combination with a noise ε_j,

$$D_{*j} = \sum_{i=1:k} a_{ij}U_i + \varepsilon_j \quad j = 1, 2, \ldots, m \quad (19)$$

Based on the dimension of input document space and (19), the larger the k is, the smaller the ε_j or the better the approximation. Let $\mathbf{A} = (a_{ij}) \in \Re^{k \times m}$ be a k×m matrix and $\boldsymbol{\varepsilon}$ is a zero-mean noise column vector of $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_m$. Then, the goal of s-LSA is to compute **U** and **A** given k, so that

$$\min_{\mathbf{U},\mathbf{A}} \tfrac{1}{2} \| \mathbf{D} - \mathbf{UA} \|_F^2 \text{ subject to } \mathbf{U}^T\mathbf{U} = \mathbf{I}_k \qquad (20)$$

At the optimum, **U** and **A** in (20) lead to the best rank-k approximation of the document-term matrix **D**. However, orthonormal condition of **U** makes it denser. To lighten denseness of the **UA**, it is imposed a sparse condition on **A**. Then, a regularization term with entry-wise l_1-norm of **A** is added and (20) becomes

$$\min_{\mathbf{U},\mathbf{A}} \tfrac{1}{2} \| \mathbf{D} - \mathbf{UA} \|_F^2 + \lambda \|\mathbf{A}\|_1 , \mathbf{U}^T\mathbf{U} = \mathbf{I}_k \qquad (21)$$

Where, λ is the non-negative regularization parameter that controls the denseness, a larger λ leads to a sparse **A**. But, if **A** is too sparse some topic-term relationships will be harm, therefore, it must be learnt by case studies to achieve a good performance of the model, [14].

The s-LSA is suitable for many problems in reality, but solving (21) is rather cumbersome. Additionally, how to choose a low rank-k is still leaving open. Moreover, s-LSA is made up of orthogonal matrix and its motivation is not to improve scalability or to utilize the online learning scheme to learn web-scale datasets.

E. Regularized Latent Semantic Analysis

Regularized Latent Semantic Analysis or r-LSA is motivated as s-LSA, but with assumption that a set of k topics $U_{*j} \in \Re^{m \times k}$ exists in the document set, [7]. The matrix **D** approximates to **UV**, $\mathbf{U} = [U_{*1}, U_{*2}, \ldots, U_{*k}] \in \Re^{n \times k}$ the term-topic matrix and $\mathbf{V} = [V_{*1}, V_{*2}, \ldots, V_{*m}] \in \Re^{k \times m}$ topic-document matrix. Furthermore, r-LSA uses ℓ_1-norm to control sparseness of **U** and ℓ_2-norm to regularize shrinkage **V**. Therefore, the problem of r-LSA is

$$\text{Min}_{\mathbf{U},\mathbf{V}} \| \mathbf{D} - \mathbf{UV} \|_F^2 + \lambda_1 \|\mathbf{U}\|_1 + \lambda_2 \|\mathbf{V}\|_2 \qquad (22)$$

Where, λ_1 and λ_2 are non-negative regularization parameters, the larger value of λ_1 the more sparse **U** and the larger λ_2 the larger amount of shrinkage **V**.

Although the optimization problems (21) and (22) are non-convex, but by fixing one variable, the objective function with respect to the other is convex. That why, both s-LSA and r-LSA used the method of splitting to solve their problems separately for **U** then **A** in (21) or **V** in (22) by alternative iterations.

IV. EXPERIMENTS

A. Dataset and dimension of document space

A case study has been done to test the proposed hk-LSA model with a dataset combining with the ones in [1], [16]. This dataset includes n = 26 documents coming from five categories B, C, G, H, M. All terms with respect to each of these documents are italicized in the Table I.

After alphabet sorting, there are m = 29 distinct terms in the document set. By row-echelon reducing, three vectors with respect to terms *Medicine*, *Treat*, *User* may be ignored due to their linear dependence. Hence, h = 26 term columns of **D** are highlighted as linear independent vectors and listed by the following order:

1:*Bread;* 2:*Composition;* 3:*Computer;* 4:*Demonstration;* 5:*Dough;* 6:*Drowsiness;* 7:*Drug;* 8:*Drum;* 9:*Effect;* 10:*EPS;* 11:*Graph;* 12:*Homemade;* 13:*Human;* 14:*Ingredients;* 15:*Interface;* 16:*Make;* 17:*Minors;* 18:*Music;* 19:*Recipe;* 20:*Response;* 21:*Rock;* 22:*Roll;* 23:*Survey;* 24:*System;* 25:*Time;* 26:*Trees*.

At this moment, dimension of the document space is 26 and the document-term matrix **D** is square of size 26.

TABLE I. SET OF DOCUMENTS AND TERMS

Doc.	Terms are italicized
1:B1	How to *Make Bread* and *Roll*s, a *Demonstration*
2:B2	*Ingredient*s for Crescent *Roll*s
3:B3	A *Recipe* for Sour*dough Bread*
4:B4	A Quick *Recipe* for Pizza *Dough* using Organic *Ingredients*
5:B5	Basic *Homemade Bread Recipe*
6:B6	4 Ways to *Make* Delicious *Homemade Bread*
7:C1	*Human* machine *interface* for Lab ABC *computer* applications
8:C2	A *survey* of *user* opinion of *computer system response time*
9:C3	The *EPS user interface* management *system*
10:C4	*System* and *human system* engineering testing of *EPS*
11:C5	Relation of *user*-perceived *response time* to error measurement
12:G1	The generation of random, binary, unordered *trees*
13:G2	The intersection *graph* of paths in *trees*
14:G3	*Graph minors* IV: Widths of *trees* and well-quasi-ordering
15:G4	*Graph minors*: A *survey*
16:H1	*Medicine ingredient* that *make*s you drowsy
17:H2	Common *Drugs* and Medications to *treat Drowsiness*
18:H3	Typhoid fever *treat*ments and *drugs*
19:M1	*Rock* and *Roll Music* in the 1960's
20:M2	Different *Drum Roll*s, a *Demonstration* of Techniques
21:M3	*Drum* and Bass *Composition*
22:M4	A Perspective of *Rock Music* in the 90's
23:M5	*Music* and *Composition* of Popular Bands
24:M6	The *Effect* of *Music* on the *Human* Stress *Response*
25:M7	*Rock Music*, the Star-*System* and the Rise of Consumerism
26:M8	The *effect*s of different *music* genres on physical performance

B. Dimension and Bases of Latent topic Space

Let G be the graph whose vertices are corresponding to h term vectors in $\mathbf{B}_h$. Each edge of G is weighted by the similarity between two vectors of the ending vertices. The MST of G is illustrated in Fig. 1 with solid lines with their cosine similarities in Table II. Based on MST, an extended sub-Graph MSG of G is created by adding edges to connected triples of MST as shown by dotted lines in Fig. 1.

TABLE II. SIMILARITIES AND CLUSTERS OF MSG

Vertex		Cosine similarity	C#	Vertex		Cosine similarity	C#
u	*v*			*u*	*v*		
11	17	0.81649	1	18	***2***	0.28868	3
11	25	0.66667	1	2	8	0.5	3,4
17	23	0.5	1	8	4	0.5	4
23	3	0.5	1,2	4	22	0.70711	4
3	15	0.5	2	4	16	0.40825	4
15	10	0.5	2	16	1	0.57735	4
10	**24**	0.80178	2	1	12	0.70711	4
24	13	0.43644	2	1	19	0.57735	4
24	26	0.43644	2	**19**	5	0.81649	4
26	20	0.66667	2,3	5	14	0.40825	4
20	9	0.40825	3	16	6	0.40825	4,5
9	18	0.57735	3	6	**7**	0.5	5
18	21	0.70711	3	MSG of G, 5 *clusters*			

Nodes with underline belong to both clusters.

After clustering, MSG consists of k = 5 clusters, named C_1, C_2,…,C_5 in Fig. 1. All representative vectors chosen as presented at the end paragraph of Section II.A, are $B_1 = D_{*7}$, $B_{.2} = D_{*11}$, $B_{.3} = D_{*18}$, $B_{.4} = D_{*19}$ and $B_{.5} = D_{.24}$. These vectors form a 5-basis $\mathbf{B}_k$ of the latent topic space. Vertices and terms with respect to these chosen vectors are 7: *Drug*, 11: *Graph*, 18: *Music*, 19: *Recipe* and 24: *System*. In Table II these vertices are in bold faces with arrows in Fig.1.

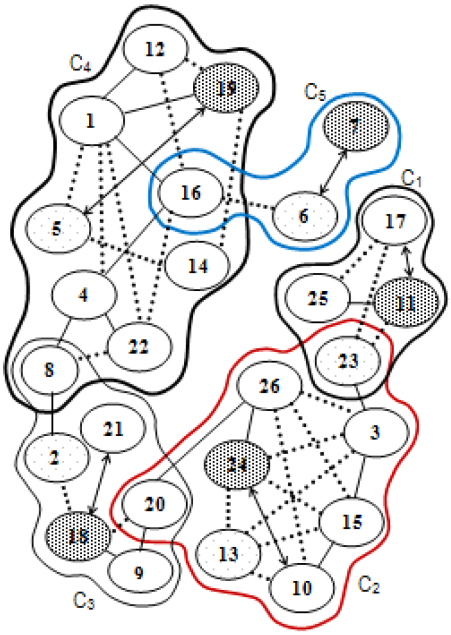

Figure 1. MSG of the term-graph G with five clusters

C. Sparseness of Projection and Similarity Matrices

The projection matrix $\mathbf{T}^{T\dagger} \in \Re^{5\times 29}$ in (2) and its document projections (3) of standard LSA are very dense as illustrated in Table III.(a). By setting $\lambda = 0, 0.01, 0.05$ and basic ways of hk-LSA, the projection matrix $\mathbf{P}^{T\dagger}$ (11) and projections (12) are sparse, as shown in Table III.(b).

By definition, sparseness of a matrix is ratio of number of zero-entries to non-zeros ones. Then hk-LSA model gives the sparseness of the projection matrix onto the latent topic space is about 59%, of the document similarity matrix 50% approximately. While using standard LSA, the sparseness of those matrices is 0%. Therefore, the proposed hk-LSA model increased significantly sparseness of the matrices.

D. Statistical Comparison on Document Similarity Matrices

To compare the document similarity matrices in hk-LSA model and standard LSA, The two-tail statistical testing at a significant level of 2.5% of dependent samples is applied, [17]. First, the main diagonal of these matrices are ignored, the remainder of them is divided in two blocks. The inside block, named i-B, includes similarity entries of between documents in the same category either B or C, G, H, M. The outside of i-B is similarity entries between different categories, called o-B. Each similarity matrix is of size 26×26, hence i-B or o-B consists of 124 or 526 samples, enough for using 2.5%-percentile of t-distribution.

Let t_{obs}(i-B), t_{obs}(o-B) be an observation-calculated t-value of differences between pairs of corresponding similarity entries in i-B or o-B of matrices obtained by the models. Computing shown that t_{obs}(i-B) = 8.388 > $t_{0.975}(123)$ = 1.96. Thus, similarities in i-B calculated by hk-LSA are usually much greater than the ones by LSA. However, t_{obs}(o-B) = −0.409 > $t_{0.025}(525)$ = −1.96 that means similarities in o-B calculated by hk-LSA and LSA are almost the same.

TABLE III. SPARSENESS OF THE PROJECTION DOCUMENT MATRICES

(a). **Projections of the first 15 documents on the 5-dimensional latent topic vector space in standard LSA model**

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	-0.033	-0.022	-0.007	-0.006	-0.008	-0.010	-0.177	-0.510	-0.384	-0.504	-0.232	-0.002	-0.008	-0.014	-0.058
2	0.480	0.206	0.301	0.222	0.315	0.344	-0.042	-0.152	-0.115	-0.104	-0.074	-0.002	-0.008	-0.012	-0.029
3	-0.161	-0.018	-0.222	-0.164	-0.230	-0.223	-0.059	-0.271	-0.199	-0.138	-0.135	-0.008	-0.024	-0.037	-0.070
4	-0.017	-0.013	0.019	0.018	0.017	0.011	0.043	-0.197	0.148	0.271	-0.148	-0.184	-0.419	-0.590	-0.516
5	-0.091	0.078	-0.238	-0.042	-0.254	-0.124	-0.001	0.001	-0.007	-0.007	0.001	0.004	0.010	0.014	0.011

(b). **Projections of the first 15 documents on the 5-dimensional latent topic vector space in hk-LSA model, HN-way**

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0.2	0.2	0.6	0.6	0.6	0.4	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0.196	0.196	0	0	0	0	0.015	0.030	-0.065	-0.001	0.067	0.001	0.002	0.003	-0.002
4	-0.016	-0.016	0	0	0	0	0.396	0.692	0.595	0.545	0.356	-0.009	-0.018	-0.027	0.021
5	0.001	0.001	0	0	0	0	-0.024	0.159	-0.036	-0.033	-0.021	0.287	0.574	0.861	0.772

* Number of zeros in (b) part is 36 or 48% in 15 documents.

E. Latent Topics of Proposed and Standard LSA

A similarity matrix between terms allows finding relationships between terms in dataset. Group of terms with similarities of two terms greater than a given threshold can be considered as the same topic. In the dataset, LSA gives six topic groups with a threshold greater than 90%, Table IV-a. Similarly, with the same threshold hk-LSA gives five topic groups in Table V-b. The terms *Roll*, *Ingredients*, *Survey* in group 1,6 Table V-a move to 3,1,5 of Table V-b, three terms *Demonstration*, *Drum*, *Make* are absent due to similarity between two of them in hk-LSA is less than 0.50.

F. Relevance Scores

For queries, e.g. q_1: *Make bread* at home; q_2: *Treatment* patterns for *drowsiness*; q_3: *Effect* of *rock music* on brain, how to show which documents are appropriate to the queries.

TABLE IV. TOPIC GROUPS FROM MODELS

a. **Standard LSA model**	
Group	**Terms in each topic group**
1	Bread, Demonstration, Dough, Drum, Homemade, Recipe, Roll
2	Drowsiness, Drug, Medicine, Treat
3	Composition, Effect, Music, Rock
4	Computer, EPS, Human, Interface, Response, Survey, System, Time, User
5	Graph, Minors, Trees
6	Ingredients, Make
b. **Proposed hk-LSA model**	
Group	**Terms in each topic group**
1	Bread, Dough, Homemade, Ingredients, Recipe
2	Drowsiness, Drug, Medicine, Treat
3	Composition, Effect, Music, Rock, Roll
4	Computer, EPS, Human, Interface, Response, System, Time, User
5	Graph, Minors, Survey, Trees

* Similarity between any two terms in each topic is greater than 90%

Let $Q_j \in \Re^{29}$ be vector of q_j, j =1, 2, 3. The projection matrices $\mathbf{T}^{T\dagger}$ and $\mathbf{P}^{T\dagger}$ provide projections D'_{i*}, Q'_j of D_{i*}, Q_j and similarity $s_{ij} = Sim_k(D'_{i*},Q'_j)$, i =1, 2,…, 26.

TABLE V. RELEVANCE SCORES IN TOPIC SPACE

	hk-LSA			**Standard LSA**		
Doc	Q_1	Q_2	Q_3	Q_1	Q_2	Q_3
B1	0.71	0	0.7	0.96	0.09	0.15
B2	0.71	0	0.7	0.75	0.48	0.35
B3	1	0	0	0.97	-0.24	-0.14
B4	1	0	0	0.99	0.13	-0.15
B5	1	0	0	0.97	-0.25	-0.13
B6	1	0	0	1	0.	-0.09
C1	0	0	0	0.	0.01	0.10
C2	0	0	0	0.02	0.01	-0.01
C3	0	0	-0.1	0.03	0.01	-0.04
C4	0	0	0	0.00	0.01	0.13
C5	0	0	0.15	0.02	0.01	-0.03
G1	0	0	0	-0.01	-0.01	0.04
G2	0	0	0	-0.01	-0.01	0.03
G3	0	0	0	-0.01	-0.01	0.03
G4	0	0	0	0.	0.	-0.01
H1	0.37	0.93	0	0.40	0.92	-0.04
H2	0	1	0	0.00	1	-0.04
H3	0	1	0	-0.09	1	-0.03
M1	0	0	1	0.13	0.	0.97
M2	0	0	1	0.81	0.07	0.48
M3	0	0	1	0.31	-0.04	0.89
M4	0	0	1	-0.08	-0.04	1
M5	0	0	1	-0.10	-0.04	1
M6	0	0	0.97	-0.11	-0.03	0.92
M7	0	0	0.92	-0.06	-0.03	0.89
M8	0	0	1	-0.12	-0.04	1

* Nnumbers in the form '0.'are not zero, but with magnitude within 10^{-3}

Table V illustrates the relevance scores $(s_{ij}) \in \Re^{26\times3}$ with hk-LSA and standard LSA. Using Table V, the answers to the queries look very clear. But, standard LSA gives high dispersive similarities in comparison with hk-LSA. In this case, answers to queries are the same with the two models, but commonly scattered with LSA.

V. CONCLUSION

In this paper, an adaptive model called hk-LSA has been proposed. This model overcomes difficulties of standard LSA such as time complexity by using SVD, orthogonal matrices, denseness and numerical dispersion. Sparseness in hk-LSA ensures precise and compact form of probabilistic projection to latent topic space. Algorithms in hk-LSA are of low complexity, easy for parallelization. Doing experiments on real word data sets, computing in parallel to improve the scalability of hk-LSA are necessary future works.

ACKNOWLEDGMENT

This research is funded by the HCMUT, VNU-HCM under grant number TNCS-2015-KHMT-40.

REFERENCES

[1] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R., "Indexing by latent semantic analysis," *Jour, Amer. Soc. Info. Sci.*, vol. 4, 1990.

[2] Wikipedia, "Low-rank approximation," https://en.wikipedia.org/wiki/Low-rank_approximation, 2017.

[3] Atreya, A. and Elkan C., "Latent semantic indexing (LSI) fails for TREC collections," *ACM SIGKDD Exp. Newslet*. 12, 2010.

[4] Thomas Hoffmann, "Unsupervised Learning by Probabilistic Latent Semantic Analysis," *Machine Learning*, 42,177-196, Kluwer Acad. Publ., the Netherlands, 2001.

[5] Ayman Farahat, "Improving probabilistic latent semantic analysis using principle component analysis," in *the 7th Conf. of European chapter of the Association for Comp.Linguistics*, EACL, 2006.

[6] Xi Chen , Yanjun Qi , Bing Bai , Qihang Lin , J. G. Carbonell, "Sparse Latent Semantic Analysis," in *repository.cmu.edu*, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.206.2889 , 2010.

[7] Quan Wang, Jun Xu, Hang Li, Nick Craswell, "Regularized Latent Semantic Indexing: A New Approach to Large-Scale Topic Modeling," *ACM Trans. on Info. Systems*, Vol. 31, No. 1, Article 5, Jan. 2013, http://dx.doi.org/10.1145/2414782.2414787, 2013.

[8] Khu Phi Nguyen, Hong Tuyet Tu, "Locality Mutual Clustering for Document Retrieval," *ACM (IMCOM)'14*, Cambodia, 2014.

[9] Eric C. Chi and Kenneth Lange. "Splitting Methods for Convex Clustering," *Human Genetics and Statistics*, Dept. of Bio-math., University of California, Los Angeles, CA 90095-7088. arXiv:1304.0499v2 [stat.ML] 18, Mar 2014.

[10] Reuven Cohen,Shlomo Havlin, *Complex Networks, Structure, Robustness and Function*. Cambridge University Press, the Edinburgh Building, Cambridge CB2 8RU, UK, 2010.

[11] Tong Tong Wu and Kenneth Lange, "Coordinate Descent Algorithms for Lasso Penalized Regression," in *The Annals of App.Stat.*, vol. 2, vo. 1, pp. 224–244, DOI: 10.1214/07-AOAS147, 2008.

[12] Lu, Y., Mei, Q., and Zhai, C., "Investigating task performance of probabilistic topic models: An empirical study of pLSA and LDA," *Information Retrieval*, vol. *14*. 2011.

[13] Burges, C. J., Ragno, R., and Le, Q. V, "Learning to rank with non-smooth cost functions," in *Advances in Neural Information Processing Systems*. MIT Press, Cambridge, MA, 2007.

[14] Rubinstein, R., Zibulevsky, M., Elad, M., "Double sparsity: Learning sparse dictionaries for sparse signal approximation," *IEEE Trans. Signal Process*, vol. 58, Issue 3, pp 1553-1564, NJ, USA, 2010.

[15] R. Zass and A. Shashua, "Nonnegative sparse PCA," in *Advances in Neural Information Processing Systems* (NIPS), 2007.

[16] Dian I. Martin, Michael W. Berry, "Mathematical foundations behind Latent Semantic Analysis," in *Handbook of Latent Semantic Analysis*, Thomas K. Landauer, Danielle S. McNamara, Simon Dennis, Walter Kintsch, Psychology Press, May 13, 2013.

[17] Kandethody M.Ramachandran, Chris P.Tsokos, *Mathematical Statistics with Applications*. San Diego, California, USA: Elsevier Academic Press, 2009, ch7 pp.382-385.