Adv Data Anal Classif (2018) 12:537–558
https://doi.org/10.1007/s11634-017-0280-3

REGULAR ARTICLE

Cluster-based sparse topical coding for topic mining and document clustering

Parvin Ahmadi[1] · **Iman Gholampour**[2] · **Mahmoud Tabandeh**[1]

Received: 3 June 2016 / Revised: 7 February 2017 / Accepted: 13 February 2017 /
Published online: 28 February 2017

Abstract In this paper, we introduce a document clustering method based on Sparse Topical Coding, called Cluster-based Sparse Topical Coding. Topic modeling is capable of improving textual document clustering by describing documents via bag-of-words models and projecting them into a topic space. The latent semantic descriptions derived by the topic model can be utilized as features in a clustering process. In our proposed method, document clustering and topic modeling are integrated in a unified framework in order to achieve the highest performance. This framework includes Sparse Topical Coding, which is responsible for topic mining, and K-means that discovers the latent clusters in documents collection. Experimental results on widely-used datasets show that our proposed method significantly outperforms the traditional and other topic model based clustering methods. Our method achieves from 4 to 39% improvement in clustering accuracy and from 2% to more than 44% improvement in normalized mutual information.

Keywords Document clustering · Topic model · Sparse topical coding · K-means

Mathematics Subject Classification 68T50

✉ Parvin Ahmadi
parvinahmadi@ee.sharif.edu

Iman Gholampour
imangh@sharif.edu

Mahmoud Tabandeh
tabandeh@sharif.edu

[1] Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran

[2] Electronics Research Institute, Sharif University of Technology, Tehran, Iran

1 Introduction

Document clustering (Fritzke 1995; Lamirel 2012; Lamirel et al. 2015) and topic modeling (Hofmann 1999; Blei et al. 2003; Teh et al. 2006; Zhu and Xing 2011) are two widely studied areas with practically many applications. Document clustering aims to organize similar documents into the same groups, which is a crucial building block in document organization, browsing, summarization, classification and retrieval. On the other hand, topic modeling develops probabilistic generative models to discover the latent semantics embedded in document collections and have shown a huge success in text modeling and analysis (Xie and Xing 2013). Recently, topic models have been widely used in many text mining applications, such as document retrieval, summarization, and clustering for different languages.

Probabilistic topic models such as Probabilistic Latent Semantic Analysis (PLSA) (Hofmann 1999), Latent Dirichlet Allocation (LDA) (Blei et al. 2003) and Hierarchical Dirichlet Processes (HDP) (Teh et al. 2006) were first developed to capture latent topics in a large collection of textual documents. Zhu and Xing (2011) presented a non-probabilistic topic model called Sparse Topical Coding (STC). Using STC, the document representations based on topics are "sparse" i.e. each document is described by small number of topics.

Topic modeling and document clustering are highly correlated and can have mutually beneficial relation. Topic models can enhance clustering by discovering the latent semantics embedded in document corpuses. This semantical information can be more useful for the identification of document groups (clusters) than using the raw features. Using topic models, a document corpus is projected into a topic space, in which the noise in measuring similarity is reduced. Moreover, the grouping structure of the corpus can be identified more effectively (Xie and Xing 2013).

In a simple process, it is possible to perform the document clustering and topic modeling tasks separately. We can use topic models to project documents from high dimensional word space into a low dimensional topic space; and then perform a clustering algorithm such as K-means in the topic space (Xie and Xing 2013). However, performing document clustering and topic modeling separately fails to make them to mutually promote each other in order to achieve the best overall performance. For this reason, Wallach (2008) proposed the Cluster based Topic Model (CTM) which integrates document clustering and topic modeling in a unified framework to jointly perform the two tasks. CTM generates the group indicator for each document and then samples the local topic distribution from the Dirichlet prior specific to the selected group. Based on CTM, Xie and Xing (2013) further introduced global topics to capture the global semantics and proposed the Multi-Grain Cluster Topic Model (MGCTM). MGCTM must select between local and global topics, and then generates local or global topics using its choice. Li et al. (2014) developed the Group Latent Dirichlet Allocation (GLDA), which is less complicated than MGCTM. The GLDA model samples the document-topic distributions from a combination of the Dirichlet prior with the selected group's local prior and the global prior. In the vision domain, Wang et al. (2009) proposed three hierarchical Bayesian models to simultaneously learn the model and clusters of documents; namely: the LDA mixture model, the HDP mixture model, and the Dual-HDP model.

As far as we know, all the research conducted so far on integrating document clustering into the topic modeling process is based on probabilistic topic models. However, a limitation of probabilistic topic models is lack of a controlling mechanism to directly tune the sparsity of the inferred representations in a semantic space; which is desirable in text modeling applications (Xie and Xing 2013). In this paper, we develop an extension of the Sparse Topical Coding which integrates document clustering and non-probabilistic topic modeling. We refer to this method as Cluster-based Sparse Topical Coding (CSTC) and employ it for text clustering. In latent semantic space derived by STC, each document is represented by a linear combination of the basic topics. This representation of the documents in the topic space can be utilized as input features for the K-means clustering procedures. In CSTC method, the latent variables of cluster membership, document-topic distribution and topics are jointly inferred. Clustering and modeling are seamlessly connected and mutually promoted. Although we propose the CSTC method for clustering of text documents, it can also be applied to non-textual documents clustering. The major contribution of this paper is to propose a unified sparse topical coding to integrate document clustering and topic modeling together and demonstrating its advantages in text clustering applications. The remainder of the paper is organized as follows: Sect. 2 describes the STC and elaborates the CSTC method for document clustering. The experimental results are presented in Sect. 3. Finally, Sect. 4 concludes the paper.

2 Proposed method

In this section, we introduce CSTC, in which, the latent variables of cluster membership, document-topic distribution and topics are jointly inferred. A brief introduction to this method has been presented in Ahmadi et al. (2015). The main idea of CSTC is that document clustering and topic modeling can be integrated in a unified framework, in order to make two tasks mutually benefit each other and achieve the higher performance.

2.1 Notations

Suppose a collection of D documents $\{\mathbf{w}_1, \ldots, \mathbf{w}_d, \ldots, \mathbf{w}_D\}$ is given where each document contains words from a vocabulary, $\mathbf{v}$ (labeled by indexes), of size N. The dth document, $\mathbf{w}_d$, is simply represented by an $|I_d|$-dimensional vector $\mathbf{w}_d = \{w_{d,1}, \ldots, w_{d,|I_d|}\}$, where I_d is the index set of words that appear in $\mathbf{w}_d$ and the nth entry, $w_{d,n}(n \in I_d)$, denotes the number of appearances of the nth word in the dth document. In topic modeling, a "topic" consists of a group of words that frequently occur together and a "dictionary" refers to all topics. Let $\boldsymbol{\beta} = [\boldsymbol{\beta}_{kn}] \in \mathbb{R}^{K \times N}$, called dictionary, be a matrix with K rows, where each row is assumed to be a topic. We use $\boldsymbol{\beta}_{kn}$ to denote the element of matrix $\boldsymbol{\beta}$ in the kth row and nth column, $\boldsymbol{\beta}_{.n}$ to denote the nth column of $\boldsymbol{\beta}$ and $\boldsymbol{\beta}_{k.}$ to denote the kth row of $\boldsymbol{\beta}$. The kth row of $\boldsymbol{\beta}$, $\boldsymbol{\beta}_{k.}$, represents the kth topic of the dictionary. If P is an $(N\text{-}1)$-simplex, then we can regard $\boldsymbol{\beta}_{k.}$ as a distribution over $\mathbf{v}$ on this simplex. In other words, $\boldsymbol{\beta}_{k.}$ is a positive N-dimensional vector that its elements sum to one. For the dth document $\mathbf{w}_d$, STC projects (maps)

Table 1 Notations of variables

Notations	Descriptions
$d = 1, \ldots, D$	The index of documents
$k = 1, \ldots, K$	The index of topics
$n = 1, \ldots, N$	The index of words
I_d	The set of word indexes in the dth document
$w_{d,n}$	The count of the nth word within the dth document
$\boldsymbol{\beta}$	The dictionary of topics
$\boldsymbol{\theta}_d$	The latent representation for the$d$th document
$\boldsymbol{s}_{d,n}$	The latent representation for the nth word within the dth document
$l_d = 1, \ldots, L$	The group (cluster) indicator of the dth document

$\mathbf{w}_d$ from the low-level space of words into a high-level semantic space spanned by a set of automatically learned topics $\{\boldsymbol{\beta}_{k.}\}_{k=1}^{K}$ and achieves a joint high-level representation of the entire document. STC is a hierarchical latent variable model, where $\boldsymbol{\theta}_d = (\boldsymbol{\theta}_{d,1}, \ldots, \boldsymbol{\theta}_{d,k}, \ldots, \boldsymbol{\theta}_{d,K}) \in \mathbb{R}^{K\times 1}$ is called the document code of dth document and $\boldsymbol{s}_{d,n} = (\boldsymbol{s}_{d,n;1}, \ldots, \boldsymbol{s}_{d,n;k}, \ldots, \boldsymbol{s}_{d,n;K}) \in \mathbb{R}^{K\times 1}$ is the word code of the nth word in the dth document. For clarity, Table 1 presents a summary of key notations used in this paper.

2.2 Sparse topical coding

In probabilistic topic models, each document is represented by a normalized distribution over topics, and each topic is described by a normalized distribution over words. STC relaxes the normalization constraints made in probabilistic topic models. These relaxations lead to nice properties, such as direct control on the sparsity of discovered representations, efficient learning algorithm, and seamless integration with a convex loss function for learning predictive latent representations (Zhu and Xing 2011). In STC, each individual input feature (e.g., a word count) is reconstructed based on a linear combination of topics, and the coefficient vectors (or codes) are un-normalized. Besides, the representation of a given document is derived via an aggregation strategy (e.g., truncated averaging) from the codes of all individual features extracted from that document.

STC assumes that for the dth document, the word codes $\boldsymbol{s}_{d,n}$ are conditionally independent given its document code $\boldsymbol{\theta}_d$, and the observed word counts are independent given their latent representations $\boldsymbol{s}_{d,n}$. With the conditionally independent assumptions, a generative procedure can be summarized as follows:

1. Sample a dictionary $\boldsymbol{\beta}$ from a prior distribution $p(\boldsymbol{\beta})$.
2. For the dth document ($d \in \{1, \ldots, D\}$)
   (a) Sample the document code $\boldsymbol{\theta}_d$ from a prior distribution $p(\boldsymbol{\theta}_d)$.
   (b) For nth observed word ($n \in \boldsymbol{I}_d$)
      i. Sample the word code $\boldsymbol{s}_{d,n}$ from a conditional distribution $p(\boldsymbol{s}_{d,n}|\boldsymbol{\theta}_d)$.

ii. Sample the observed word count $w_{d,n}$ from a conditional distribution $p(w_{d,n}|s_{d,n}, \boldsymbol{\beta}_{.n})$.

According to the generative procedure, a joint probability distribution for $\boldsymbol{\theta}, \boldsymbol{s}, w, \boldsymbol{\beta}$ can be defined as follows:

$$\begin{aligned} p(\boldsymbol{\theta}, \boldsymbol{s}, \mathbf{w}, \boldsymbol{\beta}) &= p(\boldsymbol{\beta}) p(\boldsymbol{\theta}, \boldsymbol{s}, \mathbf{w}|\boldsymbol{\beta}) \\ &= p(\boldsymbol{\beta}) \prod\nolimits_{d\in D} p(\boldsymbol{\theta}_d) \prod\nolimits_{n\in I_d} p(\boldsymbol{s}_{d,n}|\boldsymbol{\theta}_d) p(w_{d,n}|\boldsymbol{s}_{d,n}, \boldsymbol{\beta}_{.n}) \end{aligned} \tag{1}$$

In the STC model, the dictionary $\boldsymbol{\beta}$ is sampled from a uniform distribution. STC assumes that the discrete word counts in the dth document obey a Poisson distribution with $\boldsymbol{s}_{d,n}^T \boldsymbol{\beta}_{.n}$ as the mean parameter, i.e.,

$$p(w_{d,n}|\boldsymbol{s}_{d,n}, \boldsymbol{\beta}_{.n}) = \frac{\left(\boldsymbol{s}_{d,n}^T \boldsymbol{\beta}_{.n}\right)^{w_n} \exp\left(-\boldsymbol{s}_{d,n}^T \boldsymbol{\beta}_{.n}\right)}{w_{d,n}!} \tag{2}$$

In order to achieve sparse representations for document codes $\boldsymbol{\theta}$ and word codes $\boldsymbol{s}$, STC chooses the Laplace prior and the super-Gaussian distribution (Hyvarinen 1999), as:

$$p(\boldsymbol{\theta}_d) \propto \exp(-\lambda_1 \|\boldsymbol{\theta}_d\|_1) \tag{3}$$

$$p(\boldsymbol{s}_{d,n}|\boldsymbol{\theta}_d) \propto \exp\left(-\lambda_2 \left\|\boldsymbol{s}_{d,n}\right\|_1 - \lambda_3 \left\|\boldsymbol{s}_{d,n} - \boldsymbol{\theta}_d\right\|_1\right) \tag{4}$$

Let $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_d, \boldsymbol{s}_d\}_{d=1}^D$ denote the codes for a collection of documents $\{\mathbf{w}_d\}_{d=1}^D$. STC solves the optimization problem:

$$\begin{aligned} \min_{\boldsymbol{\Theta}, \boldsymbol{\beta}} & \sum_{d=1}^{D} \sum_{n=1}^{|I_d|} \left(\boldsymbol{s}_{d,n}^T \boldsymbol{\beta}_{.n} - w_{d,n} \ln\left(\boldsymbol{s}_{d,n}^T \boldsymbol{\beta}_{.n}\right)\right) + \lambda_1 \sum_{d=1}^{D} \|\boldsymbol{\theta}_d\|_1 \\ & + \lambda_2 \sum_{d=1}^{D} \sum_{n=1}^{|I_d|} \left\|\boldsymbol{s}_{d,n}\right\|_1 + \lambda_3 \sum_{d=1}^{D} \sum_{n=1}^{|I_d|} \left\|\boldsymbol{s}_{d,n} - \boldsymbol{\theta}_d\right\|_2^2 \\ s.t. \quad & \boldsymbol{\theta}_d \geq 0, \forall d; \quad \boldsymbol{s}_{d,n} \geq 0, \forall d, n; \boldsymbol{\beta}_{k.} \in P, \forall k \end{aligned} \tag{5}$$

where $\lambda_1, \lambda_2, \lambda_3$ are non-negative hyper-parameters set by users. The L_1-norm will bias towards finding sparse codes $\boldsymbol{\theta}$ and $\boldsymbol{s}$. Minimizing the log-Poisson loss in first part of (5) is actually equivalent to minimizing an un-normalized KL-divergence between observed word counts $w_{d,n}$ and their reconstructions $\boldsymbol{s}_{d,n.}^T \boldsymbol{\beta}_{.n}$ (Zhu and Xing 2011).

2.3 Cluster-based sparse topical coding

Topic models are flexible tools for constructing models in prediction tasks. The motivation behind such models is that the given documents may include unobserved groups of different topics. By incorporating the structure in topic model, we are able to obtain more accurate predictions. We are also interested in uncovering that group structure.

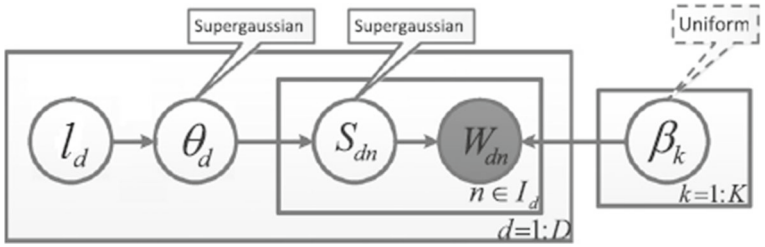

Fig. 1 A graphical representation for the CSTC

This looks like a clustering problem. For example, in the case of modeling textual documents it would be useful to understand the types of documents based on their topics.

To address the clustering problem, we extend STC to the CSTC. In CSTC, closely related documents are clustered together as latent groups (clusters). Each document first selects a group, and then generates the topic distribution with respect to the selected group. Finally, it samples words from the corresponding topic-word distributions. In CSTC, we assume that there is an L-dimensional distribution, $p(l)$, that generates the group indicator $l = 1, \ldots, L$ for the documents. Therefore, the generation process for the dth document can be formulated as follows: A group indicator l_d is chosen from the distribution $p(l_d)$. Then, we sample the document-topic distribution $\boldsymbol{\theta}_d$ with respect to the selected group l_d. The words are then generated as in STC. As shown in Fig. 1, the generative process of CSTC method is as follows:

1. Sample a dictionary $\boldsymbol{\beta}$ from a prior distribution $p(\boldsymbol{\beta})$.
2. For the dth document ($d \in \{1, \ldots, D\}$)
   (a) Sample a cluster l_d from a prior distribution $p(l_d)$.
   (b) Sample the document code $\boldsymbol{\theta}_d$ from a prior distribution $p(\boldsymbol{\theta}_d | l_d)$.
   (c) For nth observed word ($n \in I_d$)
      i. Sample the word code $\boldsymbol{s}_{d,n}$ from a conditional distribution $p(\boldsymbol{s}_{d,n}|\boldsymbol{\theta}_d)$.
      ii. Sample the observed word count w_n from a conditional distribution $p(w_n|\boldsymbol{s}_{d,n}, \boldsymbol{\beta}_{.n})$.

According to the generative procedure, a joint probability distribution can be defined as follows:

$$p(\boldsymbol{\theta}, \boldsymbol{s}, \mathbf{w}, \boldsymbol{\beta}, l) = p(\boldsymbol{\beta}) \prod_{d \in D} p(l_d) p(\boldsymbol{\theta}_d|l_d) \prod_{n \in I_d} p(\boldsymbol{s}_{d,n}|\boldsymbol{\theta}_d) p(w_{d,n}|\boldsymbol{s}_{d,n}, \boldsymbol{\beta}_{.n}) \tag{6}$$

In CSTC, similar to the STC, the dictionary $\boldsymbol{\beta}$ is sampled from a uniform distribution. The discrete word counts follows a Poisson distribution. The word codes $\boldsymbol{s}_{d,n}$ are sampled from the Laplace prior. However, for document codes $\boldsymbol{\theta}$, CSTC chooses the super-Gaussian distribution, as:

$$p(\boldsymbol{\theta}_d|l_d) \propto \exp\left(-\lambda_1 \|\boldsymbol{\theta}_d\|_1 - \lambda_4 \left\|\boldsymbol{\theta}_d - \boldsymbol{\mu}_{l_d}\right\|_2^2\right) \tag{7}$$

where $\boldsymbol{\mu}_{l_d}$ is the mean of $\boldsymbol{\theta}$'s from the cluster l_d.

CSTC formulation for document clustering solves the optimization problem:

$$\begin{aligned}\min_{\boldsymbol{\Theta},\boldsymbol{\beta}} f(\boldsymbol{\Theta};\boldsymbol{\beta}) = \min_{\boldsymbol{\Theta},\boldsymbol{\beta}} \sum_{d=1}^{D}\sum_{n=1}^{|I_d|}\left(w_{d,n} - \boldsymbol{s}_{d,n}^T\boldsymbol{\beta}_{.n}\right)^2 + \lambda_1\sum_{d=1}^{D}\|\boldsymbol{\theta}_d\|_1 \\ + \lambda_2\sum_{d=1}^{D}\sum_{n=1}^{|I_d|}\|\boldsymbol{s}_{d,n}\|_1 + \lambda_3\sum_{d=1}^{D}\sum_{n=1}^{|I_d|}\|\boldsymbol{s}_{d,n} - \boldsymbol{\theta}_d\|_2^2 \\ + \lambda_4\sum_{d=1}^{D}\|\boldsymbol{\theta}_d - \boldsymbol{\mu}_{l_d}\|_2^2 \quad s.t. \quad \boldsymbol{\theta}_d \geq 0, \forall d; \\ \boldsymbol{s}_{d,n} \geq 0, \forall d, n; \quad \boldsymbol{\beta} \geq 0, \sum_{n=1}^{N}\boldsymbol{\beta}_{kn} = 1, \quad \forall k \end{aligned} \tag{8}$$

where $f(\boldsymbol{\Theta};\boldsymbol{\beta})$ denotes the objective function of the optimization problem presented in (8) and $\lambda_1, \ldots, \lambda_4$ are non-negative hyper-parameters which must be set by users. Note that $\boldsymbol{s}$, $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ must be non-negative as well. In other words, the elements of $\boldsymbol{s}$, $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ are non-negative real numbers. In CSTC formulation, unlike STC which minimizes KL-divergence, we minimize the L_2-norm of reconstructions error, which leads to simpler mathematical forms. Figure 2 illustrates another graphical representation of CSTC method.

We have:

$$\sum_{d=1}^{D}\|\boldsymbol{\theta}_d - \boldsymbol{\mu}_{l_d}\|_2^2 = \sum_{l=1}^{L}\sum_{d\in C_l}\|\boldsymbol{\theta}_d - \boldsymbol{\mu}_l\|_2^2 \tag{9}$$

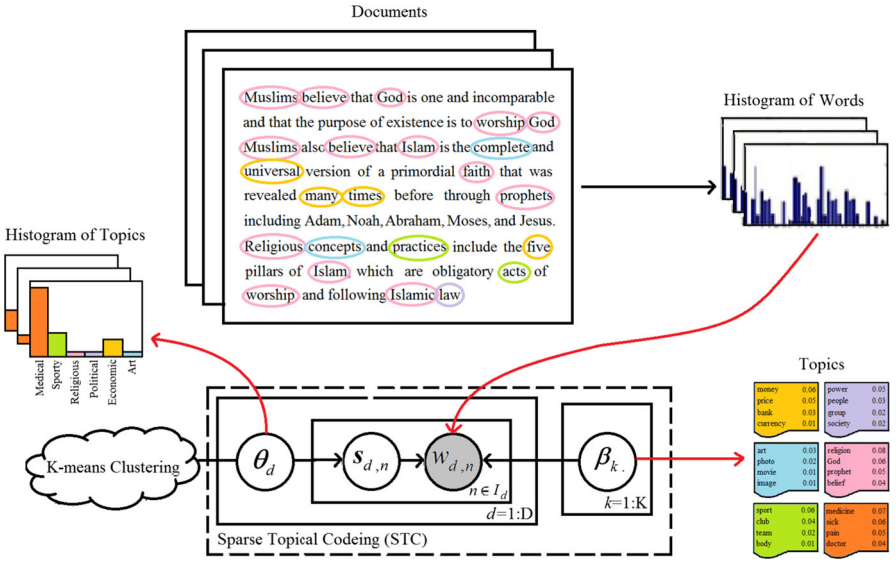

Fig. 2 A graphical representation of our proposed CSTC method for document clustering

where $\boldsymbol{\mu}_l$ is the center of the lth cluster, C_l, for clustering the document codes, $\boldsymbol{\theta}$. Therefore, CSTC can be seen as the STC integrated with a K-means clustering, building a unified framework. This integration encourages the clustering to put documents with the same sparse representation in one cluster. The document code, $\boldsymbol{\theta}$, can be regarded as a representation of the corresponding document in the topic space. Thus, we can treat $\boldsymbol{\theta}$ as input to a K-means clustering process. The centroids of L clusters, $\boldsymbol{\mu}_l, l = 1, \ldots, L$, can be treated as the corresponding hidden variables to sparse document codes, $\boldsymbol{\theta}$'s. An intuitive interpretation of the K-means clustering term is that the document codes, $\boldsymbol{\theta}$, are computed with respect to $\boldsymbol{\mu}_l$.

The objective function $f(\boldsymbol{\Theta}; \boldsymbol{\beta})$ is bi-convex, i.e. f is convex with respect to either $\boldsymbol{\Theta}$ or $\boldsymbol{\beta}$ when the other one is fixed. A typical solution to this problem is provided by the coordinate descent algorithm (Lee et al. 2006), which alternatively applies the optimization to $\boldsymbol{\Theta}$ and $\boldsymbol{\beta}$, as shown in Fig. 3 (Algorithm 1). The procedure of finding the sparse codes (the document codes $\boldsymbol{\theta}$ and the word codes $\boldsymbol{s}$) via optimization over $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_d, \boldsymbol{s}_d\}_{d=1}^{D}$, as specified in (8), is called "sparse coding". In the sparse coding, we actually find the sparsest representations of the documents and words in the current dictionary according to the L_1-norm of $\boldsymbol{\theta}$ and $\boldsymbol{s}$. The procedure of finding the dictionary via the same optimization over $\boldsymbol{\beta}$ is called "dictionary learning". After learning the dictionary of topics $\boldsymbol{\beta}$ and finding the centroids of clusters $\{\boldsymbol{\mu}_c\}_{c=1}^{L}$ through training phase, the cluster of a test document can be defined according to Fig. 4 (Algorithm 2).

A. Sparse coding

The sparse coding step aims to find the codes $\boldsymbol{\Theta}$ when $\boldsymbol{\beta}$ is fixed. The procedure of finding the word codes $\boldsymbol{s}_{d,n}$ via optimization over $\boldsymbol{s}$, as specified in (8), is called "word coding" and the procedure of finding the document codes $\boldsymbol{\theta}_d$ via the same optimization over $\boldsymbol{\theta}$ is called "document coding".

- *Word coding*

When $\boldsymbol{\theta}$ is fixed, $\boldsymbol{s}$ is obtained is obtained by solving the optimization problem:

$$\min_{\boldsymbol{s}} \sum_{d=1}^{D} \sum_{n=1}^{|I_d|} \left(w_{d,n} - \boldsymbol{s}_{d,n}^T \boldsymbol{\beta}_{.n} \right)^2 + \lambda_2 \sum_{d=1}^{D} \sum_{n=1}^{|I_d|} \left\| \boldsymbol{s}_{d,n} \right\|_1 + \lambda_3 \sum_{d=1}^{D} \sum_{n=1}^{|I_d|} \left\| \boldsymbol{s}_{d,n} - \boldsymbol{\theta}_d \right\|_2^2 \quad s.t. \;\; \boldsymbol{s}_{d,n} \geq 0, \forall d, n \tag{10}$$

Due to the conditional independence between the elements of $\boldsymbol{s}$, we can perform this step for each document and each word separately by solving the optimization problem specified with:

$$\min_{\boldsymbol{s}_n} \left(w_n - \boldsymbol{s}_n^T \boldsymbol{\beta}_{.n} \right)^2 + \lambda_2 \left\| \boldsymbol{s}_n \right\|_1 + \lambda_3 \left\| \boldsymbol{s}_n - \boldsymbol{\theta} \right\|_2^2 \quad s.t. \;\; \boldsymbol{s}_n \geq 0, \forall n \tag{11}$$

Fig. 3 Training phase

Algorithm 1:

Inputs: training documents $\{\mathbf{w}_d\}_{d=1}^{D}$, the number of topics K, the number of clusters L, the hyper parameters $\lambda_1,\ldots,\lambda_4$

Outputs: dictionary $\boldsymbol{\beta}$, training document codes $\boldsymbol{\theta}$, training word codes $\boldsymbol{s}$, centroids of clusters $\{\boldsymbol{\mu}_c\}_{c=1}^{L}$

Initialize $\boldsymbol{\beta}$ to a random matrix with non-negative elements chosen from a uniform distribution

Initialize $\{\boldsymbol{\theta}_d, \boldsymbol{s}_d\}_{d=1}^{D}$ randomly with non-negative elements chosen from a uniform distribution

repeat

$\{\boldsymbol{\mu}_c\}_{c=1}^{L} \leftarrow$K-means algorithm on $\{\boldsymbol{\theta}_d\}_{d=1}^{D}$

for d=1:D

$$l = \underset{c=1:L}{\operatorname{argmin}} \left\| \boldsymbol{\theta}_d - \boldsymbol{\mu}_c \right\|_2^2$$

for n=1:$|I_d|$

repeat

for k=1:K

$$\boldsymbol{s}_{d,n;k} = \frac{w_{d,n}\boldsymbol{\beta}_{kn} - \boldsymbol{\beta}_{kn}\sum_{l=1,l\neq k}^{K} \boldsymbol{s}_{d,n;l}\boldsymbol{\beta}_{ln} - 0.5\lambda_2 + \lambda_2\boldsymbol{\theta}_{d,k}}{\boldsymbol{\beta}_{kn}^2 + \lambda_3}$$

$$\boldsymbol{s}_{d,n;k} \leftarrow \max(\boldsymbol{s}_{d,n;k}, 0)$$

end

until convergence

end

for k=1:K

$$\boldsymbol{\theta}_{d,k} = \frac{\lambda_3 \sum_{n=1}^{|I_d|} \boldsymbol{s}_{d,n;k} + \lambda_4 \boldsymbol{\mu}_{l,k} - 0.5\lambda_1}{\lambda_3 |I_d| + \lambda_4}$$

$$\boldsymbol{\theta}_{d,k} \leftarrow \max(\boldsymbol{\theta}_{d,k}, 0)$$

end

end

for n=1:N

$$\boldsymbol{\beta}_n = \left(\sum_{d=1}^{D} \boldsymbol{s}_{d,n} \boldsymbol{s}_{d,n}^T \right)^{-1} \left(\sum_{d=1}^{D} w_{d,n} \boldsymbol{s}_{d,n} \right)$$

end

for k=1:K

$$\boldsymbol{\beta}_{k,} \leftarrow \frac{\boldsymbol{\beta}_{k,}}{\left\| \boldsymbol{\beta}_{k,} \right\|_1}$$

end

until convergence i.e.: $f(\boldsymbol{\Theta}, \boldsymbol{\beta}) < \varepsilon$

Algorithm 5:

Inputs: A test document represented by $\mathbf{w}=(w_1,\ldots,w_N)$, the dictionary $\boldsymbol{\beta}$ and centroids of clusters $\{\boldsymbol{\mu}_c\}_{c=1}^{L}$ learned which have been learned via Algorithm 1, outlier detection threshold *Thr*

Outputs: test document code $\boldsymbol{\theta}$, test word code $\boldsymbol{s}$, the cluster of test document l

Initialize $\boldsymbol{\theta}$ and $\boldsymbol{s}$ elements randomly with non-negative values chosen from a uniform distribution

repeat

$$l = \underset{c=1:L}{\operatorname{argmin}} \left\| \boldsymbol{\theta} - \boldsymbol{\mu}_c \right\|_2^2$$

for n=1:$|I|$

repeat

for k=1:K

$$\boldsymbol{s}_{n;k} = \frac{w_n \boldsymbol{\beta}_{kn} - \boldsymbol{\beta}_{kn} \sum_{l=1, l\neq k}^{K} \boldsymbol{s}_{n;l} \boldsymbol{\beta}_{ln} - 0.5\lambda_2 + \lambda_3 \boldsymbol{\theta}_k}{\boldsymbol{\beta}_{kn}^2 + \lambda_3}$$

$$\boldsymbol{s}_{n;k} \leftarrow \max(\boldsymbol{s}_{n;k}, 0)$$

end

until convergence

end

for k=1:K

$$\boldsymbol{\theta}_k = \frac{\lambda_3 \sum_{n=1}^{|I_d|} \boldsymbol{s}_{n;k} + \lambda_4 \boldsymbol{\mu}_{l,k} - 0.5\lambda_1}{\lambda_3 |I| + \lambda_4}$$

$$\boldsymbol{\theta}_k \leftarrow \max(\boldsymbol{\theta}_k, 0)$$

end

until convergence

if $\sum_{n=1}^{|I|} (w_n - \boldsymbol{s}_n^T \boldsymbol{\beta}_{.n})^2 + \lambda_1 \left\| \boldsymbol{\theta} \right\|_1 + \lambda_2 \sum_{n=1}^{|I|} \left\| \boldsymbol{s}_n \right\|_1 + \lambda_3 \sum_{n=1}^{|I|} \left\| \boldsymbol{s}_n - \boldsymbol{\theta} \right\|_2^2 + \lambda_4 \left\| \boldsymbol{\theta} - \boldsymbol{\mu}_l \right\|_2^2 > Thr$

Test document is an outlier

else

Cluster of test document = l

Fig. 4 Test phase

Due to the non-negativity constraint on $\boldsymbol{s}_n = (\boldsymbol{s}_{n;1}, \ldots, \boldsymbol{s}_{n;k}, \ldots, \boldsymbol{s}_{n;K})$, the L_1-norm can be written as:

$$\|\boldsymbol{s}_n\|_1 = \sum_{k=1}^{K} \left|\boldsymbol{s}_{n;k}\right| = \sum_{k=1}^{K} \boldsymbol{s}_{n;k} \tag{12}$$

Substituting (12) in (11), each $\boldsymbol{s}_{n;k}$ can be calculated iteratively via:

$$\boldsymbol{s}_{n;k} = \arg\min_{\boldsymbol{s}_{n;k}} \left\{ \left(w_n - \boldsymbol{s}_n^T \boldsymbol{\beta}_{.n} \right)^2 + \lambda_2 \sum\nolimits_{l=1}^{K} \boldsymbol{s}_{n;l} + \lambda_3 \|\boldsymbol{s}_n - \boldsymbol{\theta}\|_2^2 \right\} \tag{13}$$

Setting the gradient to zero yields:

$$\boldsymbol{s}_{n;k} = \frac{w_n \boldsymbol{\beta}_{kn} - \boldsymbol{\beta}_{kn} \sum_{l=1, l\neq k}^{K} \boldsymbol{s}_{n;l} \boldsymbol{\beta}_{ln} - 0.5\lambda_2 + \lambda_3 \boldsymbol{\theta}_k}{\boldsymbol{\beta}_{kn}^2 + \lambda_3} \tag{14}$$

Non-negativity constraint on $\boldsymbol{s}$ is imposed according to Proposition 1 in Zhu and Xing (2011) as $\boldsymbol{s}_{n;k} \leftarrow \max(\boldsymbol{s}_{n;k}, 0)$. Proposition 1 proposes a method to impose the non-negativity constraint in the convex optimization problems. According to this proposition, if $h(x)$ is a strictly convex function, the optimum solution x^* of the constrained problem $P_0 : \min_{x\geq 0} h(x)$ is $x^* = \max(0, x_0)$, where x_0 is the solution of the unconstrained problem $P_1 : \min_x h(x)$.

- *Document coding*

When $\boldsymbol{s}$ is fixed, $\boldsymbol{\theta}$ is obtained is obtained by solving the optimization problem:

$$\begin{aligned} \min_{\boldsymbol{\theta}} \lambda_1 \sum_{d=1}^{D} \|\boldsymbol{\theta}_d\|_1 + \lambda_3 \sum_{d=1}^{D} \sum_{n=1}^{|I_d|} \left\|\boldsymbol{s}_{d,n} - \boldsymbol{\theta}_d\right\|_2^2 \\ + \lambda_4 \sum_{l=1}^{L} \sum_{d\in C_l} \left\|\boldsymbol{\theta}_d - \boldsymbol{\mu}_l\right\|_2^2 \quad s.t. \;\; \boldsymbol{\theta}_d \geq 0, \forall d \end{aligned} \tag{15}$$

The optimization formula (15), can be further simplified for a specific document and any chosen cluster as:

$$\min_{\boldsymbol{\theta}} \lambda_1 \|\boldsymbol{\theta}\|_1 + \lambda_3 \sum_{n=1}^{|I|} \|\boldsymbol{s}_n - \boldsymbol{\theta}\|_2^2 + \lambda_4 \left\|\boldsymbol{\theta} - \boldsymbol{\mu}_l\right\|_2^2 \quad s.t. \;\; \boldsymbol{\theta} \geq 0 \tag{16}$$

Due to the non-negativity constraint on $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_k, \ldots, \boldsymbol{\theta}_K)$, the L_1-norm can be simplified as:

$$\|\boldsymbol{\theta}\|_1 = \sum_{k=1}^{K} |\boldsymbol{\theta}_k| = \sum_{k=1}^{K} \boldsymbol{\theta}_k \tag{17}$$

Since different dimensions of $\boldsymbol{\theta}$ are not coupled to each other, each $\boldsymbol{\theta}_k$ is calculated separately. Substituting (17) in (16), each $\boldsymbol{\theta}_k$ can be expressed in terms of the optimization formula in:

$$\boldsymbol{\theta}_k = \arg\min_{\boldsymbol{\theta}_k} \left\{ \lambda_1 \boldsymbol{\theta}_k + \lambda_3 \sum_{n=1}^{|I|} (\boldsymbol{s}_{n;k} - \boldsymbol{\theta}_k)^2 + \lambda_4 (\boldsymbol{\theta}_k - \boldsymbol{\mu}_{l,k})^2 \right\} \tag{18}$$

Setting the gradient to zero yields:

$$\boldsymbol{\theta}_k = \frac{\lambda_3 \sum_{n=1}^{|I|} \boldsymbol{s}_{n;k} + \lambda_4 \boldsymbol{\mu}_{l,k} - 0.5\lambda_1}{\lambda_3 |I| + \lambda_4} \tag{19}$$

Non-negativity constraint on $\boldsymbol{\theta}$ is imposed according to Proposition 1 in Zhu and Xing (2011) as $\boldsymbol{\theta}_k \leftarrow \max(\boldsymbol{\theta}_k, 0)$.

B. Dictionary learning

The dictionary learning step aims to find the $\boldsymbol{\beta}$ when $\boldsymbol{\Theta}$ is fixed. After finding all document codes and word codes of the collection, the dictionary $\boldsymbol{\beta}$ is updated by minimizing:

$$\min_{\boldsymbol{\beta}} \sum_{d=1}^{D} \sum_{n=1}^{|I_d|} \left(w_{d,n} - \boldsymbol{s}_{d,n}^T \boldsymbol{\beta}_{.n} \right)^2 \quad s.t. \quad \boldsymbol{\beta} \geq 0, \sum_{n=1}^{N} \boldsymbol{\beta}_{kn} = 1, \forall k \tag{20}$$

By assigning zero values to $w_{d,n}$ and $\boldsymbol{s}_{d,n}$ for $n \notin |I_d|$, we can replace $|I_d|$ with N. Then (20) can be written as:

$$\sum_{n=1}^{N} \left(\sum_{d=1}^{D} \left(w_{d,n} - \boldsymbol{s}_{d,n}^T \boldsymbol{\beta}_{.n} \right)^2 \right) s.t. \quad \boldsymbol{\beta} \geq 0, \sum_{n=1}^{N} \boldsymbol{\beta}_{kn} = 1, \forall k. \tag{21}$$

For the nth column of $\boldsymbol{\beta}$, $n = 1, \ldots, N$, we have:

$$\boldsymbol{\beta}_{.n} = \arg\min_{\underline{\boldsymbol{\beta}}} \left\{ \sum_{d=1}^{D} \left(w_{d,n} - \boldsymbol{s}_{d,n}^T \underline{\boldsymbol{\beta}} \right)^2 \right\}. \tag{22}$$

Equation (22) is a convex optimization problem that can be efficiently solved by setting the gradient to zero. Thus, the nth column of $\boldsymbol{\beta}$ is calculated as:

$$\boldsymbol{\beta}_{.n} = \left(\sum_{d=1}^{D} \boldsymbol{s}_{d,n} \boldsymbol{s}_{d,n}^T \right)^{-1} \left(\sum_{d=1}^{D} w_{d,n} \boldsymbol{s}_{d,n} \right). \tag{23}$$

3 Experimental results

In this section, experimental evaluation is performed on two popular text datasets and the performance is compared with traditional and some topic model based clustering methods.

3.1 Datasets

The 20-Newsgroups and WebKB datasets[1] are two popular English datasets for experiments in text applications of machine learning techniques, such as text classification and text clustering. In this paper, experiments for text clustering are conducted on these datasets. The 20-Newsgroups dataset is a collection of 18,744 documents with 61,188 distinct words. It contains 11,269 (60%) documents for training and 7505 (40%) documents for testing. This dataset is equally divided into 20 categories, each with around 1000 documents. The WebKB dataset is a collection of 4199 documents and consists of 4 categories. In contrast to 20-Newsgroups, it is an unbalanced dataset, where the largest category contains 1641 documents and the smallest category contains only 504 documents. It contains 2803 (about 67%) documents for training and 1396 (about 33%) documents for testing. Datasets are pre-processed with stop-word removal and stemming. In our experiments, the input cluster number required by clustering algorithms is set to the number of categories in the dataset ground truth which is equal to 20 for 20-Newsgroups dataset and 4 for WebKB dataset.

3.2 Document sparsity

Different from the probabilistic topic models like LDA, CSTC method is a non-probabilistic topic model like STC, which directly controls the sparsity of inferred documents representations as a mixture of topics. By imposing a sparse bias on the document codes $\boldsymbol{\theta}$, a document can be sufficiently reconstructed by a few topics of the dictionary.

The document sparsity is defined as the sparsity ratio of learned document codes and is calculated based on proportion of entries in the D document codes $\boldsymbol{\theta}_d \in \mathbb{R}^K$, i.e.:

$$\text{Document sparsity} = \frac{\#\text{real number zero of } \{\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_d, ..., \boldsymbol{\theta}_D\}}{K \times D}. \tag{24}$$

Figure 5 depicts that both STC and CSTC methods can discover sparse document codes. This means that each document belongs to just a limited number of topics. But, due to exploiting the clustering based regularization term in CSTC, a sparser representation for $\boldsymbol{\theta}$ can be achieved.

[1] http://web.ist.utl.pt/~acardoso/datasets/.

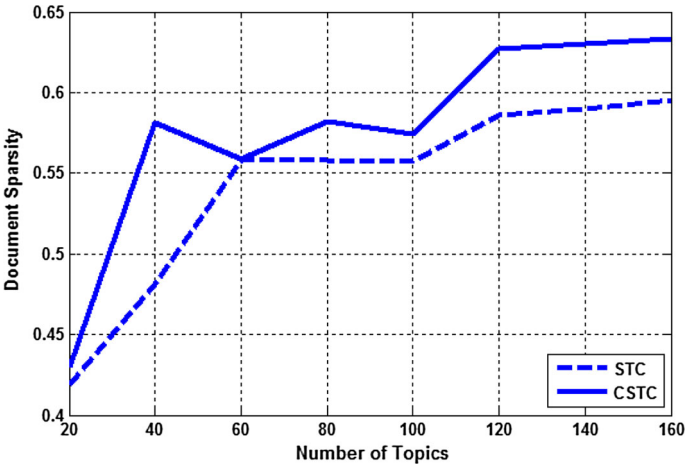

Fig. 5 Sparsity comparison of document codes learned by STC and CSTC methods using 20-Newsgroups dataset

3.3 Clustering evaluation metrics

We use two common evaluation metrics to measure the clustering performance: clustering accuracy and Normalized Mutual Information (NMI). For both metrics, a larger score represents a better performance.

The clustering accuracy is evaluated by comparing the recognized label of each document with the labels provided by the ground truth. To define the cluster labels, Kuhn–Munkres algorithm (Kuhn 1955) is used to find which cluster gives a maximal match to a ground truth class. Kuhn-Munkres algorithm finds the best permutation mapping of each cluster label to the equivalent label from the ground truth. Given a document $\mathbf{w}_d$, let $\tilde{c}_d$ and c_d be the extracted cluster label and the label provided by the ground-truth, respectively. The clustering accuracy is defined as follows:

$$\text{Clustering Accuracy} = \frac{\sum_{d=1}^{D} \delta(c_d, \tilde{c}_d)}{D} \tag{25}$$

where D is the total number of documents and δ (x,y) is the delta function that equals one if $x = y$ and zero otherwise.

NMI measures the similarity between two label sets of the same data. Let $\tilde{C}$ be the set of clusters obtained by the clustering algorithm and C be the set of clusters obtained from the ground truth. Their NMI is defined as:

$$\text{NMI}(\tilde{C}, C) = \frac{\text{MI}(\tilde{C}, C)}{\max(\text{H}(\tilde{C}), \text{H}(C))}. \tag{26}$$

In (26), MI and H define the mutual information and the entropy, respectively, that are calculated as follows:

$$\mathrm{MI}(\tilde{C}, C) = \sum_{i=1}^{L}\sum_{j=1}^{L} p(\tilde{C}_i, C_j)\log\frac{p(\tilde{C}_i, C_j)}{p(\tilde{C}_i)p(C_j)} \tag{27}$$

$$\mathrm{H}(\tilde{C}) = -\sum_{i=1}^{L} p(\tilde{C}_i)\log p(\tilde{C}_i) \tag{28}$$

$$\mathrm{H}(C) = -\sum_{i=1}^{L} p(C_i)\log p(C_i) \tag{29}$$

where L is the number of clusters that is set to the number of categories in the dataset ground truth; $p(\tilde{C}_i)$ and $p(C_i)$ denotes the probabilities that a document belongs to the cluster $\tilde{C}_i$ and cluster C_i, respectively; $p(\tilde{C}_i, C_i)$ is the joint probability that a document belongs to the cluster $\tilde{C}_i$ and cluster C_i at the same time. It is easy to check that $\mathrm{NMI}(\tilde{C}_i, C_i)$ ranges from 0 to 1. $\mathrm{NMI} = 1$ the two sets of clusters are identical, and $\mathrm{NMI} = 0$ if the two sets are independent.

3.4 Hypothesis test for statistical significance evaluation of the clustering results

In order to determine whether the difference between the document clustering results of two methods is significant, we perform hypothesis testing (Papoulis and Pillai 2002). Let X_1 and X_2, where $X_1 > X_2$, be the number of documents that are correctly clustered using the methods A and B, respectively and D be the total number of test documents (i.e., X_1/D and X_2/D are the clustering accuracies). Also, let P_1 and P_2 be the actual probabilities of correctly clustering a test document using the methods A and B, respectively. We wish to test the assumption that $P_1 - P = 0$ (the null hypothesis) against the assumption that $P_1 > P_2$ (the alternative hypothesis). That is, we wish to test whether the numbers of truly clustered documents X_1 and X_2 for the given number of test documents D support the rejection of the null hypothesis. If these numbers support the rejection of the null hypothesis, we conclude that the difference between the clustering results is significant.

The number of truly clustered documents X for a given method can be viewed as a random variable with binomial distribution, where the probability of success P is the probability of correctly clustering a test document and the number of trials is equal to the total number of test documents D. We know that, for a sufficiently large sample size D, the binomial distribution converges to the normal distribution $N(DP,DP(1-P))$. Let X_1 and X_2 be two random variables with binomial distributions with probabilities of success P_1 and P_2, respectively and the numbers of trials D. It can be shown that, under the assumption that $P_1 = P_2$,

$$q = \frac{X_1 - X_2}{\sqrt{\frac{(X_1+X_2)(2D-X_1-X_2)}{2D}}} \tag{30}$$

has approximately a standard normal distribution (i.e., $N(0,1)$). We use q as the test statistic for our hypothesis testing.

In the hypothesis testing, we find a region on the real line where under the null hypothesis, the density of the test statistic is negligible. This region is called critical region of the test. If the observed test statistic falls in the critical region, we reject the null hypothesis. The critical region is obtained according to the chosen level of significance. For 95% level of significance, the critical region for our hypothesis test is $q > z_{.95}$, where $z_{.95}$ is the standard normal percentile and is equal to 1.64.

3.5 Clustering experiments

In order to study how topic modeling affects document clustering, we compare CSTC method with two topic model based methods. The first one uses STC to learn a code vector for each document; then performs K-means on document codes to obtain clusters. We use "STC K-means" to refer to this method. The second one, similar to the method proposed in Lu et al. (2011) for PLSA and LDA, treats each topic as a cluster. Document code θ can be deemed as a mixture proportion vector over clusters and can be utilized for clustering. A document is assigned to cluster x if $x = \text{argmax}_j \boldsymbol{\theta}_j$. We refer to the second method by "STC".

In CSTC method, the hyper-parameters $\lambda_1, \ldots, \lambda_4$ should be tuned to achieve the best clustering performance. We set the hyper-parameters as $\lambda_1 = \lambda_3 = 0.5, \lambda_2 = 0.2, \lambda_4 = 5$ for both 20-Newsgroups and WebKB datasets. Details of selecting appropriate values for hyper-parameters are discussed in Sect. 3.6. In STC method, the number of topics (K) should be set to the number of clusters that is 20 for 20-Newsgroups dataset and 4 for WebKB dataset. For STC K-means and CSTC, the number of topics could be set to the number of clusters or even higher.

Figure 6 shows the clustering accuracy versus the number of topics and Fig. 7 depicts the NMI with respect to the number of topics, for these methods. According to these figures, the best results, i.e. the highest clustering accuracy and NMI, are achieved by using 120 topics in STC K-means and 100 topics in CSTC. In STC K-means and CSTC methods, the number of topics defines the dimension of input feature vectors of the K-means and has an important impact on accuracy. Generally, accuracy increases with the number of topics in a certain range and then begins to decrease. The phenomenon of accuracy decline is recognized as over-fitting and is a direct result of the curse of dimensionality. The larger value at which the over-fitting starts, the model can handle the larger number of latent topic features. On one hand, a large number of topics increase the possibility of over-fitting; on the other hand, it provides more latent features for building the K-means.

Tables 2 and 3 summarize the clustering accuracy and NMI values for different clustering methods using 20-Newsgroups and WebKB datasets, respectively. For STC K-means as well as CSTC, in which clustering accuracy and NMI values are changed versus the number of topics, the best results are considered. The results of traditional clustering methods including K-means, Normalized Cut (NC), Non-negative Matrix Factorization (NMF) and other topic model based methods [LDA (Lu et al. 2011); CTM (Wallach 2008); MGCTM (Xie and Xing 2013); GLDA (Wallach 2008); LDA mixture model (Wang et al. 2009)] as performance baselines are also reported in Tables 2 and 3. In NC, we use Gaussian kernel as the similarity measure between

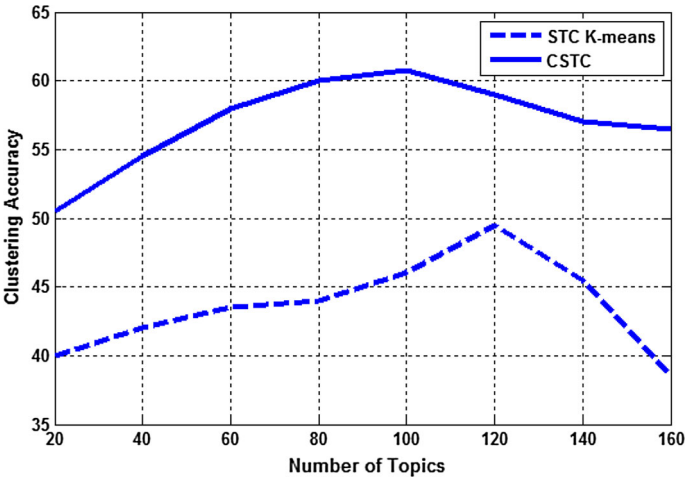

Fig. 6 Clustering accuracy (%) versus the number of topics on the 20-Newsgroups dataset

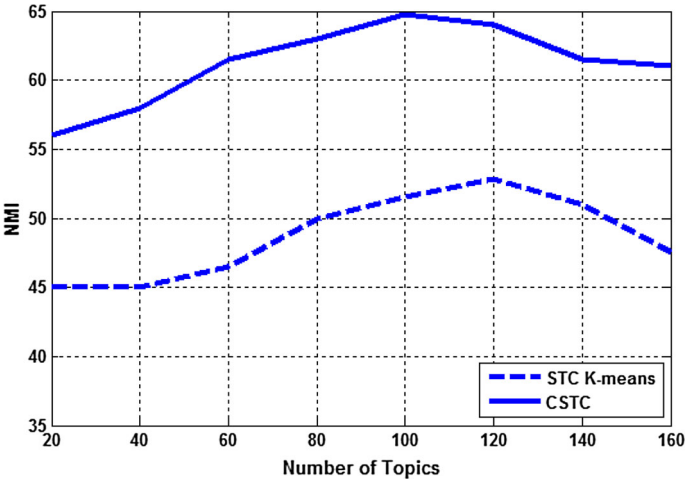

Fig. 7 NMI (%) versus the number of topics on the 20-Newsgroups dataset

documents. The bandwidth parameter is set to 10. In LDA, we use symmetric Dirichlet priors α and β to draw document-topic distribution and topic-word distribution, setting to 0.1and 0.01 respectively. For the CTM, we set the number of topics to 120 for the 20-Newsgroups dataset and to 40 for the WebKB dataset. In MGCTM and GLDA, we used 10 local topics for each group and 20 global topics for the 20-Newsgroups dataset, and 32 local topics for each group and 32 global topics for the WebKB dataset. As illustrated in Tables 2 and 3, our proposed CSTC method achieves the highest accuracy and NMI in the task of textual document clustering. It performs much better than the traditional approaches (i.e., K-means, NC and NMF) and performs better compared to other topic model based clustering methods (like STC, STC K-means, LDA, CTM, MGCTM, GLDA and LDA mixture model). The results show that CSTC outperforms

Table 2 Evaluation of different clustering methods on the 20-Newsgroups dataset

Clustering method	Clustering accuracy (%) (q value: reject/accept)	NMI (%)
CSTC	**60.75**	**64.78**
STC K-means	49.53 (13.82: Reject)	52.82
STC	55.87 (6.06: Reject)	59.58
K-means	33.65 (34.22: Reject)	31.54
NC	22.03 (47.51: Reject)	20.31
NMF	34.14 (32.64: Reject)	31.62
LDA	53.35 (9.15: Reject)	56.51
CTM	47.25 (16.59: Reject)	50.92
MGCTM	56.26 (5.58: Reject)	60.00
GLDA	56.42 (5.38: Reject)	61.19
LDA mixture model	55.92 (6.00: Reject)	59.63

Bold values indicate the highest values of the clustering accuracy and NMI between all methods

Table 3 Evaluation of different clustering methods on the WebKB dataset

Clustering method	Clustering accuracy (%) (q value: reject/accept)	NMI (%)
CSTC	**61.20**	**39.45**
STC K-means	49.90 (11.33: Reject)	32.16
STC	56.28 (2.64: Reject)	36.28
K-means	43.98 (9.11: Reject)	23.84
NC	22.19 (27.15: Reject)	12.36
NMF	46.71 (7.68: Reject)	33.28
LDA	53.63 (4.04: Reject)	34.17
CTM	52.97 (4.39: Reject)	33.96
MGCTM	56.68 (2.42: Reject)	36.54
GLDA	56.95 (2.28: Reject)	37.11
LDA mixture model	56.33 (2.61: Reject)	36.31

Bold values indicate the highest values of the clustering accuracy and NMI between all methods

traditional methods by at least 26% in clustering accuracy and 33% in NMI and outperforms the topic model based clustering methods by at least 4% in clustering accuracy and about 4% in NMI, using 20-Newsgroups dataset. This improvement for WebKB dataset is at least 14% in clustering accuracy and 6% in NMI, compared to traditional methods and at least 4% in clustering accuracy and 2% in NMI, compared to topic model based clustering methods.

To determine the statistical significance of the results, the test statistic for our proposed CSTC method versus other clustering methods is also shown in Tables 2 and 3. The decision to reject or accept the null hypothesis is based on the 95% level of significance, i.e., if $q > (z_{0.95} = 1.64)$, we reject the null hypothesis and conclude that the improvement in the performance is significant, otherwise, we conclude that the results do not support the rejection of the null hypothesis (i.e., accept). As shown in

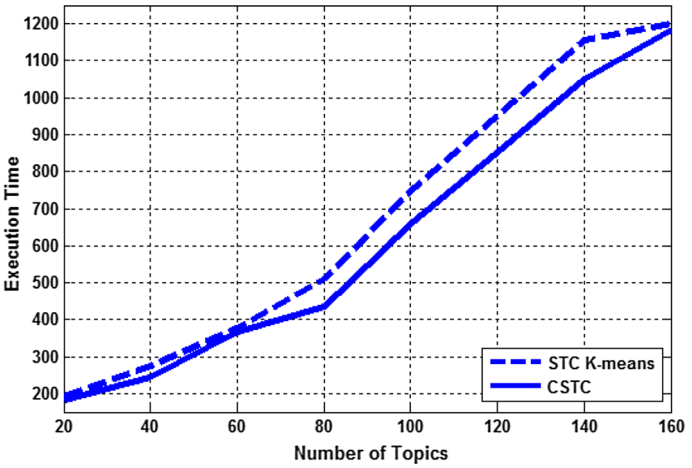

Fig. 8 Execution time (in seconds) versus the number of topics on the 20-Newsgroups dataset

Table 4 Execution time of different clustering methods on the 20-Newsgroups dataset

Bold value shows the minimum execution time between different methods

Clustering method	Execution time (s)
CSTC	659
STC K-means	952
STC	207
K-means	**9**

Tables 2 and 3, the clustering accuracy improvement achieved by the proposed CSTC method over all other clustering methods is significant.

We also measure the execution time as total time of training and testing. Figure 8 shows the execution time versus the number of topics, on the 20-Newsgroups dataset, for STC K-means and CSTC methods. As it can be seen, for both methods, execution time increases with the number of topics. Table 4 summarizes the execution time of K-means, STC, STC K-means and CSTC methods. For STC K-means, the execution time in 120 topics, and for CSTC, the execution time in 100 topics are reported. The result indicates that K-means has the least execution time compared to other methods. This demonstrates that STC increase the performance of clustering with loss of speed.

3.6 Selecting the hyper-parameter values

The hyper-parameters $\lambda_1, \lambda_2, \lambda_3$ control the sparsity of document codes $\boldsymbol{\theta}$ and word codes $\boldsymbol{s}$ (Wang et al. 2014). In general, larger values of λ_1, λ_2 and λ_3 lead to a sparser $\boldsymbol{\theta}$ and $\boldsymbol{s}$. On the other hand, a topic model with highly sparse $\boldsymbol{\theta}$ and $\boldsymbol{s}$ will miss some useful topic-word relationships. This will lead to a poor reconstruction performance. Therefore, in practice, we must try larger λ_1 and λ_2 to obtain sparser $\boldsymbol{\theta}$ and $\boldsymbol{s}$ such that the reconstruction performance (over the training data) is kept in an acceptable level. In Wang et al. (2014), these parameters are being selected using a generic grid

search based on cross-validation over a portion of the training data. These values are kept fixed for evaluating the method over the test data. The search process can be simplified by setting $\lambda_1 = \lambda_3$ (Zhu and Xing 2011). According to Wang et al. (2014), the code sparsity parameters λ_1, λ_2 and λ_3 are insensitive to the dataset. Therefore, $\lambda_1 = \lambda_3 = 0.5$ and $\lambda_2 = 0.2$ experientially for all datasets.

Parameter λ_4 controls the effect of the K-means clustering term. As λ_4 increases, the K-means clustering term plays a more important role in the optimization problem stated in (8). For tuning the K-means clustering term, coefficient λ_4 must be set to a value that the highest clustering accuracy is achieved through cross-validation over the training data. A practical choice for the value of these coefficients can be found according to the ratio of the corresponding terms. In this manner, we can estimate λ_4 as shown in:

$$\frac{\lambda_1}{\lambda_4} \propto \frac{\sum_{l=1}^{L} \sum_{d \in C_l} \left\| \boldsymbol{\theta}_d - \boldsymbol{\mu}_l \right\|_2^2}{\sum_{d=1}^{D} \left\| \boldsymbol{\theta}_d \right\|_1} = \frac{\text{inter-cluster-variance}(\boldsymbol{\theta})}{\text{mean}(\boldsymbol{\theta})} \tag{31}$$

As the mean value is much larger than the inter-cluster-variance value, λ_4 should be considered much larger than λ_1. For example we set λ_4 to ten times λ_1, and $\lambda_4 = 5$ is used in our experiments.

3.7 Discussion

According to experimental clustering results, topic model based clustering methods including STC, STC K-means and CSTC, LDA, CTM, MGCTM, GLDA and LDA mixture model are generally better than K-means, NC and NMF. This shows that topic modeling can promote document clustering. The semantics discovered by topic models can effectively facilitate accurate similarity measure, which is helpful to recognize coherent clusters. In STC K-means method, the STC is first learned off-line and K-means clustering is conducted subsequently on the document codes built from the STC. Compared with STC K-means which performs clustering and modeling separately, the STC, CSTC, LDA, CTM, MGCTM, GLDA and LDA mixture model which jointly perform two tasks achieve much better results. This demonstrates that clustering and modeling can mutually promote each other. Information on the category of documents helps to solve the ambiguity of word meaning in discovering the topics, and vice versa. Thus, coupling document clustering and topic modeling into a unified framework produces superior performance than separating them into two procedures.

Among the methods which unify clustering and modeling, our proposed CSTC method achieves the best clustering result. In STC and LDA methods, one cluster of documents only corresponded to one topic. Assigning each cluster only one topic may not be sufficient to capture the diverse semantics within each cluster. On the contrary, CTM, MGCTM, GLDA, LDA mixture model and CSTC assign each cluster a set of topics.

CSTC combines document modeling and clustering in a unified framework where these two tasks are jointly accomplished. In each iteration of the inference and learning process, the cluster assignments of documents depend on the current learned document

codes and the estimation of document codes depends on the current inferred cluster labels. Learning the document codes is continually guided by intermediate clustering results. As such, they are specifically suitable for clustering task in the end. Compared to CTM, MGCTM, GLDA and LDA mixture model which are based on LDA, CSTC achieves higher clustering accuracy based on STC.

Intuitively, instead of letting all the topics to contribute in descriptions, it is reasonable to assume that each document or word has a few salient topical meanings. In comparison to LDA, the fact that STC directly achieves the document sparsity by imposing sparsity-inducing regularization on the inferred document representations, makes it advantages over LDA. The problem is that LDA is not able to discover sparse latent representations. That is mainly because LDA uses a Dirichlet distribution which prevents any zero contributions of words to topics and of topics to documents. As a result, due to the extreme density of the learned topics and new representations of documents, document sparsity will be equal to zero. Contrary to LDA, learning sparse latent representations of documents and deliberate contribution of only few topics to a document are allowed by STC. In document clustering and information retrieval, which are considered as large scale text mining applications, this sparsity is to be considered.

4 Conclusion

In this paper, we proposed a clustering topic model to simultaneously perform document clustering and modeling. Experiments demonstrated the fact that topic modeling task is closely related to document clustering and can mutually promote it. We conducted our experiments on two popular text datasets to evaluate the proposed model. The results indicated that through jointly topic modeling, our proposed CSTC method can achieve a much better performance compared to the traditional methods (Kmeans, NC and NMF) and better performance compared to other topic model based clustering methods (LDA, CTM, MGCTM, GLDA and LDA mixture model). According to the experimental results, CSTC significantly improves the accuracy and NMI of other methods by at least about 4% in the 20-Newsgroups dataset. These accuracy and NMI improvements are at least 4 and 2% respectively in the WebKB dataset.

References

Ahmadi P, Kaviani R, Gholampour I, Tabandeh M (2015) Clustering improvement via integrating with sparse topical coding. In: 23rd Iranian conference on electrical engineering, IEEE, pp 466–471. http://ieeexplore.ieee.org/document/7146260/

Blei DM, Ng AY, Jordan MI, Lafferty J (2003) Latent Dirichlet allocation. J Mach Learn Res 3(Jan):993–1022

Fritzke B (1995) A growing neural gas network learns topologies. Adv Neural Inf Process Syst 7:625–632

Hofmann T (1999) Probabilistic latent semantic analysis. In: Proceedings of the fifteenth conference on uncertainty in artificial intelligence, Morgan Kaufmann Publishers Inc., pp 289–296

Hyvarinen A (1999) Sparse code shrinkage: denoising of nongaussian data by maximum likelihood estimation. Neural Comput 10:1739–1768

Kuhn HW (1955) The hungarian method for the assignment problem. Naval Res Logist Q 2(1–2):83–97

Lamirel JC (2012) A new approach for automatizing the analysis of research topics dynamics: application to optoelectronics research. J Scientometr 93(1):151–166

Lamirel JC, Falk I, Gardent C (2015) Federating clustering and cluster labelling capabilities with a single approach based on feature maximization: French verb classes identification with IGNGF neural clustering. Neurocomputing 147:136–146

Lee H, Battle A, Raina R, Ng AY (2006) Efficient sparse coding algorithms. In: Advances in neural information processing systems, pp 801–808

Li X, Ouyang J, Lu Y, Zhou X, Tian T (2014) Group topic model: organizing topics into groups. Inf Retr J 18(1):1–25

Lu Y, Mei Q, Zhai C (2011) Investigating task performance of probabilistic topic models: an empirical study of plsa and lda. Inf Retr 14(2):178–203

Papoulis A, Pillai SU (2002) Probability, random variables and stochastic processes, 4th edn. McGraw-Hill, New York

Teh YW, Jordan MI, Beal MJ, Blei DM (2006) Hierarchical Dirichlet processes. J Am Stat Assoc 101(476):1566–1581

Wallach HM (2008) Structured topic models for language. Doctoral dissertation, Univ. of Cambridge

Wang X, Ma X, Grimson WEL (2009) Unsupervised activity perception in crowded and complicated scenes using hierarchical Bayesian models. IEEE Trans Pattern Anal Mach Intell 31(3):539–555

Wang J, Fu W, Lu H, Ma S (2014) Bilayer sparse topic model for scene analysis in imbalanced surveillance videos. IEEE Trans Image Process 23(11):5198–5208

Xie P, Xing EP (2013) Integrating document clustering and topic modeling. In: Proceedings of the twenty-ninth conference on uncertainty in artificial intelligence, p 694. http://auai.org/uai2013/prints/papers/35.pdf

Zhu J, Xing E (2011) Sparse topical coding. In: Proceedings of the twenty-seventh conference annual conference on uncertainty in artificial intelligence (UAI), pp 831–838. http://bigml.cs.tsinghua.edu.cn/~jun/code/stc/stc.pdf