Anomaly Aware Symmetric Non-negative Matrix Factorization for Short Text Clustering

Bo Fu (fubo@lnnu.edu.cn)
Liaoning Normal University, Dalian University of Technology

Ximing Li
Jilin University

Yuanyuan Guan
Jilin University

Zhongxuan Luo
Dalian University of Technology

Research Article

Keywords: Short text clustering, Anomaly detection, Affinity matrix, Matrix factorization

Posted Date: April 12th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1543071/v1

Noname manuscript No.
(will be inserted by the editor)

Anomaly Aware Symmetric Non-negative Matrix Factorization for Short Text Clustering

Ximing Li · Yuanyuan Guan · Bo Fu* · Zhongxuan Luo

Received: date / Accepted: date

Abstract Short text clustering is a significant yet challenging task, where short texts generated from the Internet are extremely sparse, noisy, and ambiguous. The sparse nature makes traditional clustering methods, *e.g., k*-means family and topic modeling, much less effective. Fortunately, recent arts of document distance, *e.g.,* word mover's distance, and document representation, *e.g.,* BERT, can accurately measure the similarities of short texts, especially their nearest neighbors. Inspired by those arts and observations, we induce short text clusters by directly factorizing the informative affinity matrix of nearest neighbors into the product of the cluster assignment matrix, following the intuition that neighboring short texts tend to be assigned to the same cluster. However, due to the noisy nature of short texts, many of them can be regarded as outliers or near-outliers, resulting in many noisy neighboring similarities within the affinity matrix. To further alleviate this problem, we enhance the affinity matrix factorization by (1) incorporating a sparse noisy matrix to directly capture noisy neighboring similarities, and (2) regularizing the cluster assignment matrix by $\ell_{2,1}$ norm to eliminate hard-to-clustering short texts (called pseudo-outliers), so as to indirectly neglect noisy neighboring similarities corresponding to them. After this factorization for pre-clustering, we train a classifier over the resulting clusters, and adopt it to assign each pseudo-outlier to one cluster finally. We call this novel clustering method as *A*nomaly *A*ware *S*ymmetric *N*on-negative *M*atrix *F*actorization

Ximing Li and Yuanyuan Guan (Contributing equally with the first author)
College of Computer Science and Technology, and Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, China.
E-mail: liximing86@gmail.com; guanyy16@gmail.com

Bo Fu (corresponding author, denoted by *)
School of Computer and Information Technology, Liaoning Normal University, and School of Software, Dalian University of Technology, China.
E-mail: fubo@lnnu.edu.cn

Zhongxuan Luo
School of Software Technology, and Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian University of Technology, China.
E-mail: zxluo@dlut.edu.cn

(A^2SNMF). Experimental results on benchmark short text datasets demonstrate that A^2SNMF performs very competitive with the existing baseline methods.

Keywords Short text clustering, Anomaly detection, Affinity matrix, Matrix factorization

1 Introduction

Nowadays, short text has become a more fashionable form of text information, such as social media posts, question titles, and news headlines, to name just a few. Due to the big volume of short texts generated from the Internet everyday, they potentially involve a tremendous amount of valuable information and knowledge, which are significant for basic text mining tasks of data mining and information retrieval [9,25]. In this paper, we concentrate on the active research topic of **short text clustering**, whose goal is to partition unlabeled short texts into a number of clusters. Typically, it plays a vital role in pre-processing and benefits multiple real-world applications of short texts [48,53].

Unfortunately, short texts from the Internet are often extremely short, noisy, and ambiguous, imposing great challenges to clustering. Generally speaking, one primary difficulty is that each short text only contains very few word tokens (see examples in Fig.1(a)), and this extremely sparse nature leads to frequent near-orthogonality Bag-of-Words (BoW) representations, formally referred to as the so-called **sparsity problem**. Due to this problem, many traditional clustering methods, *e.g.,* prototype-based clustering methods such as *k*-means family, and model-based methods such as Latent Dirichlet Allocation (LDA) topic modeling [4], almost fail to handle short texts. In terms of *k*-means, it is intractable to learn discriminative cluster prototypes with extremely sparse BoW features [14]; and in terms of LDA, short texts lack the word co-occurrence information at the document level, so as to hurt the topic inference as well as clustering [43,9]. To resolve the sparsity problem, some recent attempts are motivated by leveraging the spirit of short text enrichment with various techniques, *e.g.,* external knowledge bases [14], global word co-occurrences [9,42], and pre-trained word embeddings [26,27]. However, those methods may often generate incoherent enrichment of short texts and even mix with noise, still limiting the performance of short text clustering.

Fortunately, several previous studies [48,29] report that the methodology of capturing the local structure, *i.e., k* Nearest Neighbors (NNs), of short texts, can effectively boost the clustering performance, and meanwhile, the recent new arts of document distance, *e.g.,* Word Mover's Distance (WMD) [22], and document representation, *e.g.,* BERT [10], can accurately measure the similarities of short texts, especially their nearest neighbors. Motivated by those arts and observations, we investigate a proposal for short text clustering: It first computes an accurate affinity matrix of k short text NNs by using recent arts such as WMD or BERT embeddings, and then factorizes this informative affinity matrix into the product of cluster assignment matrix as Symmetric Non-negative Matrix Factorization (SymNMF) [20]. The basic idea of factorizing the affinity matrix (*i.e.,* SymNMF) is that neighboring short texts tend to be assigned to the same cluster. Formally, let $\mathbf{S} \in \mathbb{R}^{n \times n}$ and $\mathbf{C} \in \mathbb{R}_{+}^{n \times l}$

Table 1 Early empirical results of *k*-means, LDA, SymNMF, and our A^{2}SNMF. All experimental settings are described in Section 4.

Methods	*StackOverFlow*		*Biomedical*	
	ACC ↑	**NMI** ↑	**ACC** ↑	**NMI** ↑
***k*-means**	0.36±0.04	0.45±0.04	0.21±0.02	0.22±0.02
LDA [4]	0.41±0.03	0.37±0.06	0.27±0.02	0.26±0.03
SymNMF [20]	0.65±0.02	0.56±0.01	0.39±0.01	0.34±0.00
A^{2}SNMF (Ours)	**0.70±0.01**	**0.58±0.02**	**0.45±0.02**	**0.37±0.01**

denote the affinity matrix of n short texts (with k NNs only) and cluster assignment matrix with l clusters, respectively. The formulation of SymNMF can be described below:

$$\min_{\mathbf{C}\geq\mathbf{0}} \ \|\mathbf{S}-\mathbf{C}\mathbf{C}^\top\|_F^2, \tag{1}$$

where $\|\cdot\|_F$ is the Frobenius norm. To show the effectiveness of SymNMF (*i.e.,* Eq.1) for short text clustering, we present early experimental results of benchmark datasets in Table 1. We can observe that SymNMF outperforms *k*-means and LDA by a large margin, indicating that for short texts, inducing clusters from the affinity matrix may be much more effective than from BoW features.

Our contributions. Based on aforementioned analyses and early empirical results, we regard SymNMF as a candidate backbone for short text clustering. However, it also suffers from the **noisy neighbor problem**: Due to the noisy nature of short texts, many of them can be regarded as outliers or near-outliers, potentially resulting in many noisy neighboring similarities within the affinity matrix (see examples in Fig.1), so as to hurt SymNMF clustering.

To remedy this problem, in this paper we propose a novel clustering method, namely **A**morally **A**ware **S**ymmetric **N**on-negative **M**atrix **F**actorization (**A^{2}SNMF**). Specifically, A^{2}SNMF consists of two stages: **pre-clustering** and **cluster reassignment**. In the pre-clustering stage, we formulate a novel objective of affinity matrix factorization, whose main idea is two-fold: First, we incorporate a noisy matrix with sparse ℓ_1 norm regularization to directly capture noisy neighboring similarities. Second, we regularize the cluster assignment matrix by $\ell_{2,1}$ norm to eliminate hard-to-clustering short texts (called **pseudo-outliers**), so as to indirectly neglect noisy neighboring similarities corresponding to them. Therefore, optimizing this novel objective of A^{2}SNMF can output more accurate clusters without pseudo-outliers. In the cluster reassignment stage, we regard the clusters computed from the pre-clustering stage as a pseudo-training dataset, and then train a prototype-based classifier with it. Each pseudo-outlier is predicted to one cluster by adopting this classifier. To evaluate the performance of A^{2}SNMF, we empirically compare it against existing clustering methods of various types. The experimental results demonstrate that A^{2}SNMF performs very competitive with the baseline methods measured by the widely used external metrics of clustering.

In a nutshell, the contributions of the paper are summarized as follows.

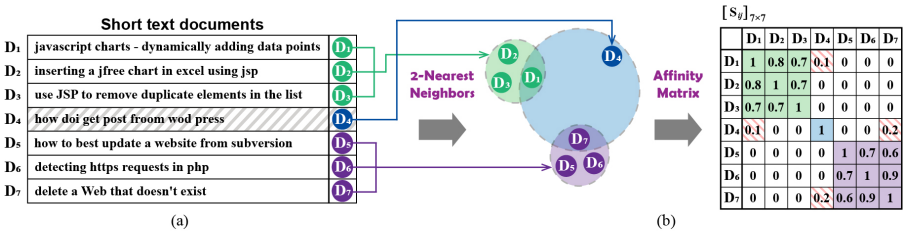

Fig. 1 Examples of short texts and inaccurate affinity matrix with near-outlier. (a) Examples of 7 short texts $D_{1,\cdots,7}$ with very few words. We emphasize that D_4 is a near-outlier with many typos. (b) The corresponding affinity matrix $[S_{ij}]_{7\times 7}$ with 2 NNs. Specifically, the component $S_{ij} = 0$ denotes that D_i and D_j are non-neighbors, and vice-versa S_{ij} describes the similarity between the corresponding neighbors. In this example, we can observe that the near-outlier D_4 results in some noisy neighboring similarities with lower values, which are indicated by red shadow cells. Best viewed in **color**.

– We analyze previous studies and also empirically indicate that inducing clusters from the affinity matrix is an effective solution for short text clustering, and SymNMF can be regarded as a candidate backbone.
– We propose a novel short text clustering method named A^{2}SNMF, which alleviates the noisy neighbor problem of SymNMF.
– We conduct a number of experiments on benchmark short text datasets. Empirical results demonstrate the superior performance of A^{2}SNMF.

The rest of this paper is organized as follows: In Section 2, we introduce some related works. In Section 3, we introduce the proposed A^{2}SNMF method in detail. Experimental results and discussions are presented in Section 4. In Section 5, the conclusion is made.

2 Related Work

In this section, we briefly review the related works on short text clustering, nonnegative matrix factorization, and anomaly detection, respectively.

2.1 Short Text Clustering

Short text clustering has attracted much attention from the community, and several methods have been recently developed [14,51,9,39,47,52,26,48,50,28,15,42,27,29,53]. To our knowledge, the current mainstream methods can be roughly divided into two categories in the literature: *topic modeling method* and *deep learning method*.

The topic modeling methods uncover the latent topic structures of documents by leveraging various generative processes [4,3]. They estimate the topic representations of documents and then assign clusters with them, based on the concept correspondence between topic and cluster. To effectively handle the sparse short texts, existing topic models mainly focus on word co-occurrence enrichment, so as to estimate more discriminative topic representation for each short text. For example, the

Biterm Topic Model (BTM) [9] directly leverages global word co-occurrences at the corpus level; and the Dirichlet Multinomial Mixture (DMM) [51] supposes that each short text covers only a single topic, indirectly enriching word co-occurrences at the document level. Recently, they are upgraded by incorporating pre-trained word embeddings [26,27] and variational manifold regularization [29]. Comparing with those aforementioned methods, our A^{2}SNMF trains short text clusters from the affinity matrix, and further addresses the noisy nature of short texts.

The main spirit of deep learning methods is to induce more discriminative deep features beyond sparse BoW features, making short text clustering much easier. The STC^2 method [47,48], learns deep features by optimizing with traditional dimensionality reduction methods, followed by k-means clustering. The ST-STC method [13] first pre-trains a deep auto-encoder to initialize the deep features, and then performs an end-to-end self-training process for the final clustering [46]. Additionally, another method [53] computes more robust deep features under the adversarial training manner. Orthogonal to those methods, A^{2}SNMF is flexible, also enabling to employ deep features, *i.e.,* BERT embeddings, to construct the affinity matrix. We will investigate the end-to-end version of A^{2}SNMF in the future.

2.2 Non-negative Matrix Factorization and Anomaly Detection

Typically, the Non-negative Matrix Factorization (NMF) [24] can be interpreted as a dimensionality reduction method, which factorizes the instance matrix into non-negative low-rank approximations. The literature [11] analyzes the equivalence of NMF and spectral clustering, indicating that NMF can be also applied to clustering. Many variants of NMF have been proposed by leveraging various techniques. For example, the Graph regularized NMF (GNMF) [7,6] preserves the local structures of instances by using the manifold regularization, and the Semantics-assisted NMF (SeaNMF) [42] handles the sparsity problem of short texts with corpus-level word co-occurrences. Many studies aim to efficiently solve these NMF-like methods, such as Alternating Non-negative Least Squares (ANLS) [19,21]. The SymNMF method [20] can be viewed as a special case of NMF, which factorizes the affinity matrix of instances, instead of the original instance matrix. It has been long investigated [45,40,17,33], and widely applied to various real applications, such as community detection [32,34] and recommender systems [35]. In many scenarios, SymNMF can achieve competitive clustering performance [20,16]. Beyond it, our A^{2}SNMF further considers the noisy nature of short texts, so as to achieve a more robust clustering performance.

Anomaly detection [18] is a well-established yet fundamental task in data analysis. Broadly speaking, the typical kinds of anomaly detection methods include proximity-based methods [1,49], rule-based methods [36], and information-theoretic methods [44], *etc.* Specifically, several studies investigate joint clustering and anomaly detection [8,37,30]. However, they are not applicable to extremely sparse data such as short texts.

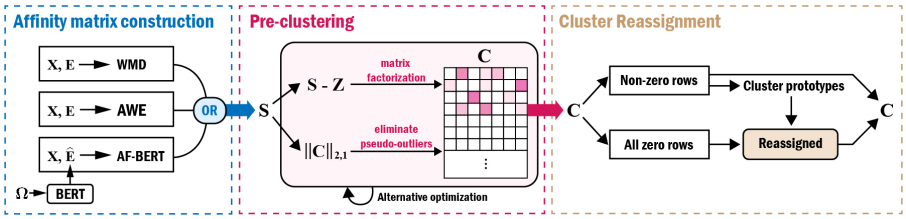

Fig. 2 The overall framework of A^{2}SNMF. We construct the affinity matrix $\mathbf{S}$ by any WMD, AWE, or AF-BERT. We then perform pre-clustering and cluster reassignment.

3 Proposed A^{2}SNMF Method

In this section, we clarify the proposed **A**morally **A**ware **S**ymmetric **N**on-negative **M**atrix **F**actorization (**A^{2}SNMF**) method in detail.

3.1 Overall Description of A^{2}SNMF

Consider a collection $\mathbf{\Omega}$ of n short texts with a fixed vocabulary of v words, which can be generally described by the normalized BoW matrix $\mathbf{X} \in \mathbb{R}^{n\times v}$. Commonly, each short text contains only very few word tokens, *i.e.,* each row of $\mathbf{X}$ involves very few non-zero values. The task of short text clustering refers to partitioning those short texts into l clusters. We define the cluster assignment matrix $\mathbf{C} \in \mathbb{R}_+^{n\times l}$ to represent the resulting cluster, where each short text is assigned to the cluster with maximum value in its corresponding row of $\mathbf{C}$.

To handle this task, the basic idea of A^{2}SNMF is to estimate $\mathbf{C}$ by factorizing the affinity matrix of k short text NNs, denoted by $\mathbf{S} \in \mathbb{R}^{n\times n}$. The overall framework of A^{2}SNMF is shown in Fig.2. To be specific, we first construct $\mathbf{S}$ by three various document distance techniques as three options. Given $\mathbf{S}$, A^{2}SNMF induces short text clusters by the following two stages.

- **Pre-clustering**: To alleviate the noisy neighbor problem, we formulate a novel SymNMF-like objective by (1) incorporating a noisy matrix with sparse ℓ_1 norm regularization to directly capture noisy neighboring similarities, and (2) regularizing $\mathbf{C}$ by $\ell_{2,1}$ norm to eliminate hard-to-clustering short texts (called **pseudo-outliers**), so as to indirectly neglect noisy neighboring similarities corresponding to pseudo-outliers
- **Cluster reassignment**: In this stage, we regard the resulting clusters computed from the pre-clustering stage as a pseudo-training dataset, and then train a prototype-based classifier with it. Each pseudo-outlier is predicted to one cluster by adopting this classifier.

We describe the details of affinity matrix construction, pre-clustering, and cluster reassignment stages in the following parts of this section. For clarity, we outline the important notations in Table 2.

3.2 Affinity Matrix Construction

In this work, we attempt to construct $\mathbf{S}$ by using one of the three document distance techniques, including WMD, Average pre-trained Word Embeddings (AWE), and Average Fine-tuned BERT embeddings (AF-BERT). We show implementation details below.

- **WMD**: The WMD is a novel document distance that measures the optimal transport plan between documents [22]. To apply it, we compute the cost of each word pair by employing the pre-trained GloVe word embeddings [38], denoted by $\mathbf{E} \in \mathbb{R}^{v\times d}$, where d is the embedding dimension. The relaxed version of WMD is finally used for speedup [22].
- **AWE**: Given the GloVe word embeddings $\mathbf{E}$, we can represent each short text by a weighted average of embeddings corresponding to occurring words. Formally, we transform $\mathbf{X}, \mathbf{E}$ into the AWE matrix $\mathbf{A} \triangleq \mathbf{XE} \in \mathbb{R}^{n\times d}$. For each short text pair $\{\mathbf{A}_i, \mathbf{A}_j\}$, we compute their distance by the cosine measure, formulated as $\frac{1-\cos(\mathbf{A}_i,\mathbf{A}_j)}{2}$.
- **AF-BERT**: We fine-tune the pre-trained BERT model [10] over the raw data $\mathbf{\Omega}$. Each short text is represented by a weighted average of contextualized embeddings corresponding to occurring words, This gives the AF-BERT matrix $\widehat{\mathbf{B}} \in \mathbf{R}^{n\times\widehat{d}}$, where $\widehat{d}$ is the embedding dimension. For each short text pair $\{\mathbf{B}_i, \mathbf{B}_j\}$, we compute their distance as $\frac{1-\cos(\mathbf{B}_i,\mathbf{B}_j)}{2}$.

With any document distance technique, the affinity matrix is finally computed by the following formula:

$$\mathbf{S}_{ij} = \begin{cases} \exp\left(-\mathbf{dis}(i,j)\right), & i \in \mathit{\Pi}_j \text{ or } j \in \mathit{\Pi}_i \\ 0, & \text{otherwise} \end{cases} \qquad \forall i,j \in [n]$$

where $\mathit{\Pi}_i$ denotes the k NNs set of the i-th short text; and $\mathbf{dis}(i,j)$ is the distance between two short texts by any document distance technique.

3.3 Pre-clustering

Due to the noisy nature of short texts, the affinity matrix $\mathbf{S}$ potentially contains many noisy neighboring similarities among short texts. The main goal of pre-clustering is to alleviate this noisy neighbor problem. To this end, we propose a novel SymNMF-like objective with two tricks. **First**, we incorporate a learnable noisy matrix $\mathbf{Z} \in \mathbb{R}^{n\times n}$ to directly capture noisy neighboring similarities, and a learnable clean approximating affinity matrix $\mathbf{G} \in \mathbb{R}^{n\times n}$, satisfying $\mathbf{S} = \mathbf{G} + \mathbf{Z}$. We then compute the cluster assignment matrix $\mathbf{C}$ by factorizing $\mathbf{G}$, instead of $\mathbf{S}$. Further, we suppose that the noisy matrix $\mathbf{Z}$ tends to be sparse, so as to place the sparse ℓ_1 norm regularization on it. Accordingly, we formulate the following objective:

$$\begin{aligned} \min_{\mathbf{C},\mathbf{G},\mathbf{Z}} \quad & \|\mathbf{G} - \mathbf{C}\mathbf{C}^\top\|_F^2 + \lambda_1\|\mathbf{Z}\|_1 \\ \textbf{s.t.} \quad & \mathbf{G} + \mathbf{Z} = \mathbf{S}, \qquad \mathbf{C}, \mathbf{G}, \mathbf{Z} \succeq \mathbf{0}\,, \end{aligned} \tag{2}$$

Table 2 Summary of the important notations.

Notation	Description
n	number of short texts
v	size of vocabulary
l	number of clusters
k	number of nearest neighbors
$\mathbf{X} \in \mathbb{R}^{n \times v}$	normalized BoW matrix
$\mathbf{C} \in \mathbb{R}_{+}^{n \times l}$	cluster assignment matrix
$\mathbf{S} \in \mathbb{R}^{n \times n}$	affinity matrix of k short text NNs
$\mathbf{E} \in \mathbb{R}^{v \times d}$	GloVe word embeddings
$\mathbf{A} \in \mathbb{R}^{n \times d}$	AWE matrix
$\widehat{\mathbf{E}} \in \mathbb{R}^{\mathbf{v} \times \widehat{\mathbf{d}}}$	fine-tuned BERT embeddings
$\mathbf{B} \in \mathbb{R}^{n \times \widehat{d}}$	AF-BERT matrix
$\mathbf{G} \in \mathbb{R}^{n \times n}$	clean approximation of $\mathbf{S}$
$\mathbf{Z} \in \mathbb{R}^{n \times n}$	noisy matrix

where $\|\cdot\|_1$ is the ℓ_1 norm; and λ_1 is a regularization parameter. **Second**, we regularize $\mathbf{C}$ by $\ell_{2,1}$ norm to eliminate pseudo-outliers, *i.e.,* the short texts correspond to the zero rows of $\mathbf{C}$. The intuition is that the resulting pseudo-outliers are more likely to generate noisy neighboring similarities. By further incorporating this $\ell_{2,1}$ norm regularizer, the final objective of A^{2}SNMF with respect to $\{\mathbf{C}, \mathbf{G}, \mathbf{Z}\}$ is formulated below:

$$\begin{aligned} \min_{\mathbf{C},\mathbf{G},\mathbf{Z}} \quad & \|\mathbf{G} - \mathbf{C}\mathbf{C}^\top\|_F^2 + \lambda_1 \|\mathbf{Z}\|_1 + \lambda_2 \|\mathbf{C}\|_{2,1} \\ \textbf{s.t.} \quad & \mathbf{G} + \mathbf{Z} = \mathbf{S}, \qquad \mathbf{C}, \mathbf{G}, \mathbf{Z} \succeq \mathbf{0}\,, \end{aligned} \tag{3}$$

where λ_2 denotes the regularization parameter.

Alternative optimization. Naturally, the objective of Eq.3 is intractable to optimize. To efficiently solve it, we employ the following formulation by incorporating an auxiliary cluster assignment matrix $\mathbf{V} \in \mathbb{R}^{n \times l}$ as in [21]:

$$\begin{aligned} \min_{\mathbf{V},\mathbf{C},\mathbf{G},\mathbf{Z}} \quad & \|\mathbf{G} - \mathbf{C}\mathbf{V}^\top\|_F^2 + \lambda_1 \|\mathbf{Z}\|_1 + \lambda_2 \|\mathbf{C}\|_{2,1} + \alpha \|\mathbf{C} - \mathbf{V}\|_F^2 \\ \textbf{s.t.} \quad & \mathbf{G} + \mathbf{Z} = \mathbf{S}, \qquad \mathbf{V}, \mathbf{C}, \mathbf{G}, \mathbf{Z} \succeq \mathbf{0}\,, \end{aligned} \tag{4}$$

where α is a scalar parameter to measure the importance of the difference between $\mathbf{C}$ and $\mathbf{V}$. Accordingly, the objective of Eq.4 can be treated as a special version of NMF, which can be efficiently solved by existing methods of NMF such as ANLS [19]. A potential issue of this non-symmetric formulation is that the difference between $\mathbf{C}$ and $\mathbf{V}$ may be relatively larger with smaller α values. Fortunately, in the early experiments we have empirically found the clustering results are insensitive to the value of α and we always fixed α to 1.

However, the objective of Eq.4 is also intractable to compute due to the $\ell_{2,1}$ regularization of $\mathbf{C}$. To make the optimization simple with the $\ell_{2,1}$ norm [31], we leverage

the idea of Alternating Direction Method of Multipliers (ADMM) [5] to further convert Eq.4 into the following objective with another auxiliary matrix $\mathbf{D} \in \mathbb{R}^{n \times l}$:

$$\begin{aligned} \min_{\mathbf{V},\mathbf{D},\mathbf{C},\mathbf{G},\mathbf{Z},\mathbf{\Theta_1},\mathbf{\Theta_2}} \ & \|\mathbf{S} - \mathbf{Z} - \mathbf{D}\mathbf{V}^\top\|_F^2 + \lambda_1\|\mathbf{Z}\|_1 + \lambda_2\|\mathbf{C}\|_{2,1} \\ & + \alpha\|\mathbf{D} - \mathbf{V}\|_F^2 + \frac{\mu_1}{2}\|\mathbf{S} - \mathbf{Z} - \mathbf{G} + \frac{\mathbf{\Theta_1}}{\mu_1}\|_F^2 \\ & + \frac{\mu_2}{2}\|\mathbf{C} - \mathbf{D} + \frac{\mathbf{\Theta_2}}{\mu_2}\|_F^2 \\ \textbf{s.t.} \quad & \mathbf{V}, \mathbf{D}, \mathbf{C}, \mathbf{G}, \mathbf{Z} \succeq \mathbf{0}\,, \end{aligned} \tag{5}$$

where $\mathbf{\Theta}_1 \in \mathbb{R}^{n \times n}$ and $\mathbf{\Theta}_2 \in \mathbb{R}^{n \times l}$ are the Lagrange multipliers; and $\{\mu_1, \mu_2\}$ are the penalty parameters. We can alternately update each variable of interest, *i.e.,* $\{\mathbf{V}, \mathbf{D}, \mathbf{C}, \mathbf{G}, \mathbf{Z}, \mathbf{\Theta_1}, \mathbf{\Theta_2}\}$, as the other ones fixed. Details are briefly clarified in the following parts.

[Update $\mathbf{V}$] By referring to [21], the sub-problem of $\mathbf{V}$ is formulated below:

$$\min_{\mathbf{V} \succeq \mathbf{0}} \ \|\widehat{\mathbf{S}} - \widehat{\mathbf{D}}\mathbf{V}^\top\|_F^2\,, \tag{6}$$

where $\widehat{\mathbf{S}} = \left[\mathbf{S}^\top - \mathbf{Z}^\top, \sqrt{\alpha}\mathbf{D}\right]^\top$ and $\widehat{\mathbf{D}} = \left[\mathbf{D}^\top, \sqrt{\alpha}\mathbf{I}\right]^\top$. We can alternatively update each column of $\mathbf{V}$ by leveraging the ANLS method [19,21]:

$$\mathbf{V}_{\cdot j} \leftarrow \left[\mathbf{V}_{\cdot j} + \frac{\left(\widehat{\mathbf{S}}^\top\widehat{\mathbf{D}}\right)_{\cdot j} - \left(\mathbf{V}\widehat{\mathbf{D}}^\top\widehat{\mathbf{D}}\right)_{\cdot j}}{\left(\widehat{\mathbf{D}}^\top\widehat{\mathbf{D}}\right)_{jj}}\right]_+, \tag{7}$$

where $[\,\cdot\,]_+ = \max(\,\cdot\,, \mathbf{0})$.

[Update $\mathbf{D}$] By referring to [21], the sub-problem of $\mathbf{D}$ is formulated below:

$$\min_{\mathbf{D} \succeq \mathbf{0}} \ \|\widetilde{\mathbf{S}} - \widetilde{\mathbf{V}}\mathbf{D}^\top\|_F^2\,, \tag{8}$$

where $\widetilde{\mathbf{S}} = \left[\mathbf{S}^\top - \mathbf{Z}^\top, \sqrt{\alpha + \frac{\mu_2}{2}}\left(\frac{\alpha\mathbf{V} + \frac{\mu_2}{2}(\mathbf{C} - \frac{\mathbf{\Theta_2}}{\mu_2})}{\alpha + \frac{\mu_2}{2}}\right)\right]^\top$, and $\widetilde{\mathbf{V}} = \left[\mathbf{V}^\top, \sqrt{\alpha + \frac{\mu_2}{2}}\mathbf{I}\right]^\top$. Similar to $\mathbf{V}$, we can alternatively update each column of $\mathbf{D}$ as follows:

$$\mathbf{D}_{\cdot j} \leftarrow \left[\mathbf{D}_{\cdot j} + \frac{\left(\widetilde{\mathbf{S}}^\top\widetilde{\mathbf{V}}\right)_{\cdot j} - \left(\mathbf{D}\widetilde{\mathbf{V}}^\top\widetilde{\mathbf{V}}\right)_{\cdot j}}{\left(\widetilde{\mathbf{V}}^\top\widetilde{\mathbf{V}}\right)_{jj}}\right]_+ \tag{9}$$

[Update $\mathbf{C}$] By removing the terms without $\mathbf{C}$, the corresponding sub-problem is given by:

$$\min_{\mathbf{C} \succeq \mathbf{0}} \frac{\mu_2}{2}\|\mathbf{C} - \mathbf{D} + \frac{\mathbf{\Theta_2}}{\mu_2}\|_F^2 + \lambda_2\|\mathbf{C}\|_{2,1} \tag{10}$$

After some derivations, we can solve the above objective with $\ell_{2,1}$ norm by updating each row of $\mathbf{C}$ (derivations given in the **Appendix**):

$$\mathbf{C}_i = \begin{cases} \left(1 - \frac{\lambda_2}{\mu_2 \|\widehat{\mathbf{C}}_i^+\|}\right) \widehat{\mathbf{C}}_i^+, \text{ if } \|\widehat{\mathbf{C}}_i^+\| > \frac{\lambda_2}{\mu_2} \\ \mathbf{0}, \qquad\qquad \text{otherwise} \end{cases}, \tag{11}$$

where $\widehat{\mathbf{C}}^+ = [\mathbf{D} - \mathbf{\Theta}_2/\mu_2]_+$.

Given this update rule, we can observe that the regularization parameter λ_2 plays a key role in controlling the zero rows of $\mathbf{C}$, *i.e.,* estimating near-outliers. However, A^{2}SNMF may be sensitive to λ_2, and it is intractable to be tuned. To handle this problem, before performing the update of Eq.11, each row of $\widehat{\mathbf{C}}^+$ is normalized to be the vector whose sum is one,

$$\overline{\mathbf{C}_i^+} = \frac{\widehat{\mathbf{C}}_i^+}{\|\widehat{\mathbf{C}}_i^+\|_1},$$

so that every $\|\widetilde{\mathbf{C}}_i\|$ is bounded. The update rule is then given as follows:

$$\mathbf{C}_i = \begin{cases} \|\widehat{\mathbf{C}}_i^+\|_1 \left(1 - \frac{\lambda_2}{\mu_2 \|\overline{\mathbf{C}_i^+}\|}\right) \overline{\mathbf{C}_i^+}, \text{ if } \|\overline{\mathbf{C}_i^+}\| > \frac{\lambda_2}{\mu_2} \\ \mathbf{0}, \qquad\qquad\qquad\qquad \text{otherwise} \end{cases}, \tag{12}$$

[Update G] By removing the terms without $\mathbf{G}$, we obtain the following sub-problem:

$$\min_{\mathbf{G} \succeq \mathbf{0}} \ \frac{\mu_1}{2} \|\mathbf{S} - \mathbf{Z} - \mathbf{G} + \frac{\mathbf{\Theta_1}}{\mu_1}\|_F^2 \tag{13}$$

It can be computed as follows:

$$\mathbf{G} = \left[\mathbf{S} - \mathbf{Z} + \frac{\mathbf{\Theta}_1}{\mu_1}\right]_+ \tag{14}$$

[Update Z] By removing the terms without $\mathbf{Z}$, the corresponding sub-problem is given by:

$$\min_{\mathbf{Z} \succeq \mathbf{0}} \|\mathbf{S} - \mathbf{Z} - \mathbf{D}\mathbf{V}^\top\|_F^2 + \lambda_1 \|\mathbf{Z}\|_1 + \frac{\mu_1}{2} \|\mathbf{S} - \mathbf{Z} - \mathbf{G} + \frac{\mathbf{\Theta}_1}{\mu_1}\|_F^2 \tag{15}$$

We can easily compute a closed-form solution by using the *soft thresholding operator* δ [5]. The solution is as follows:

$$\mathbf{Z} = \delta_{\lambda_1/2} \left(\frac{(2 + \mu_1)\mathbf{S} - 2\mathbf{D}\mathbf{V}^\top - \mu_1 \mathbf{G} + \mathbf{\Theta}_1}{2 + \mu_1} \right), \tag{16}$$

where $\delta_{\mathcal{K}}(x) = \max(\text{abs}(x) - \mathcal{K}, \mathbf{0})$.

[Update $\mathbf{\Theta_1}$ and $\mathbf{\Theta_2}$] The Lagrange parameters can be directly updated as follows:

$$\begin{aligned} \mathbf{\Theta_1} &\longleftarrow \mathbf{\Theta_1} + \mu_1(\mathbf{S} - \mathbf{Z} - \mathbf{G}), \\ \mathbf{\Theta_2} &\longleftarrow \mathbf{\Theta_2} + \mu_2(\mathbf{C} - \mathbf{D}) \end{aligned} \tag{17}$$

Convergence analysis. We now briefly analyze the convergence characteristics of pre-clustering, *i.e.,* Eq.5. Because it is derived from the well-established ADMM method [5], we omit its Lagrange parameters $\{\mathbf{\Theta}_1, \mathbf{\Theta}_2\}$ updated by gradient decent, but only focus on other important parameters of interest, *i.e.,* $\mathbb{P} \triangleq \{\mathbf{V}, \mathbf{D}, \mathbf{C}, \mathbf{G}, \mathbf{Z}\}$. At each iteration t, we denote $\ell(\mathbb{P}^{(t)})$ as the current loss of Eq.5. Because we update each parameter by holding other ones fixed, we specifically define $\widehat{\ell}(\cdot)$ as the loss with respect to one parameter, *e.g.,* $\widehat{\ell}(\mathbf{V}^{(t)})$, for convenience. Basically, we attempt to show $\ell(\mathbb{P}^{(t-1)}) \geq \ell(\mathbb{P}^{(t)})$ with alternative optimization.

For $\mathbf{V}$, we alternatively update each column by ANLS, *i.e.,* Eq.7, where it is the unique solution of its sub-problem by referring to **Theorem 2** in [19]. Accordingly, we have the following inequalities:

$$\widehat{\ell}(\mathbf{V}_{\cdot 1}^{(t-1)}) \geq \widehat{\ell}(\mathbf{V}_{\cdot 1}^{(t)}) \geq, \cdots, \geq \widehat{\ell}(\mathbf{V}_{\cdot l}^{(t-1)}) \geq \widehat{\ell}(\mathbf{V}_{\cdot l}^{(t)}) \tag{18}$$

For $\mathbf{D}$ also updated by ANLS, *i.e.,* Eq.9, we have similar inequalities:

$$\widehat{\ell}(\mathbf{D}_{\cdot 1}^{(t-1)}) \geq \widehat{\ell}(\mathbf{D}_{\cdot 1}^{(t)}) \geq, \cdots, \geq \widehat{\ell}(\mathbf{D}_{\cdot l}^{(t-1)}) \geq \widehat{\ell}(\mathbf{D}_{\cdot l}^{(t)}) \tag{19}$$

For $\mathbf{C}$, the update equation of Eq.12 is the optimal solution of its sub-problem (see details in the **Appendix**). Due to the dependency among each row of $\mathbf{C}$, we show the inequality of full $\mathbf{C}$:

$$\widehat{\ell}(\mathbf{C}^{(t-1)}) \geq \widehat{\ell}(\mathbf{C}^{(t)}) \tag{20}$$

For $\mathbf{G}$, the update equation of Eq.14 is obviously the optimal solution of its sub-problem, leading to the following:

$$\widehat{\ell}(\mathbf{G}^{(t-1)}) \geq \widehat{\ell}(\mathbf{G}^{(t)}) \tag{21}$$

Finally, for $\mathbf{Z}$, it is also updated by a closed-form solution of its sub-problem by referring to [5], giving the following:

$$\widehat{\ell}(\mathbf{Z}^{(t-1)}) \geq \widehat{\ell}(\mathbf{Z}^{(t)}) \tag{22}$$

Upon those inequalities 18, 19, 20, 21, and 22, the overall loss of Eq.5 will iteratively decrease with alternative optimization:

$$\ell(\mathbb{P}^{(t-1)}) \geq \ell(\mathbb{P}^{(t)}) \tag{23}$$

Further, we know that $\ell(\mathbb{P})$ must be greater than 0. Therefore, the alternaative optimization of Eq.5 will converge to a stationary point with a sufficient number of iterations.

Algorithm 1 Full process of A^{2}SNMF

Input: short text dataset $\mathbf{X}$, number of clusters l
Construct the affinity matrix $\mathbf{S}$ by any WMD, AWE, or AF-BERT
Initialize parameters and variables of pre-clustering
While not convergence Do
    Update $\mathbf{V}$ using Eq.7
    Update $\mathbf{D}$ using Eq.9
    Update $\mathbf{C}$ using Eq.12
    Update $\mathbf{G}$ using Eq.14
    Update $\mathbf{Z}$ using Eq.16
    Update $\{\mathbf{\Theta}_1, \mathbf{\Theta}_2\}$ using Eq.17
End While
Compute the cluster prototypes over pseudo-training datset using Eq.24
Reassign pseudo-outliers to their nearest clusters
Output: cluster assignment matrix $\mathbf{C}$

3.4 Cluster Reassignment

Given the optimal $\mathbf{C}$ learned from pre-clustering, each short text is assigned to the cluster with maximum value in its corresponding row of $\mathbf{C}$. Specially, by referring to Eq.12, the pre-clustering may lead to a number of all-zero rows in $\mathbf{C}$, corresponding to pseudo-outliers that have not been assigned to any cluster. To resolve this, we regard the resulting clusters as a pseudo-training dataset, and compute a v-dimensional prototype for each cluster. Inspired by [12], for cluster i, the weight of word j in its prototype is computed as follows:

$$p_{ij} = b^{\frac{SF_{ij}}{S_i}} \times \ln\left(\frac{l}{CF_j}\right), \tag{24}$$

where SF_{ij} denotes the document frequency of word j in cluster i; S_i is the total number of short texts that are associated with cluster i; CF_j is the number of clusters that contain word j; and b denotes a constant value fixed as $e-1$ as suggestions in [12]. Accordingly, each pseudo-outlier is assigned to its nearest cluster measured by the cosine similarities between its BoW feature and cluster prototypes.

For clarity, we present the full process of A^{2}SNMF in **Algorithm 1**.

3.5 Analysis of Time Complexity

We now analyze the time complexities of affinity matrix construction, pre-clustering, and cluster reassignment, respectively. First, we propose three options to construct the affinity matrix. For WMD, we employ its relaxed version, leading to $\mathcal{O}(n^2p^2 + v^2d)$ time in total, where p denotes the average length of short texts. For AWE and AF-BERT, their complexities are $\mathcal{O}(nvd + n^2d)$ and $\mathcal{O}(\beta + nv\widehat{d} + n^2\widehat{d})$, respectively, where we use $\mathcal{O}(\beta)$ to denote the fine-tuning cost of BERT. Second, we turn to the pre-clustering stage that optimizes the factorization objective of Eq.3. Overall, the update equations of all variables mainly involve matrix multiplication and addition operators. We directly summarize the per-iteration complexities of $\{\mathbf{V}, \mathbf{D}, \mathbf{C}, \mathbf{G}, \mathbf{Z}, \mathbf{\Theta_1}, \mathbf{\Theta_2}\}$ are $\mathcal{O}(n^2l + nl^3)$, $\mathcal{O}(n^2l + nl^2)$, $\mathcal{O}(nl)$, $\mathcal{O}(n^2)$, $\mathcal{O}(n^2l)$, $\mathcal{O}(n^2)$ and $\mathcal{O}(nl)$, respectively. The full per-iteration complexity is thus about $\mathcal{O}(n^2l+nl^3)$, It is worth noting

Table 3 Characteristics of the short text datasets. *AvgDoc* denotes the average document length.

Dataset	*#Doc*	*#Word*	*AvgDoc*	*#Label*
Trec	4,434	1,051	3.3	6
Snipptes	12,140	5,462	14.3	8
StackOverFlow	18,358	2,220	3.8	20
Biomedical	19,918	4,472	7.3	20
UCInews	10,379	7,889	6.7	30

that $\mathbf{S}, \mathbf{Z}$ and $\mathbf{G}$ are very sparse, so the complexity can be actually $\mathcal{O}(|\mathbf{S}|l + nl^3)$.[1] Besides, empirical results show that the optimization can converge within very few iterations (less than 10 in our early experiments), this stage is thus efficiently competitive. Finally, in the cluster reassignment stage, we compute the cluster prototypes, spending $\mathcal{O}((n-\widehat{n})\widehat{v}vl)$ time, where $\widehat{n}$ and $\widehat{v}$ denote the number of pseudo-outliers and average document length, respectively. We then compute the distances between pseudo-outliers and cluster prototypes, requiring $\mathcal{O}(\widehat{n}\widehat{v}l)$ time. Therefore, the full complexity of this stage is $\mathcal{O}((n-\widehat{n})\widehat{v}vl + \widehat{n}\widehat{v}l)$, which is linear to the volume of the dataset.

4 Experiment

In this section, we show and discuss the experimental results on short text clustering. We first clarify details of datasets, baselines, and metrics.

Datasets. In the experiments, we totally employ 5 short text datasets from various domains. They include *Trec*, *Snippets*, *StackOverFlow*, *Biomedical*, and *UCInews*. The *UCInews* dataset we used is a subset of the UCI News Aggregator dataset. The "STORY" identifier is treated as a label, and we use only the 30 stories with the most documents. During the pre-processing, we eliminate the words appearing in less than 5 documents and further remove the empty short texts. The characteristics of those datasets are clarified in Table 3.

Baselines. In the experiments, we employ 12 existing baseline clustering methods, including prototype-based method, spectral clustering method, NMF-based methods, topic modeling methods, and deep learning methods. For clarity, we outline the details of all comparing methods below.

- ***k*-means**: We apply the standard *k*-means algorithm over the TF-IDF representations.
- **Normalized cut (Ncut)** [41]: This is a standard spectral clustering method.
- **NMF** [11]: We adopt the standard NMF model solved by the ANLS method [19]. The code is available on the net.[2]
- **GNMF** [6]: The model is an enhanced version of NMF with manifold regularization. The code is available on the net.[3]

[1] The notation $|\mathbf{S}|$ denotes the number of non-zero elements in $\mathbf{S}, \mathbf{Z}, \mathbf{G}$.

[2] https://github.com/hiroyuki-kasai/NMFLibrary

[3] http://www.cad.zju.edu.cn/home/dengcai/Data/GNMF.html

- **SymNMF** [20]: This is the standard symmetric NMF model solved by the symmetric ANLS method [21]. The code is available on the net.[2]
- **SeaNMF** [42]: This model is an enhanced version of NMF with word co-occurrences for short texts. The code is available on the net.[4]
- **LDA** [4]: The traditional LDA topic model inferred by variational inference. The code is available on the net.[5]
- **BTM** [9]: This is a topic model that captures the corpus-level word co-occurence patterns. The code is available on the net.[6]
- **GSDMM** [51]: This is the DMM topic model inferred by Gibbs sampling. In GSDMM, each document is supposed to cover only a single topic, so as to be applicable to short texts. The code is available on the net.[7]
- **FGSDMM+** [52]: The improved version of GSDMM with online initialization.
- **Para2vec** [23]: This is a paragraph vector model that learns distributed representations of short texts, and then applies *k*-means. The code is available on the net.[8]
- **STC**^{2} [48]: This is a deep learning method inducing short text representations with locality preserving indexing, and then feeding the resulting representations to k-means. The code is available on the net.[9]
- **ST-STC** [13]: This is another self-training deep learning method, which regards the SIF representation [2] as the input. The code is available on the net.[10]
- **A**^{2}**SNMF**: This is our proposed method. For convenience, the versions that construct the affinity matrix by WMD, AWE, and AF-BERT are referred to as **A**^{2}**SNMF+W**, **A**^{2}**SNMF+A**, and **A**^{2}**SNMF+B**, respectively. The other parameter settings are specified as follows: (1) The nearest neighbor number k within the affinity matrix are chosen from $\{10, 20\}$ for Trec and Snippets, and from $\{100, 200\}$ for StackOverFlow, Biomedical, and UCInews; (2) The parameter λ_1 is tuned over the range $\{10^{-2}, 10^{-1}, 10^{0}\}$. (3) The parameter λ_2 are chosen from $\{0.8, 0.9\}$ for Trec and Snippets, and from $\{0.6, 0.7\}$ for StackOverFlow, Biomedical, and UCInews. More empirical analyses are shown in Section 4.2. (4) By referring to [21], the parameter α is fixed to 1.

We further declare that for all comparing methods, the cluster number is set as the true number of labels for each dataset. Besides, for all comparing methods requiring word embeddings, we employ the pre-trained GloVe[11] embeddings [38]. We randomly generate embeddings for the words that don't appear in the GloVe vocabulary.

Metrics. In the experiments, we evaluate the performance by three metrics named ACCuracy (ACC), Normalized Mutual Information (NMI) and Adjusted Rand Index

[4] https://github.com/tshi04/SeaNMF
[5] http://www.cs.columbia.edu/ blei/lda-c/index.html
[6] https://github.com/xiaohuiyan/BTM
[7] https://github.com/jackyin12/GSDMM
[8] https://github.com/mesnilgr/iclr15
[9] https://github.com/jacoxu/STC2
[10] https://github.com/hadifar/stc_clustering
[11] https://nlp.stanford.edu/projects/glove/

(ARI). Considering a dataset of n documents, let $\mathbb{Y} = \{y_d\}_{d=1}^n$ and $\mathbb{C} = \{c_d\}_{d=1}^n$ denote its corresponding label set and resulting cluster set, respectively.

The ACC score measures the correspondence between $\mathbb{Y}$ and $\mathbb{C}$, specifically defined below:

$$\mathbf{ACC} = \frac{\sum_{d=1}^{n} \mathcal{I}\left(y_d, m\left(c_d\right)\right)}{n}, \tag{25}$$

where $\mathcal{I}\left(\cdot\right)$ denotes the indicator function; and $m\left(c_d\right)$ is the mapping function computed by the Hungarian algorithm.

The NMI score measures the mutual information between $\mathbb{Y}$ and $\mathbb{C}$, defined below:

$$\mathbf{NMI} = \frac{MI\left(\mathbb{Y}, \mathbb{C}\right)}{\sqrt{H\left(\mathbb{Y}\right) H\left(\mathbb{C}\right)}}, \tag{26}$$

where $MI\left(\mathbb{Y}, \mathbb{C}\right)$ is the mutual information between $\mathbb{Y}$ and $\mathbb{C}$; $H(\mathbb{Y})$ and $H(\mathbb{C})$ denote the entropy values of $\mathbb{Y}$ and $\mathbb{C}$, respectively; and the denominator plays the role of 0-1 normalization.

Finally, the ARI score is the corrected-for-chance version of the rand index, defined as follows

$$\mathbf{ARI} = \frac{\sum_{ij}^{l} \binom{n_{ij}}{2} - [\sum_{i}^{l} \binom{n_{i\cdot}}{2} \sum_{j}^{l} \binom{n_{\cdot j}}{2}]/\binom{n}{2}}{\frac{1}{2}[\sum_{i}^{l} \binom{n_{i\cdot}}{2} + \sum_{j}^{l} \binom{n_{\cdot j}}{2}] - [\sum_{i}^{l} \binom{n_{i\cdot}}{2} \sum_{j}^{l} \binom{n_{\cdot j}}{2}]/\binom{n}{2}}, \tag{27}$$

where n_{ij} denote the number of documents associated with both label i and cluster j; and $n_{i\cdot}$ and $n_{\cdot j}$ denote the numbers of documents associated with label i and cluster j, respectively.

4.1 Results and Discussions

We independently run each comparing method 5 times and evaluate the average scores of ACC, NMI, and ARI. The experimental results are reported in Table 4. Overall speaking, we can see that A^{2}SNMF performs very competitive with baseline methods in most cases. More details of observations and discussions are given below.

The first observation is that our A^{2}SNMF significantly outperforms the traditional *k*-means and Ncut methods across all datasets. Unsurprisingly, the *k*-means method empirically fails to the clustering of short texts just as the previous reports [51,48], *e.g.,* it only gets about 0.25 and 0.36 ACC scores on Snippets and StackOverFlow, respectively. The results give further empirical evidence that the prototype-based clustering methods may lose effectiveness on the BoW features of short texts, due to the sparsity problem. Besides, an interesting finding is that A^{2}SNMF+W performs better than A^{2}SNMF+A and A^{2}SNMF+B, implying that the distances between short texts can be better measured by WMD.

Second, we can see that A^{2}SNMF consistently outperforms NMF-based methods in all cases. For example, the ACC scores of A^{2}SNMF are even about $0.06 \sim 0.18$ and $0.04 \sim 0.09$ higher than those of NMF-based methods across Trec and Biomedical, respectively. Among those NMF-based ones, we roughly find that the performance order is NMF $<$ SeaNMF $<$ SymNMF. The results seem reasonable: (1) The SeaNMF

Table 4 The clustering results (mean±standard deviation) of ACC, NMI and ARI. The best performance is given in boldface type.

Metric	Methods	*Trec*	*Snipptes*	*StackOverFlow*	*Biomedical*	*UCInews*
ACC	***k*-means**	0.42±0.04	0.25±0.01	0.36±0.04	0.21±0.02	0.49±0.03
	Ncut [41]	0.49±0.03	0.64±0.02	0.53±0.01	0.32±0.02	0.61±0.02
	NMF [11]	0.46±0.00	0.61±0.03	0.66±0.02	0.36±0.00	0.65±0.01
	SymNMF [20]	0.54±0.02	0.76±0.07	0.65±0.02	0.39±0.01	0.67±0.02
	SeaNMF [42]	0.55±0.02	0.56±0.02	0.66±0.04	0.39±0.01	0.66±0.03
	LDA [4]	0.33±0.01	0.49±0.05	0.41±0.03	0.27±0.02	0.58±0.03
	BTM [9]	0.36±0.01	0.61±0.02	0.49±0.05	0.36±0.03	0.63±0.02
	GSDMM [51]	0.36±0.01	0.62±0.03	0.48±0.04	0.36±0.02	0.64±0.02
	FGSDMM+ [52]	0.37±0.01	0.63±0.03	0.49±0.04	0.37±0.02	0.63±0.01
	Para2vec [23]	0.51±0.02	0.68±0.04	0.33±0.02	0.42±0.01	0.62±0.02
	STC^2 [48]	0.54±0.03	0.78±0.05	0.53±0.04	0.42±0.02	0.61±0.03
	ST-STC [13]	0.58±0.02	0.77±0.04	0.59±0.03	**0.49±0.02**	0.65±0.02
	A^{2}SNMF+W	**0.64±0.02**	**0.81±0.07**	0.67±0.01	0.45±0.02	**0.69±0.01**
	A^{2}SNMF+A	0.63±0.01	0.80±0.04	**0.68±0.03**	0.43±0.02	**0.69±0.02**
	A^{2}SNMF+B	0.63±0.02	0.78±0.02	0.67±0.03	0.45±0.02	**0.69±0.02**
Metric	**Methods**	*Trec*	*Snipptes*	*StackOverFlow*	*Biomedical*	*UCInews*
NMI	***k*-means**	0.34±0.03	0.10±0.04	0.45±0.04	0.22±0.02	0.66±0.02
	Ncut [41]	0.37±0.02	0.48±0.03	0.54±0.01	0.28±0.00	0.72±0.01
	NMF [11]	0.27±0.01	0.41±0.02	0.54±0.01	0.30±0.01	0.76±0.02
	SymNMF [20]	0.38±0.02	0.58±0.03	0.56±0.01	0.34±0.00	0.76±0.01
	SeaNMF [42]	0.40±0.01	0.38±0.02	0.56±0.05	0.32±0.01	0.75±0.03
	LDA [4]	0.11±0.01	0.33±0.05	0.37±0.06	0.26±0.03	0.66±0.02
	BTM [9]	0.16±0.02	0.52±0.03	0.46±0.05	0.34±0.02	0.73±0.02
	GSDMM [51]	0.14±0.01	0.51±0.04	0.44±0.05	0.33±0.02	0.74±0.02
	FGSDMM+ [52]	0.15±0.01	0.53±0.05	0.44±0.04	0.34±0.02	0.72±0.01
	Para2vec [23]	0.39±0.01	0.52±0.01	0.29±0.02	0.35±0.01	0.73±0.02
	STC^2 [48]	0.44±0.04	0.61±0.01	0.51±0.02	0.37±0.01	0.73±0.02
	ST-STC [13]	0.45±0.02	0.60±0.02	0.54±0.03	**0.42±0.02**	0.75±0.03
	A^{2}SNMF+W	**0.47±0.03**	**0.67±0.01**	**0.58±0.02**	0.37±0.01	**0.79±0.02**
	A^{2}SNMF+A	0.46±0.02	**0.67±0.03**	0.56±0.01	0.36±0.01	0.78±0.01
	A^{2}SNMF+B	0.44±0.01	0.62±0.01	0.57±0.01	0.37±0.01	**0.79±0.03**
Metric	**Methods**	*Trec*	*Snipptes*	*StackOverFlow*	*Biomedical*	*UCInews*
ARI	***k*-means**	0.13±0.02	0.01±0.01	0.06±0.02	0.03±0.01	0.22±0.03
	Ncut [41]	0.20±0.02	0.38±0.02	0.13±0.01	0.12±0.01	0.43±0.03
	NMF [11]	0.23±0.00	0.39±0.02	0.48±0.02	0.21±0.01	0.58±0.02
	SymNMF [20]	0.32±0.02	0.59±0.02	0.32±0.02	0.22±0.02	0.57±0.03
	SeaNMF [42]	0.31±0.01	0.31±0.02	0.46±0.04	0.20±0.02	0.57±0.02
	LDA [4]	0.06±0.00	0.26±0.04	0.24±0.03	0.15±0.01	0.47±0.02
	BTM [9]	0.09±0.00	0.38±0.03	0.29±0.04	0.22±0.03	0.54±0.01
	GSDMM [51]	0.07±0.01	0.39±0.03	0.28±0.04	0.21±0.03	0.54±0.02
	FGSDMM+ [52]	0.08±0.01	0.40±0.02	0.29±0.05	0.21±0.04	0.53±0.02
	Para2vec [23]	0.29±0.01	0.43±0.02	0.06±0.01	0.23±0.03	0.47±0.03
	STC^2 [48]	0.34±0.05	0.57±0.03	0.42±0.02	0.25±0.01	0.55±0.02
	ST-STC [13]	0.37±0.03	0.55±0.02	0.44±0.04	**0.29±0.01**	0.56±0.01
	A^{2}SNMF+W	**0.42±0.02**	**0.67±0.05**	**0.51±0.03**	0.27±0.03	0.60±0.03
	A^{2}SNMF+A	0.41±0.02	0.65±0.03	**0.51±0.01**	0.26±0.02	**0.61±0.02**
	A^{2}SNMF+B	0.39±0.02	0.60±0.03	0.50±0.01	0.28±0.02	**0.61±0.02**

method employs the global word co-occurrences of the whole dataset, enabling to effectively alleviate the sparsity problem of short texts [9,27]. (2) The SymNMF method is on par with other baseline methods, implying that directly inducing clusters from the affinity information of short texts may be more beneficial for clustering than using the manifold regularization. Additionally, we would like to emphasize the performance improvements of A^{2}SNMF over SymNMF. This phenomenon indicates the effectiveness of the near-outlier aware mechanism of A^{2}SNMF.

Third, compared with topic modeling methods, we also observe significant improvements achieved by A^{2}SNMF on all datasets. For example, the ACC and NMI scores of A^{2}SNMF are about 0.16 and 0.09 higher than those of GSDMM on Snippets. Broadly speaking, this GSDMM method focuses on solving the sparsity prob-

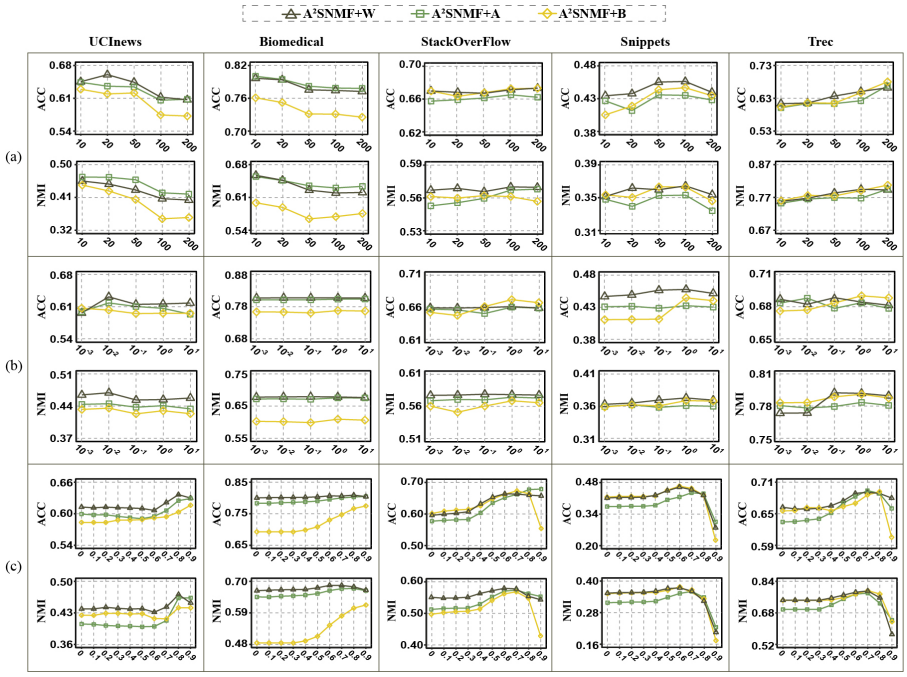

Fig. 3 The ACC and NMI scores with different values of (a) k, (b) λ_1, and (c) λ_2.

lem by constraining that each short text only covers a single topic, however, it neglects the noisy nature of short texts. That is the possible reason for the improvement of A^{2}SNMF.

Finally, we can see that A^{2}SNMF is also on a par with the most recent deep learning methods, *i.e.,* STC^2 and ST-STC. That is, our A^{2}SNMF outperforms STC^2 and ST-STC in 14/15 and 12/15 cases, respectively. The improvements of the ACC scores are about $0.11 \sim 0.17$ and $0.02 \sim 0.07$ across StackOverFlow and UCInews, respectively. These results indicate that directly factorizing the affinity matrix by shallow techniques can be also competitive with deeper methods for short text clustering. Therefore, designing affinity matrix factorization with deeper ones may be a very potential research subject in the future.

4.2 Sensitivity Analysis of Parameters

In this subsection, we evaluate the sensitivity of three significant parameters of A^{2}SNMF, including the nearest neighbor number k and two regularization parameters λ_1, λ_2. We plot the ACC and NMI results with different values of those three parameters in Fig.3

The parameter k controls the number of neighboring similarity values, *i.e.,* non-zero elements, in the affinity matrix. We examine its clustering performance with different values from the following range $\{10, 20, 50, 100, 200\}$. We observe that the datasets with fewer labels, *i.e.,* Trec and Snippets, perform better when $k \in$

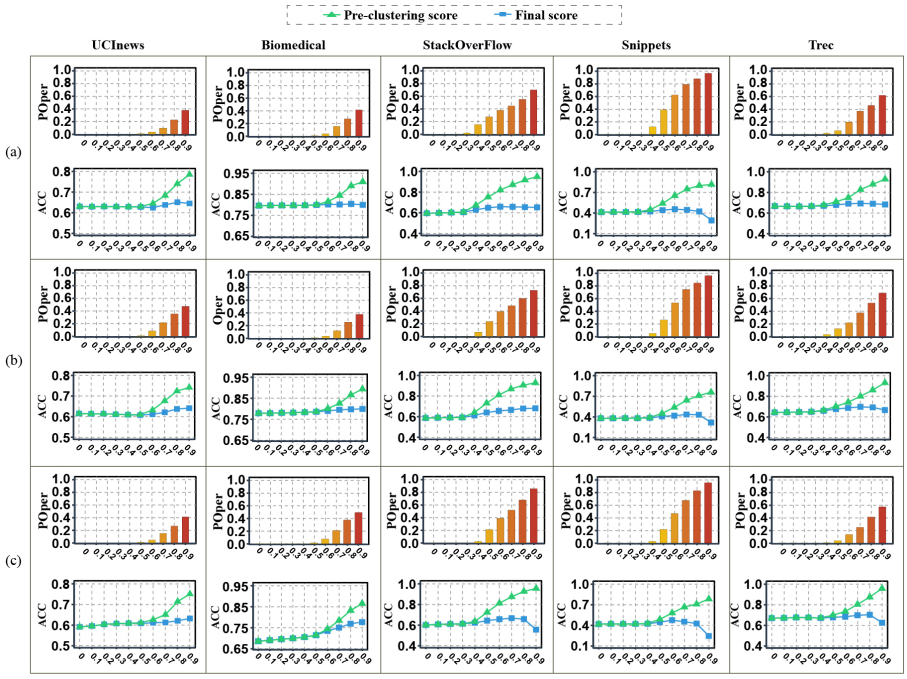

Fig. 4 The Pseudo-Outlier percentages (POper), ACC scores of per-clustering, and final ACC scores with different values of λ_2 by applying (a) **A^{2}SNMF+W**, (b) **A^{2}SNMF+A**, and (c) **A^{2}SNMF+B**.

$\{10, 20\}$, and the ones with relatively more labels, *i.e.,* StackOverFlow, Biomedical, and UCInews, perform better when $k \in \{100, 200\}$. We kindly argue the results are reasonable. For example, Trec only contains 6 labels and thus larger k values may result in many neighbors belonging to different labels in the affinity matrix for each short text. By contrast, UCInews has more labels, so it requires more neighbors to ensure the affinity matrix contains sufficient neighbors with the same labels for each short text.

The regularization parameter λ_1 corresponds to the ℓ_1 norm regularizer of noisy matrix (*i.e.,* $\|\mathbf{Z}\|_1$). The larger value of λ_1, the sparser the noisy matrix is computed. We evaluate it with different values in the range of $\{10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1\}$. Overall, we can observe that the clustering scores of A^{2}SNMF are sightly higher when $\lambda_1 \in \{10^{-2}, 10^{-1}, 10^0\}$. Those results directly indicate incorporating the regularizer with respect to the noisy matrix is beneficial for the final clustering. Even A^{2}SNMF performs relatively stable as λ_1 varies in some cases. That is because they contain fewer noisy similarity scores in the affinity matrix within optimal settings of k. We can safely suggest the tuning range $\{10^{-2}, 10^{-1}, 10^0\}$ in applications.

The regularization parameter λ_2 corresponds to the $\ell_{2,1}$ norm regularizer of cluster assignment matrix (*i.e.,* $\|\mathbf{C}\|_{2,1}$). Reviewing the update rule of Eq.12, it is used to judge the pseudo-outliers in some sense, where larger λ_2, more pseudo-outliers will be eliminated during pre-clustering. We evaluate it with different values over the range $\{0.1, 0.2, \cdots, 0.9\}$. We observe that the datasets with fewer labels, *i.e.,* Trec and Snippets, perform better when $\lambda_2 \in \{0.8, 0.9\}$, and the ones with rela-

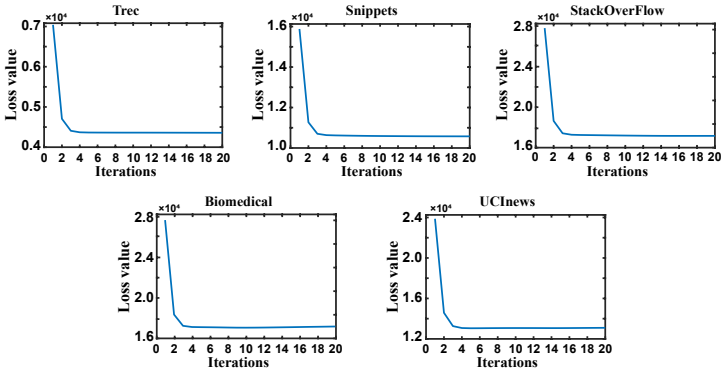

Fig. 5 The convergence curves in the pre-clustering stage.

Table 5 The per-iteration running time of the pre-clustering stage.

Dataset	Running time (seconds)
Trec	0.94
Snippts	6.52
StackOverFlow	18.83
Biomedical	23.57
UCInews	8.29

tively more labels, *i.e.,* StackOverFlow, Biomedical, and UCInews, perform better when $\lambda_2 \in \{0.6, 0.7\}$. We consider the reason is that A^{2}SNMF requires to remove an "appropriate" percentage of near-outliers for accurate clustering results in the pre-clustering stage, and by referring to Eq.12, higher λ_2 value is required for removal of near-outliers with less labels. To further analyze the sensitivity of λ_2, we further plot the percentage of pseudo-outliers, ACC scores of pre-clustering and final ACC scores, as shown in Fig.4. It can bee seen that the better performance happen when the percentage of pseudo-outliers is always about $0.4 \sim 0.6$ across all datasets.

4.3 Convergence Analysis of Pre-clustering

We evaluate the convergence property of the pre-clustering stage. Fig.5 plots the convergence curves[12] of A^{2}SNMF+W on all datasets. It can be clearly seen that A^{2}SNMF fast converges to a stationary point in very few iterations. The quick convergence property indicates the effectiveness and efficiency of A^{2}SNMF in solving the pre-clustering stage, and it guarantees the efficiency of A^{2}SNMF in real clustering applications.

4.4 Runtime of Pre-clustering

We examine the running time of the pre-clustering stage. To this end, we run the stage 100 iterations regardless of the results, and report the per-iteration running time on

[12] The convergence curves of A^{2}SNMF+A and A^{2}SNMF+B are almost the same, so we omit them here.

all datasets.[13] As shown in Table 5, we can observe that each iteration of A^{2}SNMF takes a little time, indicating its efficiency.

5 Conclusion

In this paper, we target improving the clustering performance on short text, which has already become a fashionable form of text information. By revising the previous studies, we observed that the methods can benefit from the similarity values of neighboring short texts. Therefore, we propose affinity matrix factorization as the proposal for short text clustering. However, we also found that the similarity values are to some extent inaccurate, due to the noisy nature of short texts. To remedy this problem, we propose a novel A^{2}SNMF method, enhancing the affinity matrix factorization by incorporating a sparse noisy matrix regularizing the cluster assignment matrix by $\ell_{2,1}$ norm. We empirically compare A^{2}SNMF with several representative methods on five commonly used short text datasets. The reported results show the superior performance of A^{2}SNMF. In the future, extending A^{2}SNMF with deeper model structures is a potential investigation.

Appendix

We now show the optimization details of the sub-problem of $\mathbf{C}$ (*i.e.,* Eq.10). We consider its generic form as follows:

$$\min_{\mathbf{C}\succeq\mathbf{0}} \frac{1}{2}\|\mathbf{C}-\mathbf{A}\|_F^2 + \lambda\|\mathbf{C}\|_{2,1}, \tag{28}$$

where $\mathbf{A}$ denotes any constant matrix and λ is a fixed regularization parameter. Obviously, this objective can be divided into a number of independent sub-objectives with respect to the rows of $\mathbf{C}$, where each one is formulated as follows

$$\min_{\mathbf{C}_i\succeq\mathbf{0}} \frac{1}{2}\|\mathbf{C}_i-\mathbf{A}_i\|^2 + \lambda\|\mathbf{C}_i\| \tag{29}$$

Theorem 1 *Let* $\mathbf{A}_i^+ = [\mathbf{A}_i]_+$, *where* $[\,\cdot\,]_+ = \max(\,\cdot\,,\mathbf{0})$. *The optimal solution of Eq.29 is given below:*

$$\mathbf{C}_i = \begin{cases} \left(1-\frac{\lambda}{\|\mathbf{A}_i^+\|}\right)\mathbf{A}_i^+, \text{ if } \|\mathbf{A}_i^+\| > \lambda \\ \mathbf{0}, \qquad\qquad \text{otherwise} \end{cases}, \tag{30}$$

Proof Considering the minimization problem of Eq.29 with inequality constraints, we define its Lagrangian as follows:

$$\mathcal{L}(\mathbf{C}_i,\phi) = \frac{1}{2}\|\mathbf{C}_i-\mathbf{A}_i\|^2 + \lambda\|\mathbf{C}_i\| - \phi^\top\mathbf{C}_i, \tag{31}$$

[13] The pre-clustering running times of all three versions of A^{2}SNMF are the same.

where $\phi \succeq \mathbf{0}$ denotes the vector of Lagrange multipliers. Let $\mathbf{C}_i^*$ denote the optimal solution to Eq.31. By referring to [31], it can be directly given by:

$$\mathbf{C}_i^* = \begin{cases} \left(1 - \frac{\lambda}{\|\widehat{\mathbf{A}}_i\|}\right) \widehat{\mathbf{A}}_i, \text{ if } \|\widehat{\mathbf{A}}_i\| > \lambda \\ \mathbf{0}, \qquad\qquad \text{otherwise} \end{cases}, \tag{32}$$

where $\widehat{\mathbf{A}}_i = \mathbf{A}_i + \phi$. Following the KKT conditions, we have the following:

$$\mathbf{C}_i^* \succeq \mathbf{0}, \tag{33}$$

$$\phi \otimes \mathbf{C}_i^* = \mathbf{0}, \tag{34}$$

where $\otimes$ denotes the element-wise product operation. To satisfy the above conditions, we can only consider the case of $\widehat{\mathbf{A}}_i = \mathbf{A}_i^+ = [\mathbf{A}_i]_+$, so that each element of ϕ is given as follows:

$$\phi_j = \begin{cases} -\mathbf{A}_{ij}, \text{ if } \mathbf{A}_{ij} < 0 \\ 0, \qquad \text{otherwise} \end{cases}, \tag{35}$$

When $\|\mathbf{A}_i^+\| > \lambda$, the optimal solution must be given by $\mathbf{C}_i^* = \left(1 - \frac{\lambda}{\|\mathbf{A}_i^+\|}\right) \mathbf{A}_i^+$ according to Eq.32. And the condition of Eq.34 is obviously satisfied according to Eq.35. **When** $\|\mathbf{A}_i^+\| \leq \lambda$, the optimal solution must be as $\mathbf{C}_i^* = \mathbf{0}$, so that the condition of Eq.34 can be satisfied. Combining Eq.32 with Eq.35, the proof can be finished.

Declarations

Funding We would like to acknowledge support for this project from the National Natural Science Foundation of China (No.51805203), China Postdoctoral Science Foundation (No.2019M651123), Natural Science basic research project (for universities of Liaoning) of Liaoning Provincial Department of Education, China (No.LJKZ0986), and Science and Technology Innovation Found (Youth Science and Technology Star) of Dalian, China (No.2018RQ65).

Conflicts of interest The authors declare that they have no conflict of interest.

Ethics approval Not Applicable

Consent to participate Not Applicable

Consent for publication Not Applicable

Availability of data Trec is publicly available at http://cogcomp.cs.illinois.edu/Data/QA/QC/; Snippets is publicly available at http://jwebpro.sourceforge.net/data-web-snippets.tar.gz; StackOverFlow is publicly available at https://github.com/jacoxu/STC2; Biomedical is publicly available at http://participants-area.bioasq.org/; and UCInews is publicly available at https://www.kaggle.com/uciml/news-aggregator-dataset.

Code availability The source code is implemented by Matlab. We will release the code soon.

Author contributions All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by XL and YG. The first draft of the manuscript was written by XL, BF, and ZL and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

References

1. Angiulli, F., Basta, S., Pizzuti, C.: Distance-based detection and prediction of outliers. IEEE Transactions on Knowledge and Data Engineering **18**(2), 145–160 (2006)
2. Arora, S., Liang, Y., Ma, T.: A simple but tough-to-beat baseline for sentence embeddings. In: International Conference on Learning Representations (2017)
3. Blei, D.M.: Probabilistic topic models. Communications of the ACM **55**(4), 77–84 (2012)
4. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. Journal of Machine Learning Research **3**, 993–1022 (2003)
5. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends in Machine Learning **3**(1), 1–122 (2011)
6. Cai, D., He, X., Han, J., Huang, T.S.: Graph regularized non-negative matrix factorization for data representation. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**(8), 1548–1560 (2011)
7. Cai, D., He, X., Wu, X., Han, J.: Non-negative matrix factorization on manifold. In: IEEE International Conference on Data Mining (2008)
8. Chawla, S., Gionis, A.: k-means-: A unified approach to clustering and outlier detection. In: SIAM International Conference on Data Mining (2013)
9. Cheng, X., Yan, X., Lan, Y., Guo, J.: BTM: Topic modeling over short texts. IEEE Transactions on Knowledge and Data Engineering **26**(12), 2928–2041 (2014)
10. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 4171–4186 (2019)
11. Ding, C., He, X., Simon, H.D.: On the equivalence of nonnegative matrix factorization and spectral clustering. In: SIAM International Conference on Data Mining, pp. 606–610 (2005)
12. Guan, H., Zhou, J., Guo, M.: A class-feature-centroid classifier for text categorization. In: International Conference on World Wide Web, pp. 201–210 (2009)
13. Hadifar, A., Sterckx, L., Demeester, T., Develder, C.: A self-training approach for short text clustering. In: Workshop on Representation Learning for Natural Language Processing, pp. 194–199 (2019)
14. Hu, X., Sun, N., Zhang, C., Chua, T.S.: Exploiting internal and external semantics for the clustering of short texts using world knowledge. In: ACM Conference on Information and Knowledge Management, pp. 919–928 (2009)
15. Jia, C., Carson, M.B., Wang, X., Yu, J.: Concept decompositions for short text clustering by identifying word communities. Pattern Recognition **76**, 691–703 (2018)
16. Jia, Y., Liu, H., Hou, J., Kwong, S.: Clustering-aware graph construction: A joint learning perspective. IEEE Transactions on Signal and Information Processing over Networks **6**, 357–370 (2013)
17. Jia, Y., Liu, H., Hou, J., Kwong, S.: Semisupervised adaptive symmetric non-negative matrix factorization. IEEE Transactions on Cybernetics **51**(5), 2550–2562 (2020)
18. Kannan, R., Woo, H., Aggarwaland, C.C., Park, H.: Outlier detection for text data. In: SIAM International Conference on Data Mining, pp. 489–497 (2017)
19. Kim, J., He, Y., Park, H.: Algorithms for nonnegative matrix and tensor factorizations: a unified view based on block coordinate descent framework. Journal of Global Optimization **58**(2), 285–319 (2014)
20. Kuang, D., Ding, C., Park, H.: Symmetric nonnegative matrix factorization for graph clustering. In: SIAM International Conference on Data Mining, pp. 106–117 (2012)
21. Kuang, D., Yun, S., Park, H.: SymNMF: nonnegative low-rank approximation of a similarity matrix for graph clustering. Journal of Global Optimization **62**(3), 545–574 (2015)
22. Kusner, M.J., Sun, Y., Kolkin, N.I., Weinberger, K.Q.: From word embeddings to document distances. In: International Conference on Machine Learning, pp. 957–966 (2015)
23. Le, Q., Mikolov, T.: Distributed representations of sentences and document. In: International Conference on Machine Learning, pp. 1188–1196 (2014)
24. Lee, D.D., Seung, H.S.: Learning the parts of objects by non-negative matrix factorization. Nature **401**(6755), 788–791 (1999)
25. Li, C., Ouyang, J., Li, X.: Classifying extremely short texts by exploiting semantic centroids in word mover's distance space. In: The Web Conference, pp. 838–949 (2019)
26. Li, C., Wang, H., Zhang, Z., Sun, A., Ma, Z.: Topic modeling for short texts with auxiliary word embeddings. In: International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 165–174 (2016)

27. Li, X., Zhang, A., Li, C., Guo, L., Wang, W., Ouyang, J.: Relational biterm topic model: Short-text topic modeling using word embeddings. The Computer Journal **62**(3), 359–372 (2019)
28. Li, X., Zhang, A., Li, C., Ouyang, J., Cai, Y.: Exploring coherent topics by topic modeling with term weighting. Information Processing & Management **54**(6), 1345–1358 (2018)
29. Li, X., Zhang, J., Ouyang, J.: Dirichlet multinomial mixture with variational manifold regularization: Topic modeling over short texts. In: AAAI Conference on Artificial Intelligence, pp. 7884–7891 (2019)
30. Liu, H., Li, J., Wu, Y., Fu, Y.: Clustering with outlier removal. IEEE Transactions on Knowledge and Data Engineering (Early Access) (2019)
31. Liu, J., Ji, S., Ye, J.: Multi-task feature learning via efficient l2,1-norm minimization. In: Conference on Uncertainty in Artificial Intelligence, pp. 339–348 (2009)
32. Luo, X., Liu, Z., Jin, L., Zhou, Y., Zhou, M.: Symmetric nonnegative matrix factorization-based community detection models and their convergence analysis. IEEE Transactions on Neural Networks and Learning Systems pp. 1–13 (2021). DOI 10.1109/TNNLS.2020.3041360
33. Luo, X., Liu, Z., Li, S., Shang, M., Wang, Z.: A fast non-negative latent factor model based on generalized momentum method. IEEE Transactions on Systems, Man, and Cybernetics: Systems **51**(1), 610–620 (2021)
34. Luo, X., Liu, Z., Shang, M., Lou, J., Zhou, M.: Highly-accurate community detection via pointwise mutual information-incorporated symmetric non-negative matrix factorization. IEEE Transactions on Network Science and Engineering **8**(1), 463–476 (2021)
35. Luo, X., Zhou, M., Li, S., Wu, D., Liu, Z., Shang, M.: Algorithms of unconstrained non-negative latent factor analysis for recommender systems. IEEE Transactions on Big Data **7**(1), 227–240 (2021)
36. Otey, M.E., Ghoting, A., Parthasarathy, S.: Fast distributed outlier detection in mixed-attribute data sets. Data Mining and Knowledge Discovery **12**(2–3), 203–228 (2006)
37. Ott, L., Pang, L., Ramos, F.T., Chawla, S.: On integrated clustering and outlier detection. In: Neural Information Processing Systems, pp. 1359–1367 (2014)
38. Pennington, J., Socher, R., Manning, C.D.: GloVe: global vectors for word representation. In: Empirical Methods in Natural Language Processing, pp. 1532–1543 (2014)
39. Seifzadeh, S., Farahat, A.K., Kamel, M.S., Karray, F.O.: Short-text clustering using statistical semantics. In: International Conference on World Wide Web, pp. 805–810 (2015)
40. Shang, M., Luo, X., Liu, Z., Chen, J., Yuan, Y., Zhou, M.: Randomized latent factor model for high-dimensional and sparse matrices from industrial applications. IEEE Transactions on Systems, Man, and Cybernetics: Systems **6**(1), 131–141 (2019)
41. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **22**(8), 888–905 (2000)
42. Shi, T., Kang, K., Choo, J., Reddy, C.K.: Short-text topic modeling via non-negative matrix factorization enriched with local word-context correlations. In: The Web Conference, pp. 1105–1114 (2018)
43. Wang, X., McCallum, A.: Topics over time: A non-markov continuous-time model of topical trends. In: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 424–433 (2006)
44. Wu, S., Wang, S.: Information-theoretic outlier detection for large-scale categorical data. IEEE Transactions on Knowledge and Data Engineering **25**(3), 589–602 (2013)
45. Wu, W., Jia, Y., Kwong, S., Hou, J.: Pairwise constraint propagation-induced symmetric nonnegative matrix factorization. IEEE Transactions on Neural Networks and Learning Systems **29**(12), 6348–6361 (2013)
46. Xie, J., Girshick, R.B., Farhadi, A.: Unsupervised deep embedding for clustering analysis. In: International Conference on Machine Learning, pp. 478–487 (2016)
47. Xu, J., Wang, P., Tian, G., Xu, B., Zhao, J., Wang, F., Hao, H.: Short text clustering via convolutional neural networks. In: Workshop on Vector Space Modeling for Natural Language Processing, pp. 62–69 (2015)
48. Xu, J., Xu, B., Wang, P., Zheng, S., Tian, G., Zhao, J., Xua, B.: Self-taught convolutional neural networks for short text clustering. Neural Networks **88**, 22–31 (2017)
49. Yan, Y., Cao, L., Rundensteiner, E.A.: Scalable top-n local outlier detection. In: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1235–1244 (2017)
50. Yin, J., Chao, D., Liu, Z., Zhang, W., Yu, X., Wang, J.: Model-based clustering of short text streams. In: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 2634–2642 (2018)
51. Yin, J., Wang, J.: A Dirichlet multinomial mixture model-based approach for short text clustering. In: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 233–242 (2014)

52. Yin, J., Wang, J.: A text clustering algorithm using an online clustering scheme for initialization. In: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1995–2004 (2016)
53. Zhang, W., Dong, C., Yin, J., Wang, J.: Attentive representation learning with adversarial training for short text clustering. IEEE Transactions on Knowledge and Data Engineering (Early Access) (2021)