IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 22, 2020, accepted November 23, 2020, date of publication December 7, 2020,
date of current version December 28, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3042778

Short Text Embedding Autoencoders With Attention-Based Neighborhood Preservation

CHAO WEI, LIJUN ZHU, AND JIAOXIANG SHI
National Engineering Research Center of Science and Technology Information, Institute of Scientific and Technical Information of China, Beijing 100038, China
Corresponding author: Chao Wei (weichaolx@gmail.com)

This work was supported by the Postdoctoral Science Foundation Granted of China under Grant 2019M650804.

ABSTRACT Shortness and sparsity often plague short text representation for clustering and classification. A popular solution is to extract meaningful low-dimensional embeddings as short text representation via various Dimensionality Reduction technology. However, the existing methods, such as topic models and neural networks, discover low-dimensional embeddings from the whole training sets without considering the geometrical information of short text manifold, resulting in an inability to provide a discriminative embedding of short text. In this paper, we propose a manifold-regularized method, namely Short Texts Embedding AutoEncoders (STE-AEs), aiming to incorporate the semantics from the neighborhood into a regularization training of AutoEncoders (AEs) to extract discriminative low-dimensional short text embeddings. STE-AEs first determines semantics neighborhood via an attention-based weighted matching distance and then preserves the local geometrical structure by incorporating a minimization of the weighted cross-entropy of nearby texts' embeddings into a regularization training of AEs. Finally, the encoder can act as a parametrized mapping function between observations and embeddings. Furthermore, based on the activation values of the encoder for the training set, STE-AEs employs a regression model of Random Forest (RF) to determine the feature importance so as to find certain informative and readable words for embeddings interpretation. Through extensive experiments on three real-world short text corpuses, the evidence demonstrate that STE-AEs can capture the intrinsic discriminative explanatory factors, improving the performance of short text clustering and classification. Moreover, some understandable words can be efficiently discovered to promote the interpretability of low-dimensional embeddings.

INDEX TERMS Short text representation, low-dimensional embeddings, manifold-regularized AutoEncoders, attention-based weighted matching.

I. INTRODUCTION

Short texts, including Tweets, search snippets, FAQs, product comments, and scientific abstracts, etc., are now widespread on the internet. Therefore, automatically analyzing the semantics of short texts is fundamental for a wide range of downstream NLP tasks, like Readers' Emotion Classification [1], Entity Disambiguation [2], Topic Evolution Mining [3] and Short Texts Sentiment Analysis [4]. However, due to the scarcity of word co-occurrence patterns in one short text, and a wide range of vocabulary spanned over a corpus, traditional texts representation models, such as $tf - idf$, often suffer from the serious data-sparse issue, resulting in performance degradation in clustering and classification task.

The associate editor coordinating the review of this manuscript and approving it for publication was Arianna Dulizia.

Existing short text representation works tend to address the data-sparse issue from two aspect. A simple solution is to expand short text as lengthy pseudo-text based on some external knowledge base (or knowledge graph), like WordNet, MeSH, Wikipedia, Open Directory Project, and handle the extension text via traditional text representation methods. For example, Hu *et al.* utilize Wikipedia concept and category information to improve semantic information of short texts representation [5]. Recently, Song *et al.* develop a bayesian inference mechanism to incorporate the semantics behind knowledge base (e.g., WordNet, Freebase and Wikipedia) for short texts conceptualization [6]. Huang *et al.* introduce a co-ranking framework to extract contextual keywords and combine the keywords with the attention-based strategy for short-text embedding [7]. Knowledge-guided Non-negative Matrix Factorization (KGNMF) is proposed for better short text classification by leveraging external knowledge as a

semantic regulator with low-rank formalizations [8]. Li *et al.* propose a combined method based on knowledge-based conceptualization and a transformer encoder for short text understanding [9]. Although they can relieve the sparsity of short text representation than the conventional model, the expand-based way is not an ideal solution because of the heavy dependence on the accuracy and completeness of the constructed knowledge base, which is also a challenging task [10].

Alternatively, some works aim to reduce the variables set and assemble embedded variables as low-dimensional representations (or embeddings), which is benefit from such variables may reveal the different explanatory factors of variation embedded in the texts [11]. In the past decades, some famous topic-based solutions have been proposed to extract low-dimensional meaningful embeddings, such as probabilistic Latent Semantic Analysis (pLSA) [12] and Latent Dirichlet allocation (LDA) [13]. These models conceptualize each text as a list of mixing proportions of latent topics and interpreting each topic as a distribution of vocabulary [14]. Following the idea of topic model, a biterm topic model (BTM) consider the dependency between latent topics (embedded variables) and an unordered word-pair occurrence patterns in a local context so as to discover informative embeddings of short texts [15]. Besides, Zuo *et al.* propose a novel probabilistic model called Pseudo-document-based Topic Model (PTM) for short text topic modeling which introduces the concept of pseudo document to implicitly aggregate short texts against data sparsity [16]. Moreover, various Neural-Networks-based approaches adopt a count vector of char, word, sentence or paragraph as input and compound into an intermediate low-dimensional embedding, providing a promising ability to uncover the semantics [17], [18]. Since then, Cedric *et al.* provide a learning procedure based on a novel median-based loss function to weighty aggregate word embedding for short text embedding [19]. Xu *et al.* incorporate context-relevant concepts into a convolu-tional neural network (CNN), called DE-CNN, for short text classification [20]. Different from the topic-based models, the intermediate low-dimensional embeddings are named as distributed representation, due to the fact that feeding signals are distributed over each independently activated neuron, which can provide different level embedded variables for text representation. However, the distributed representation brings a challenge to understand the specific meaning of each dimension.

In general, both topic models and neural networks are embedded with latent factors [21], preserving the salient statistical structure of intra-texts. Despite an improvement on short text representation, such methods take a global perspective that observation space as Euclidean to discover the embedded explanatory factors, resulting in the strong dependency of embeddings to all texts and the non-discriminatory of all texts. This will undoubtedly affect the effective capture of intrinsic discriminative explanatory factors. In fact, numerous studies demonstrate that natural observations, such as texts and images, concentrate in the vicinity of a smooth lower-dimensional manifold[1] (or manifold hypothesis), indicating the local geometrical structure of text manifold may be worthy of attention for text embedding [22]–[24]. In other words, nearby texts (or neighborhood) have a strong semantic dependency. For example, nearby texts tend to have more co-occurrence patterns of related words, which implies that they may express more similar semantics and closer relationships of their embeddings. Therefore, the preservation of neighborhood semantic dependency may have a positive effect on capturing the intrinsic discriminative explanatory factors. From this perspective, we propose a manifold-regularized Short Texts Embedding AutoEncoders (STE-AEs), aiming to regularize the training of AutoEncoders (AEs) by imposing an additional minimization of the cross-entropy of nearby texts' embeddings to preserve the local geometrical pattern of neighborhood. Our main contributions are summarized as follows:

- Under the regularized training framework, STE-AEs can provide an explicit parametrized mapping function between observations and embeddings, ensuring the embeddings are local invariant around the neighborhoods and improves the discriminative effect. Meanwhile, taking the local perspective will allow STE-AEs to incorporate the semantics of neighborhoods when embedding a given input text, which can enrich the semantics of single short text against the issue of data-sparse.
- Based on the manifold learning framework, the k-nearest-neighborhood (KNN) graph is used to depict the manifold structure. To avoid the impact of the sparsity on the similarity measurement of nearby short text, STE-AEs develops an attention-based weighted matching distance (AWMD), providing a robust similarity mea-surement even for two texts with similar semantics but low co-occurrence vocabulary.
- Finally, to better understand the meaning of low-dimensional embeddings, STE-AEs develops a post-processing solution that adopts a regression model of RF to determine some informative and understandable words using the activation values of the encoder and the training set, which provides a useful attempt to understand the low-dimensional embeddings based on neural networks.

II. RELATED WORK

In this section, our survey includes two lines of literature most relevant to our work: short text embedding and manifold learning, for a better understanding of the existing methods and challenges.

[1]In mathematics, manifold is interpreted as a topological space that resembles Euclidean space near each point, called locally Euclidean or local consistency.

A. SHORT TEXT EMBEDDING

The embedding is a general term for a class of technologies, which can transform objects in a high-dimensional space into a relatively low-dimensional space for representation. Different from the traditional one-hot representation, it can provide continuous values vectors, and in an ideal state, it can achieve good maintenance of the semantic structure of the object. Due to the shortness and sparsity of short texts, to apply the embedding technology for short text representation, there are two aspects need to be modified. One is to introduce external knowledge with rich semantics, and the other is to improve the ability of embedding technology to capture the inner semantics of short texts [25].

A common solution is to introduce external knowledge (i.e., WordNet, Wikipedia, Open Directory Project) to expand short text as lengthy pseudo-text and then adopt traditional text embedding methods to represent the pseudo-texts [5], [26]. Some subsequent works mostly adopt the semantically rich knowledge to formulate short texts as a series of conceptual word sets [6], [7], [27]. For example, Song *et al.* develop a bayesian inference mechanism to conceptualize short text with knowledgebase (e.g., WordNet, Freebase, and Wikipedia) for clustering [6]. Recently, [7] and [27] first model the short text as a set of relevant concepts using a large taxonomy knowledge, and then adopt different neural network frameworks for short texts embedding. KGNMF is proposed for better short text classification by leveraging external knowledge as a semantic regulator with low-rank formalizations [8]. Li *et al.* enrich the short text information from a knowledge base based on cooccurrence terms and concepts and embed these concepts into a low-dimensional vector space via a convolutional neural network (CNN) and a subnetwork based on a transformer embedding encoder [9].

Additionally, in the last decades, various embedding technologies aim to improve the deconstruction effect of the semantic structure in the text, like Topic-based and Neural-Networks-based methods [12], [13], [28], [29]. The main difference between them is that Topic-based methods belong to generative models, while Neural-Network-based methods belong to discriminative models. For this reason, the Topic-based methods mainly model the joint distribution between text, topics and words, while the Neural-Network-based method mainly model the conditional distribution between text, topics (latent factors) and words. For example, BTM learns topics over short texts by directly modeling the generation between topics and an unordered word-pair occurrence patterns [15]. Zuo *et al.* propose a novel probabilistic generative model called Pseudo-text-based Topic Model (PTM) by implicitly aggregating short texts as a pseudo text [16]. As a result, by modeling the topic distributions of latent pseudo texts rather than short texts, PTM can be better against data sparsity. Different from the Topic-based models, Neural-Networks-based approaches adopt the low-dimensional hidden layer outputs for short text embedding by training a parametrized neural network for fitting the conditional distribution [18], [30], [31]. For short text embedding, Cedric *et al.* provide a learning procedure based on a novel median-based loss function to weighty aggregate word embedding [19]. Xu *et al.* propose a neural network called DE-CNN for shot text classification, which can incorporate context-relevant concepts into a CNN [20].

B. MANIFLOD LEARNING

Numerous successful manifold learning methods preserving the high-dimensional observations in the low-dimensional embeddings show that manifold property can be a discrete approximation by the nearest neighbor graph of scattered observation points, like Locally Linear Embedding (LLE) [22], Laplacian eigenmaps (LEs) [23] and Isometric Feature Mapping (Isomap) [24]. Some probabilistic versions motivated by the idea of LEs were subsequently proposed to extract text embeddings, such as LapPLSI [32], LTM [33], and DTM [34]. Specifically, LapPLSI, LTM, and DTM develop different manifold graph regularization terms to guide the model fitting of pLSA. The manifold graph regularization generally can be summarized as follows:

$$\sum_{\mathrm{i,j}} W_{ij} Dist\left(y_i, y_j\right) \tag{1}$$

where y is the low-dimensional embeddings, W_{ij} is the edge weight between text i and j in the neighborhood graph, and $Dist()$ indicates a distance measurement of embeddings [34]. LTM employ the Kullback-Leibler (KL) divergence as $Dist()$ to measure the distance of embeddings, whereas DTM and LapPLSA define $Dist()$ as the Euclidean distance. Moreover, DTM goes further to consider negative relationships of texts to improve the full discriminating power [34].

Other promising manifold-inspired approaches exploring text embedding are some AutoEncoder-based variants. The AutoEncoders (AEs) is a one-hidden-layer multi-layer perceptron (MLP), also called AutoAssociators, aiming to reconstruct the original input as correctly as possible [35]. In general, it consists of an encoder f_θ, encoding an input vector $x \in R^d$ to low-dimensional embeddings $y = f_\theta(x) \in \mathrm{R}^k k < d$, and a decoder g_{θ^T}, decoding y back to the input space $\hat{x} = g_{\theta^T}(y) \in \mathrm{R}^d$ as the reconstruction of x, where the mutually transposed parameters θ, θ^T are learned by stochastic gradient descent (SGD) to minimize self-reconstruction error (SRE). After that, the CAE [36], GAE [37] and LEAE [38] have been proposed by incorporating various graph regularization terms, yielding better low-dimensional embeddings of observations concentrated in the vicinity of a smooth manifold. CAE employ the Frobenius norm of the encoder's Jacobian as the regularization terms of the AEs, resulting in the encoder is less sensitive to the input but being sensitive to variations along the high-density manifold [36]. GAE ex-plored the possibility of defining existing manifold learning operators (ISOMAP, LLE and LE) as regularization terms of AEs and propose Deep-GAE to handle highly complex datasets, like image [37]. For text representation, LEAE re-gularize the training of AEs to reconstruct

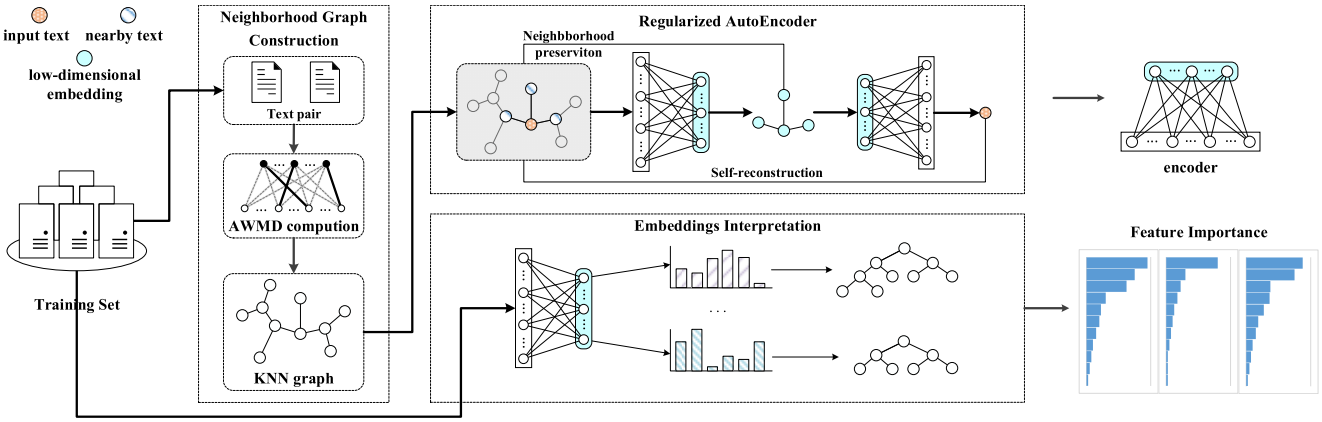

FIGURE 1. Block diagram of STE-AEs, including 3 steps, Neighborhood Graph Construction, Regularized Auto-Encoders and Embedding Interpretation. In this section, the details of the 3 steps will be explained separately.

k nearest nearby texts instead of the input text, yielding an improvement on clustering and classification [38]. However, the aforementioned works neglect the damage of data sparsity of short text to manifold learning, especially when building neighborhood graph, the data sparsity will seriously affect the effect of similarity measurement between short texts.

III. METHODOLOGY

The block diagram of our approach is shown in Figure 1, and the main idea is as follows: motivated by manifold hypothesis [39], we assume that each short text is embedded in a low-dimensional manifold and nearby texts (or neighborhood) have a strong semantics dependency. Following this hypothesis, we proposed a manifold-regularized Short Texts Embedding approach, namely STE-AEs, aiming to regularize the training of AEs to extract the intrinsic discriminative embeddings by preserving the semantic dependency of the neighborhood. Specifically, we first construct the k-nearest-neighborhood (KNN) graph based on AWMD. Then, we define the manifold graph regularization term as the weighted cross-entropy of nearby texts' embeddings using the edge weight of the KNN graph and regularize the training of AEs with a joint minimization of self-reconstruction and manifold graph regularization. Finally, the encoder $y = f_\theta(x) \in \mathrm{R}^k$ can be used as an explicit parametrized embedding mapping function to extract the short text embeddings. Additionally, we employ the Random Forest (RF) to perform regression analysis on the training set and its activation value of the encoder. As a result, the RF model can provide feature importance to select certain understandable words for embed-ding interpretation.

A. NEIGHBORHOOD GRAPH CONSTRUCTION BASED ON AWMD

1) KNN GRAPH CONSTRUCTION

In manifold learning, the neighborhood graph can be treated as a discrete approximation with respect to a smooth manifold [23], and thus the construction of neighborhood graph is usually the basic step of the manifold learning framework. In the literature review, the common construction mainly contains either connecting data points within a radius of ε, called the ε-neighborhood graph, or connecting k-nearest data points, the KNN graph. In practice, the KNN graph is more popular, since the ε-neighborhood graph provides weaker performance [40]. Therefore, in this paper, we employ KNN graph to depict the manifold structure of the entire corpus. Given a training set $\mathbf{X} = \{\boldsymbol{x}_1, \ldots \boldsymbol{x}_m\}$, where $\boldsymbol{x}_i = [x_1, \ldots x_n]^{\mathrm{T}} \in \mathrm{R}^{n\times 1}$ is a n-dimensional text vector of text i, n is the vocabulary size, and x_j is the normalized count value of the j^{th} word in vocabulary. Let $\mathbf{G} = (\mathbf{X}, \mathbf{A})$ denotes a KNN graph, where $\boldsymbol{A} = \left(a_{ij}\right)_{n\times n} \in R^{n\times n}$ is an adjacency matrix composed of similarities between any short text pair $\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)$. Specifically, given a short text $\boldsymbol{x}_i$, if short text $\boldsymbol{x}_j$ is one of the k nearest neighbors, then an edge a_{ij} connects $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, weighted with their pairwise similarities; otherwise, $a_{ij}= 0$. However, this definition leads to a directed graph, since $\boldsymbol{x}_i$ may not be among k nearest neighbors of $\boldsymbol{x}_j$. In STE-AE, we take a undirect definition of KNN, that is, if $\boldsymbol{x}_i$ is among the k nearest neighbors of $\boldsymbol{x}_j$ or if $\boldsymbol{x}_j$ is among the k nearest neighbors of $\boldsymbol{x}_i$, there is an weighted edge connecting $\boldsymbol{x}_j$ and $\boldsymbol{x}_j$. Assuming there are sufficient short texts to ensure that the short text manifold is well-sampled, this definition will endow some vertices in KNN graph more nearby texts, facilitating efficient propagation of the intrinsic discriminative semantics along with KNN graph [41]. Meanwhile, more nearby texts are beneficial to enrich the semantics of single short text against the data-sparse issue of short text. Let $\boldsymbol{N}^{i,t}=\{\boldsymbol{x}_i^1, \ldots \boldsymbol{x}_i^t\}$ denotes a set of nearby texts of $\boldsymbol{x}_i$, where $\boldsymbol{x}_i^k, k \leq t$, is the k^{th} nearest text, and the subset $\{\boldsymbol{x}_i^{k+1}, \ldots \boldsymbol{x}_i^t\}$ consist of some short texts that take $\boldsymbol{x}_i$ as their k nearest neighbors. The construction procedure of KNN graph is summarized as follows.

2) ATTENTION-BASED WEIGHTED MATCHING DISTANCE

The basic idea of STE-AEs is to preserve the neighborhood semantic dependency for short text embedding. In this paper,

Algorithm 1 The Construction Procedure of KNN Graph

Input: k is the nearest neighbor numbers, and training set $X = \{x_1, \ldots x_m\}$.
Output: a KNN graph $\mathbf{G} = (\mathbf{X}, \mathbf{A})$
Let $N^{i,t} = \emptyset$;
For each instance $x_i \in \mathbf{X}$
Compute the pairwise similarity distance $d(x_i, x_j)$ of any short text pair $(x_i, x_j), x_j \in \mathbf{X}$;
Sort in descending order according to the value of $d(x_i, x_j)$.
Based on descending order of $d(x_i, x_j), x_j \in X$, if the order of $d(x_i, x_j)$ is top-k, let $N^{i,t} = N^{i,t} \cup x_j$, and connect short text pair (x_i, x_j) with pairwise similarity
distance $d(x_i, x_j)$;
End for

the neighborhood semantic dependency is depicted as the local geometrical pattern of k nearest neighbors weighted with pairwise similarity distance. Previous works indicate pairwise similarity distance can be defined as Euclidean distance or KL divergence [33], [34], yet, for short text embedding, due to the data sparseness issue and the ignorance of the semantic connection between words from bag-of-words representation, a degradation of pairwise similarity measurement may arise when text pair has many synonyms but do not have the same word. To address this issue, we characterize the inherent semantics of words using word embeddings technology, and further model the semantic relationships between synonyms by developing an attention-based weighted matching distance (AWMD), providing a robust pairwise similarity measurement for KNN graph construction.

Word embeddings is a general term for word vectorized representation technology derived from the distributional hypothesis, allowing words with similar meaning to have a similar representation. There have been some short text embedding works using pre-trained word embeddings technology to alleviate the sparseness of short text data [27], [42], [43]. Some well-known pre-trained word embedding models include word2vec, Glove, ELMo and fastText. Based on the pre-trained word embeddings, the traditional bag-of-words can be transformed as bag-of-word embeddings and each short text can be viewed as an independent embedding set of occurred words. Meanwhile, the semantics similarity of word embeddings pair can be treated as an edge weight. Therefore, a short text pair is equivalent to a bipartite graph, and then the pairwise similarity distance $d(x_i, x_j)$ can be defined as a maximum-weighted matching distance of word embeddings. Furthermore, to better model the semantic connection between synonyms, we employed attention mechanism to assign weight coefficient so as to actively focus on the latent synonyms pattern behind short text pair (x_i, x_j). The attention mechanism can be described as mapping a query and a set of key-value pairs to an output [44]. According to whether the sources of query and key-value pairs are the same, the attention mechanism can be divided into inter-attention and intra-attention (more famous name is self-attention) [45].

Specifically, let $\mathbf{D} = \{w_1, \ldots w_n\}$ denotes the set of word embeddings, where n is the vocabulary size, d is the dimension of each word embedding, $w_i \in R^{d\times 1}$ is embedings of i^{th} word in vocabulary. Given a short text pair (x_i, x_j), let $\mathrm{T}^i = \left(w_s^i\right)_{|x_i|}$ denotes the embedding matrix of occurred word in text x_i, where $|x_i|$ is the length of unique words, similarly, we can have $\mathrm{T}^j = \left(w_t^j\right)_{|x_j|}$. Firstly, we employ self-attention to compute a contextual representation of the occurred word in text x_i, denoted as $C^i = \left(\mathrm{c}_s^i\right)_{|x_i|}$, where the c_s^i is the contextual representation of word w_s^i in text x_i (Figure 2 (a)), its computation is formalized as follows,

$$c_s^i = softmax\left(\frac{w_s^i \cdot T^i}{\sqrt{d}}\right) \cdot T^i \tag{2}$$

In self-attention, the key and value matrixes come from the same source T^i. The essence is the weighted summation of each element, where the weight is computed by an attention score function, like *soft*max() of the query with the corresponding key. Secondly, based on inter-attention, we employ *soft*max() as attention score function to compute the edge weight connecting each contextual representation pair, like $\left(c_s^i, c_t^j\right)$. For a short text pair (x_i, x_j), there are two directions of the weighted edges connecting c_s^i and c_t^j, shown as Figure 2 (b). One picks c_s^i as the query to match every element of C^j, denoted as $A\left(\mathrm{c}_s^i \to \mathrm{C}^j\right)$; the other one picks c_t^j as the query to match every element of C^i, denoted as $A\left(c_t^j \to \mathrm{C}^i\right)$.

$$A\left(c_s^i \to C^j\right) = softmax\left(\frac{c_s^i \cdot C^j}{\sqrt{d}}\right) \tag{3}$$

$$A\left(c_t^j \to C^i\right) = softmax\left(\frac{c_t^j \cdot C^i}{\sqrt{d}}\right) \tag{4}$$

Finally, the edge weight between c_s^i and c_t^j is taken by the maximum of the above two cases, denoted as $E(c_s^i, c_t^j)$. Based on this definition, $E(c_s^i, c_t^j)$ reflects the most degree of similarity between contextual representation pair. After connecting all contextual representation pairs, it produces a weighted bipartite graph of short text pair (x_i, x_j), like Figure 2 (c). Finally, the AWMD can be easily solved by finding the average of maximum weighted bipartite matching based on the famous Hungarian algorithm[2] (Figure 2 (d)). Let $\mathrm{M} = \{\ldots, E(c_s^i, c_t^j), \ldots\}$ denote the set of the connecting edges in final matching, the AWMD is defined as follows,

$$AWMD(x_i, x_j) = \frac{1}{|M|}\sum\nolimits_{\mathrm{M}} Attend\left(c_s^i, c_t^j\right) \tag{5}$$

[2]The running time of the Hungarian algorithm is up to $\mathrm{O}(V^2E)$ using the Bellman–Ford algorithm for shortest path search, or achieve $\mathrm{O}(V^2\log\{V\} + VE)$ with the Dijkstra algorithm and Fibonacci heap.

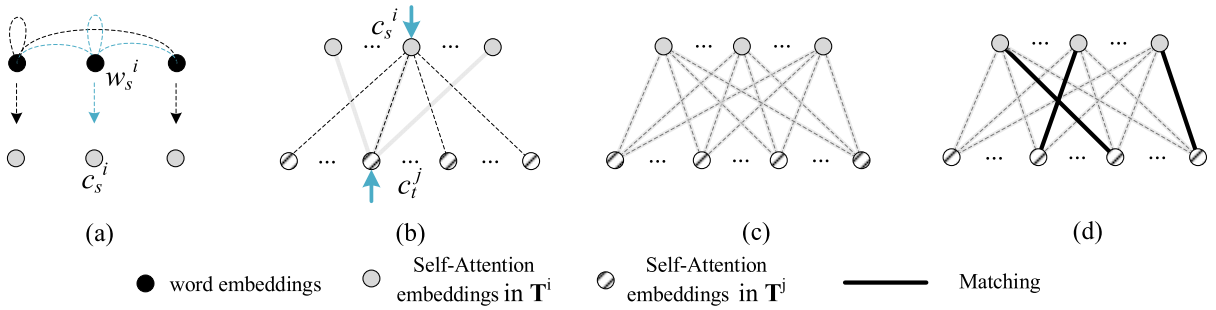

FIGURE 2. (a) contextual representation based on self-attention. (b) is an illustration of two directions of the weighted edges. (c) is the produced weighted bipartite graph of short text pair (x_i, x_j). (d) is an example of maximum-weighted bipartite matching.

where $|\mathcal{M}|$ is the numbers of edges in matching. We can see that the AWMD is the normalization of the matching, which helps to eliminate the unfairness case when the length of short text pair is different. Besides, AWMD is a similarity metric, the larger value of AWMD indicate the more similar of short text pairs.

B. REGULARIZED AUTOENCODERS WITH NEIGHBORHOOD PRESERVATION

Based on the constructed KNN graph, the neighborhood semantic dependency has been depicted as the local geometrical pattern of k nearest neighbors weighted with AWMD. In this section, we regularized the training of the AEs by preserving such local geometrical pattern from the observation space in the low-dimensional embedding space and provide an explicit parametrized mapping function between observations and embeddings. Specifically, we took the AWMD to weight the cross-entropy of low-dimensional embeddings of k-nearest-neighbors as the manifold graph regularization term and regularize the training of AEs using a joint minimization of self-reconstruction and manifold graph regularization.

1) THE OPTIMIZE OBJECTIVE FUNCTION

Formally, given a short text $\boldsymbol{x}$, let $\boldsymbol{y} = [y_1, \ldots y_d]^{\mathrm{T}} \in R^{d\times 1}$ denotes the low-dimensional embeddings, where d is the dimension of the low-dimensional embeddings, let $\hat{x} = \left[\hat{x}_1, \ldots \hat{x}_n\right]^{\mathrm{T}} \in R^{n\times 1}$ denotes the self-reconstruction of $\boldsymbol{x}$. The AEs consists of two modules: the encoder and the decoder. The encoder transforms an input vector $\boldsymbol{x}$ into the low-dimensional embeddings $\boldsymbol{y}$, whose mathematical expression is a nonlinear version of the affine map,

$$\boldsymbol{y} = f_{\boldsymbol{W},\boldsymbol{b}}(\boldsymbol{x}) = \sigma(\boldsymbol{W}\cdot\boldsymbol{x} + \boldsymbol{b}) \tag{6}$$

The decoder transforms the embeddings $\boldsymbol{y}$ back to a self-reconstruction $\hat{\boldsymbol{x}}$ whose form is same to the encoder.

$$\hat{\boldsymbol{x}} = \boldsymbol{f}_{\boldsymbol{W}^T,\boldsymbol{c}}(\boldsymbol{y}) = \boldsymbol{\sigma}\left(\boldsymbol{W}^{\boldsymbol{T}}\cdot\boldsymbol{y} + \boldsymbol{c}\right) \tag{7}$$

where $\sigma(\cdot)$ is sigmoid function $\sigma(a) = \lceil 1 + exp\{a\}\rceil^{-1}$, $\boldsymbol{b}\in R^{d\times \mathbf{1}}$ is bias vector of the encoder, and $\boldsymbol{W} \in R^{d\times n}$ is the encoder parameters. $\boldsymbol{c}$ is bias vector of the decoder, and the parameters of decoder $\boldsymbol{W}^{\mathrm{T}}\in R^{n\times d}$ is "tied weight" with $\boldsymbol{W}$, which reduce the scale of estimated parameters and can make it harder for the encoder to stay in the linear regime of its nonlinearity without paying a high price in reconstruction error [45]. Finally, the self-reconstruction error between $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ is measured with the cross-entropy, denote as $H_B(\boldsymbol{x}, \hat{\boldsymbol{x}})$.

$$\boldsymbol{H_B}(\boldsymbol{x},\hat{\boldsymbol{x}}) = -\sum\nolimits_{\mathbf{1}}^{\boldsymbol{n}}\left[\hat{\boldsymbol{x}}_{\boldsymbol{j}}\boldsymbol{log}\hat{\boldsymbol{x}}_{\boldsymbol{j}} + (\mathbf{1}-\boldsymbol{x}_{\boldsymbol{j}})\,\boldsymbol{log}\,(\mathbf{1}-\hat{\boldsymbol{x}}_{\boldsymbol{j}})\right] \tag{8}$$

The manifold graph regularization for a given short text $\boldsymbol{x}_i$ is measured with the weighted cross-entropy, denoted as $\mathcal{M}_{\boldsymbol{x}_i}(\boldsymbol{N}^{i,t})$.

$$\mathcal{M}_{\boldsymbol{x}_i}\left(\boldsymbol{N}^{i,t}\right) = \sum\nolimits_{j\in\boldsymbol{N}^{i,t}} AWMD\left(\boldsymbol{y}_i,\boldsymbol{y}_j\right)\cdot H_B\left(\boldsymbol{y}_i,\boldsymbol{y}_j\right) \tag{9}$$

Taken together, the object function for a given short text $\boldsymbol{x}_i$ is defined as follows,

$$\mathcal{J}_{\boldsymbol{x}_i}(\boldsymbol{W},\boldsymbol{b},\boldsymbol{c}) = H_B\left(\boldsymbol{x}_i,\hat{\boldsymbol{x}}_i\right) + \lambda\mathcal{M}_{\boldsymbol{x}_i}\left(\boldsymbol{N}^{i,t}\right). \tag{10}$$

Furthermore, we impose sparse constraint on low-dimensional embeddings, which is mainly based on two main reasons: one is to facilitate understanding of the low-dimensional embedding, and the other is to prevent AEs from learning an identity transformation. The sparse constraint aims to regularize AEs to reduce the difference between the average of each embedding $\bar{\boldsymbol{\rho}} = [\bar{\rho}_1, \ldots \bar{\rho}_d] \in R^{1\times d}$ and a fixed sparsity target $\boldsymbol{\rho}$[3]$= [\rho, \ldots \rho]\in R^{1\times d}$ by minimizing the Kullback-Leibler (KL) divergence, where $\bar{\rho}_j = 1/m\sum_{\boldsymbol{X}} y_j$ denotes the average output of low-dimensional embeddings over the training set $\mathbf{X} = \{\boldsymbol{x}_1, \ldots \boldsymbol{x}_m\}$. Then, the corresponding sparsity regularization term is defined as follows,

$$KL(\boldsymbol{\rho}\,\|\bar{\boldsymbol{\rho}}) = \sum\nolimits_{j=1}^{d}\rho log\frac{\rho}{\bar{\rho}_j} + (1-\rho)\,log\frac{1-\rho}{1-\bar{\rho}_j}. \tag{11}$$

Therefore, the final object function of STE-AEs over the training set $\mathbf{X}$ is summarized as follows,

$$\mathcal{J}(\boldsymbol{W},\boldsymbol{b},\boldsymbol{c}) = \frac{1}{m}\sum\nolimits_{\boldsymbol{x}_i\in\mathbf{X}}\mathcal{J}_{\boldsymbol{x}_i}(\boldsymbol{W},\boldsymbol{b},\boldsymbol{c}) + \gamma KL(\boldsymbol{\rho}\,\|\bar{\boldsymbol{\rho}}). \tag{12}$$

[3]In practice, the sparse target is a settable constant hyperparameter. Here, for the convenience of formula 12, we express it as a constant vector.

2) MODEL OPTIMIZATION

Now, STE-AEs can provide the encoder as an explicit parameterized mapping function between observations and embeddings by the minimization of $\mathcal{J}(\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{c})$. For this purpose, we employ gradient descent (GD) algorithm to optimize the parameters $\boldsymbol{W}$, $\boldsymbol{b}$, and $\boldsymbol{c}$. For convenience, some variable symbols of STE-AEs for the description of model optimization are shown as follows.

symbols	descriptions
n_x	size of inputs and outputs
n_y	size of hidden units
$x_j^{(i)}, j \in \{1,2,\ldots n_x\}$	j^{th} value input of *encoder* for $\boldsymbol{x}_i$
$\hat{x}_j^{(i)}, j \in \{1,2,\ldots n_x\}$	j^{th} value output of *decoder* for $\boldsymbol{x}_i$
$y_j^{(i)}, j \in \{1,2,\ldots n_y\}$	j^{th} value of hidden unit (embeddings) or output of *decoder* or input of *encoder*
$\boldsymbol{N}^{i,t} = \{\boldsymbol{x}_i^1, \ldots \boldsymbol{x}_i^t\}$	a set of nearby texts of $\boldsymbol{x}_i$, $\left\|\boldsymbol{N}^{i,t}\right\|$is length of nearby texts
a_{ik}	the AWMD between $\boldsymbol{x}_i$ and $\boldsymbol{x}_k$
$y_j^{(i,k)}$	j^{th} value hidden unit of $\boldsymbol{x}_k \in \boldsymbol{N}^{i,t}$
ρ	sparsity hyper-parameter, which denotes a fixed sparsity target
$\hat{\rho}_j$	the average value of y_j over the training set
W_{ij}	weight connecting i^{th} hidden unit to j^{th} input
	weight connecting i^{th} hidden unit to j^{th} output
b_i	bias of i^{th} hidden layer
c_j	bias of j^{th} output layer
λ, γ	non-negative regularization hyper-parameter

In detail, based on the gradient descent algorithm, the parameters $\boldsymbol{W}$, $\boldsymbol{b}$, and $\boldsymbol{c}$ are updated as follows,

$$\boldsymbol{W} = \boldsymbol{W}_{old} - \eta \left[\frac{1}{m} \sum_{i=1}^{m} \nabla_W \mathcal{J}(\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{c}) \right] \tag{13}$$

$$\boldsymbol{b} = \boldsymbol{b}_{old} - \eta \left[\frac{1}{m} \sum_{i=1}^{m} \nabla_b \mathcal{J}(\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{c}) \right] \tag{14}$$

$$\boldsymbol{c} = \boldsymbol{c}_{old} - \eta \left[\frac{1}{m} \sum_{i=1}^{m} \nabla_c \mathcal{J}(\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{c}) \right] \tag{15}$$

where η is learning rate, and $\nabla(\cdot)$ is partial derivatives of corresponding parameters. Therefore, the key step of model optimization is the computation of partial derivatives with respect to parameters. According to expression (6), (7) and (8) and Chain rule, we have

$$\begin{aligned} \nabla_{W_{ij}} \mathcal{J}(W; \boldsymbol{x}_i) &\overset{def}{=} \frac{\partial \mathcal{J}(W, b, c)}{\partial W_{ij}} \\ &= \left(\hat{x}_j^{(i)} - x_j^{(i)}\right) y_i^{(i)} + x_j^{(i)} y_i^{(i)} \left(1 - y_i^{(i)}\right) \\ &\quad \times \sum_k \left(\hat{x}_k^{(i)} - x_k^{(i)}\right) W_{jk} + \frac{\lambda}{\left|\boldsymbol{N}^{i,t}\right|} \\ &\quad \times \sum_{k \in \boldsymbol{N}^{i,t}} a_{ik} x_j^{(i)} \left(y_i^{(k)} - y_i^{(i)}\right) \\ &\quad + \gamma \left(\frac{1-\rho}{1-\hat{\rho}_i} - \frac{\rho}{\hat{\rho}_i}\right) x_j^{(i)} y_i^{(i)} \left(1 - y_i^{(i)}\right) \end{aligned} \tag{16}$$

$$\begin{aligned} \nabla_{b_i} \mathcal{J}(b; \boldsymbol{x}_i) &\overset{def}{=} \frac{\partial \mathcal{J}(W, b, c)}{\partial b_i} \\ &= y_i^{(i)} \left(1 - y_i^{(i)}\right) \sum_k \left(\hat{x}_k^{(i)} - x_k^{(i)}\right) W_{jk} \\ &\quad + \frac{\lambda}{\left|\boldsymbol{N}^{i,t}\right|} \sum_{k \in \boldsymbol{N}^{i,t}} a_{ik} \left(y_i^{(k)} - y_i^{(i)}\right) \\ &\quad + \gamma \left(\frac{1-\rho}{1-\hat{\rho}_i} - \frac{\rho}{\hat{\rho}_i}\right) y_i^{(i)} \left(1 - y_i^{(i)}\right) \end{aligned} \tag{17}$$

$$\nabla_{c_j} \mathcal{J}(c; \boldsymbol{x}_i) \overset{def}{=} \frac{\partial \mathcal{J}(W, b, c)}{\partial c_j} = \left(\hat{x}_j^{(i)} - x_j^{(i)}\right) \tag{18}$$

The procedure of the model optimization algorithm is shown in Algorithm 2.

Algorithm 2 Model Optimization for STE-AEs

Input: The training set $\mathbf{X} = \{\boldsymbol{x}_1, \ldots \boldsymbol{x}_m\}$.
Output: the parameter of affine mapping, $\boldsymbol{W}$, $\boldsymbol{b}$, and $\boldsymbol{c}$
Construct a knn-graph based on the AWMD over the entire training set, $\mathbf{G} = (\mathbf{X}, \mathbf{A})$.
Randomly initialized $\boldsymbol{W}$, $\boldsymbol{b}$, $\boldsymbol{c}$ of STE-AEs
For *epoch* $= 1$ to *max_epoch*
  Perform a feedforward pass with each instance $\boldsymbol{x}_i$ and its nearby set $\boldsymbol{N}^{i,t}$, computing the activations for the hidden layer, output layer and $\hat{\rho}_j$;
  Based on expressions (16), (17) and (18), compute the average partial derivatives over the training set $\Delta_{\boldsymbol{W}} = \frac{1}{m} \sum_{i=1}^{m} \nabla_W \mathcal{J}(\boldsymbol{W}; \boldsymbol{x}_i)$, $\Delta_{\boldsymbol{b}} = \frac{1}{m} \sum_{i=1}^{m} \nabla_b \mathcal{J}(\boldsymbol{b}; \boldsymbol{x}_i)$, $\Delta_{\boldsymbol{c}} = \frac{1}{m} \sum_{i=1}^{m} \nabla_c \mathcal{J}(\boldsymbol{c}; \boldsymbol{x}_i)$
  Update parameter: $\boldsymbol{W} = \boldsymbol{W}_{old} - \eta \Delta_{\boldsymbol{W}}$, $\boldsymbol{b} = \boldsymbol{b}_{old} - \eta \Delta_{\boldsymbol{b}}$, $\boldsymbol{c} = \boldsymbol{c}_{old} - \eta \Delta_{\boldsymbol{c}}$
End for

C. EMBEDDINGS INTERPRETATION BASED ON RANDOM FOREST

Now, the low-dimensional embeddings of out-of-sample short text can be extracted easily via the encoder $\boldsymbol{y} = f_{\boldsymbol{W},\boldsymbol{b}}(\boldsymbol{x})$. However, what meaning is implied by each dimension of the low-dimensional embeddings is confusing. In this section, we employ a regression model of Random Forest for feature selection to deal with this issue and try to find certain understandable keywords to improve the interpretability of such embedded variables. The process is divided into two steps: we first propose a partition strategy involving inter-pretation subset based on a ranking of the activation of each dimension of the low-dimensional embeddings, and then execute feature selection via random forest algorithm with each interpretation subset independently.

1) INTERPRETATION SUBSET PARTITION

Given an low-dimensional embeddings matrix $\mathbf{Y} = \{\boldsymbol{y}_1, \ldots \boldsymbol{y}_m\}$, where $\boldsymbol{y}_i = [y_1, \ldots y_d]^{\mathrm{T}} \in R^{d \times 1}$ consists of the activation of the hidden unit, and y_j indicate the activation of j^{th} hidden unit. As is well-known, the activation is calculated via logistic sigmoid function that depends heavily upon the dot product of the input signals and its parameters (synaptic weights and bias). Specifically, the hidden neurons are more active when their inputs are more relevant with their

synaptic weights, while the bias controls the threshold of this correlation [47]. Therefore, the parameters of the neuron determine what kind of input signal is easier to induce the neuron to give a more active activation value and the more active activation demonstrates the input signals contain richer information for embeddings interpretation. According to the sorting of activation to each hidden unit, we can choose a small part of the most active or most informative short texts from the whole training set as an interpretation subset.

There are two partition strategies to select interpretation subsets based on the sorting of activation. One is to set a threshold to select those samples whose activation value is greater than the threshold. Due to the adoption of sparsity regularization, some dimensions in low-dimensional embeddings will be infinitely close to zero. Therefore, it is difficult to set appropriate thresholds based on the absolute size of the activation value to select samples. The other is to select the top-k samples at the front of the sort, which can provide relatively stable most informative short texts. For this reason, we selected the top-k short texts based on the sorting of activation to each the hidden neuron independently and obtain interpretation subsets.

2) FEATURE SELECTION WITH INTERPRETATION SUBSET

In this paper, let $\Omega_{is} = \{\Phi_i\}_d$ denote a collection of the obtained d interpretation subsets, where d indicates the numbers of neurons in hidden layer (or the dimension of the low-dimensional embeddings) and Φ_i indicates a collection of the picked top-k short texts based on the sorting of activation to one hidden neurons. Based on the interpretation subsets, we execute feature selection for each dimension via random forest algorithm respectively. The Random Forest (RF) is an exemplar of the ensemble learning method for classification, which combines the random subspace method and bagging method. The principle of RF is to build a multitude of decision trees using several bootstrap samples from the entire training set and choosing the best split feature from a random selected subset of explanatory variables. In addition, it can provide a ranking of Variables Importance (VI) based on the out-of-bag (OOB) samples using wrapper methods of feature selection. The quantification of the VI is a crucial measure in the feature selection task since it indicates the contribution of candidate variables to the response variable for interpretation purposes. Therefore, we built RF model using each interpretation subset independently, then determine a subset of features according to the ranking of VI.

IV. EXPERIMENT

Here, we investigated the performance of the extracted low-dimensional embeddings from two aspect: discriminability and interpretability. Firstly, we provided different dimensionalities of embeddings (10, 30, 50, 70, 90, 110, 130) over three widely used text corpora (Web-snippets, 20 newsgroups and Twitter) and compared the performance of STE-AEs with the following state-of-the-art approaches in two widespread applications, clustering and classification.

- Discriminative topic model (DTM, 2012) [34].
- Biterm topic model (BTM, 2014) [15].
- Pseudo-document-based Topic Model (PTM, 2016) [16].
- Locally Embedding AutoEncoders (LEAEs, 2016) [38].
- Self-Taught Convolutional neural network (STC, 2017) [31].
- Self-Train AutoEncoders (ST-AEs, 2019) [48].
- Topic model based on Regularized Non-negative Matrix Factorization Topic (TRNMF, 2020) [42].

Secondly, we provided understandable keywords selected from each interpretation subset by the RF model to interpret what meaning is implied by each dimension of the embeddings.

A. DATASETS

We choose Web-snippets,[4] 20-Newsgroups[5] and Twitter[6] as our experimental datasets. The Web-snippets is a collection of snippets of texts presented as results of a query by a search engine, which consists of 12,340 search snippets belonging to 8 domains [49]. 20-Newsgroup is a collection of newsgroups, including across 20 different newsgroups. For short text embedding, we selected only the samples with less than 21 words, denoted as 20Nshort, as done in [43]. The Twitter consists of 5,513 hand-classified tweets divided into 4 different topics: Apple, Google, Microsoft, Twitter [42]. All datasets were preprocessed by making all the text lower case, removing non-alphabetic characters, and stopwords in a standard list. Besides, some words shorter than 3 characters or appearing under 10 times in Snippet or under 5 times in the other two datasets were removed. Table 1 shows some details of statistical information about three datasets, where D is the number of short texts, V is the size of vocabulary spanned over each domain, $\bar{D}$ and $St.Dev$ are the average mean and the standard deviation of the number of words occurred in each text.

TABLE 1. Statistical information of the three datasets.

datasets	domains	D	V	$\bar{D}$	$St.Dev$
Web-snippets	8	12,340	5,913	16.68	±2.61
20Nshort	20	1,794	6,023	7.51	±3.93
Twitter	4	2,520	1,390	5.59	±2.14

B. EXPERIMENTAL PROCEDURE

To obtain a fair experimental performance, we conducted 5-fold cross-validation (5-CV) over three datasets. There are 3 general procedures in each iteration: embedding model construction, embeddings extraction and application performance evaluation (clustering and classification). Specifically, we shuffled three datasets and divided each

[4] http://jwebpro.sourceforge.net/data-web-snippets.tar.gz
[5] http://web.ist.utl.pt/acardoso/datasets/
[6] https://github.com/zfz/twitter_corpus

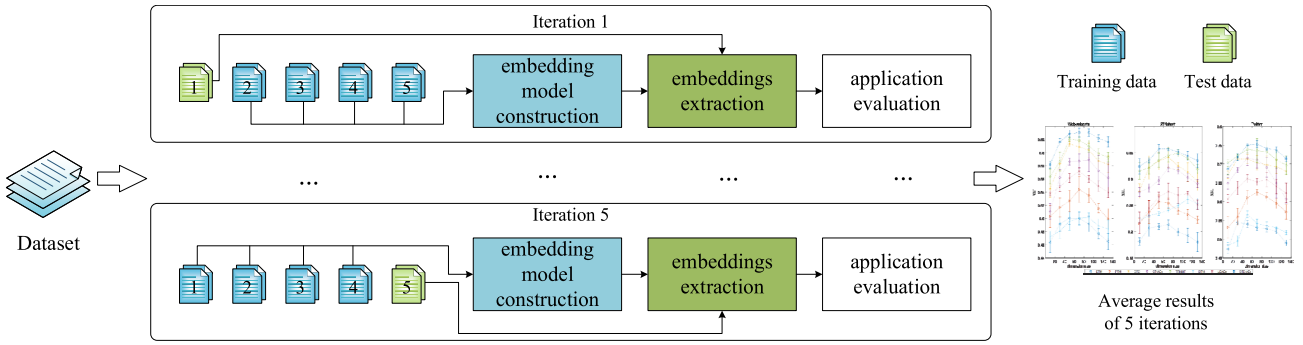

FIGURE 3. Flow diagram of the experimental procedure.

dataset into five equal subsets. Then, one of these subsets was cyclically picked for embeddings extraction and application performance evaluation (test set), and the remaining subsets were used as embedding model construction (training set) until all subsets were picked for evaluation. Therefore, we conducted five iterations estimation and obtained five application performance. Finally, the estimation results are the average performance of the five iterations. Figure 3 is the flow diagram representting this experimental procedure.

All comparison methods were performed under the uniform hyper-parameters setting over three datasets. For BTM,[7] we set $\alpha = 50/topic\ numbers$ and $\beta = 1e-2$. For PTM, we used the optimal setting as follows: $\alpha = 0.1$, $\lambda = 0.1$ and $\beta = 0.01$ [16]. For STC, the CNN have two convolutional layers, where the number of feature maps at the first convolutional layer is 12, and 8 feature maps at the second convolutional layer. The value of k-max pooling is 5. For TRNMF, we set $\alpha = 0.1$, $\lambda = 0.1$, $\beta = 0.1$ and $\gamma = 0.01$, the Gibbs sampling is run for 1,000 iterations [42]. For ST-AEs, we used two versions of pre-trained word2vec embeddings: one is for Web-snippets and 20Nshort[8], the other is for Twitter.[9] We fixed $\alpha = 0.1$ value for all corpora and set the batch size to 64 and pre-trained the autoencoder for 15 epochs [48]. Please note that we removed the words that were not in the word embedding lookup table. Although this may cause the semantic loss of the short text and increase the sparsity, it will not change the fairness of the comparison experiment.

For STE-AEs, we used the same setting of pre-trained word2vec embedding as ST-AEs, and the optimal hyper-parameters obtained after 5-CV, the learning rate $= 0.5$,$epoch = 200$, $\lambda = 50$, $\gamma = 10$, the fixed sparsity target $\rho = 1/d$ and the number of neighbors $K = 13$,. For LEAEs, the batch size is set to 100 and the neighbors is set to 7, $\eta = 1.2$, $epoch = 30$, $\lambda = 100$. For DTM,[10] we set the number of neighbors is 20 and $\lambda = 1000$. Since the graph regularize-tion of the DTM is based on LE algorithms, DTM cannot provide a specific mapping function from the manifold to the output embedding [41], leading to a limitation on handling a previously unseen short text. To address this issue, we employed inclusive approaches that rebuild similarity and dissimilarity matrices with the evaluation subset, retraining the model based on these matrices [34].

[7] https://github.com/xiaohuiyan/BTM
[8] https://github.com/jacoxu/STC2
[9] https://nlp.stanford.edu/projects/glove/
[10] http://www.cs.cmu.edu/~seungilh/dtm_codes/index.html

C. EXPERIMENTAL RESULTS

1) DISCRIMINATIVE PERFORMANCE IN UNSUPERVISED SETTING

To evaluate the performance of discriminability behind various low-dimensional embeddings of test short texts, we utilized the K-means algorithm to group them with the same cluster number as the ground truth. As is well-known, K-means automatically groups instances according to their measure of similarity of representation vector, thus the clustering results can demonstrate the quality of similarity and dissimilarity of representation. We evaluated the clustering results for 5 iterations via two common estimation metrics: accuracy (ACC) and the normalized mutual information metric (NMI). Given a short text $\boldsymbol{x}_i$, let C_i be the assigned cluster id and S_i be the original label. The ACC is defined as follows [33]:

$$ACC = \frac{\sum_{i=1}^{N} \delta\left(S_i, map\left(C_i\right)\right)}{N} \tag{19}$$

where N indicates the size of the test texts and $map\left(C_i\right)$ matches C_i to equivalent short text labels. The determination of optimal mapping can refer to the Kuhn-Munkres algorithm [50]. $\delta\left(x, y\right)$ is delta function defined as follows:

$$\delta\left(x, y\right) = \begin{cases} 1, & if\ x = y \\ 0, & otherwise \end{cases} \tag{20}$$

The NMI is defined as follows,

$$NMI\left(C, S\right) = \frac{MI\left(C, S\right)}{\sqrt{H\left(C\right), H\left(S\right)}} \tag{21}$$

where $H\left(\cdot\right)$ is the entropy, $\sqrt{H\left(C\right), H\left(S\right)}$ is used for normalizing the mutual information to be in the range of $[0, 1]$. $MI\left(C, S\right)$ denotes the mutual information between C and S, which is measured as,

$$MI\left(C, T\right) = \sum\nolimits_{C_i \in C, T_i \in T} p\left(C_i, T_i\right) log \frac{p\left(C_i, T_i\right)}{p\left(C_i\right) p\left(T_i\right)} \tag{22}$$

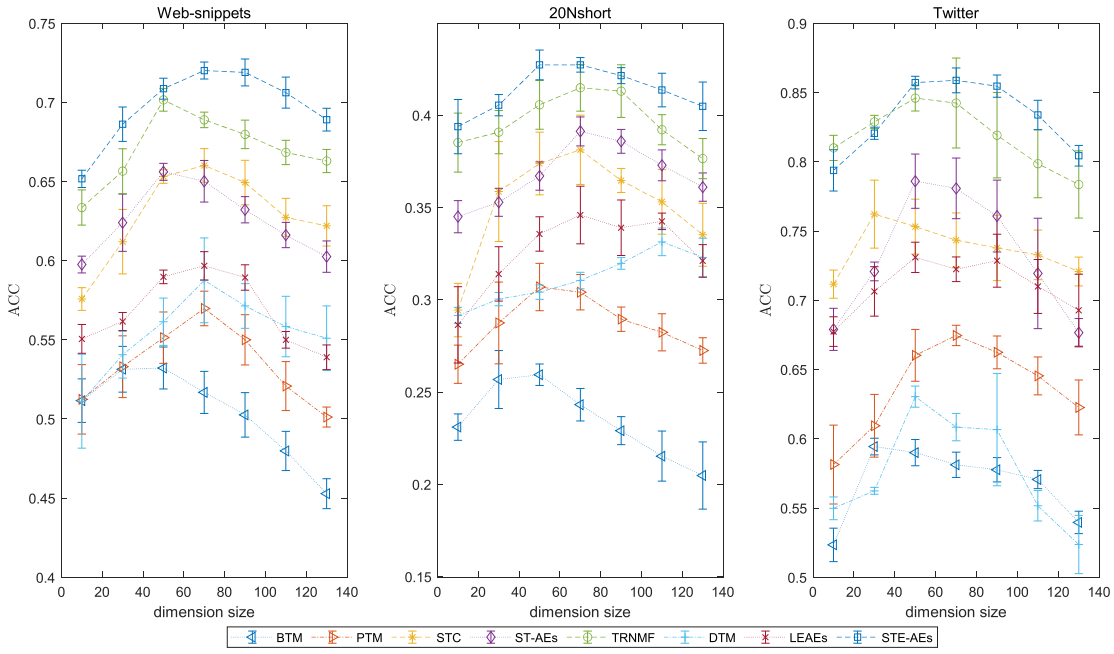

FIGURE 4. The *ACC* of various low-dimensional embeddings in unsupervised settings after 5-CV. The figure from left to right indicates the average performance of embeddings on Web-snippets, 20Nshot, and Twitter, where each point consisting of a mean value as well as standard deviations.

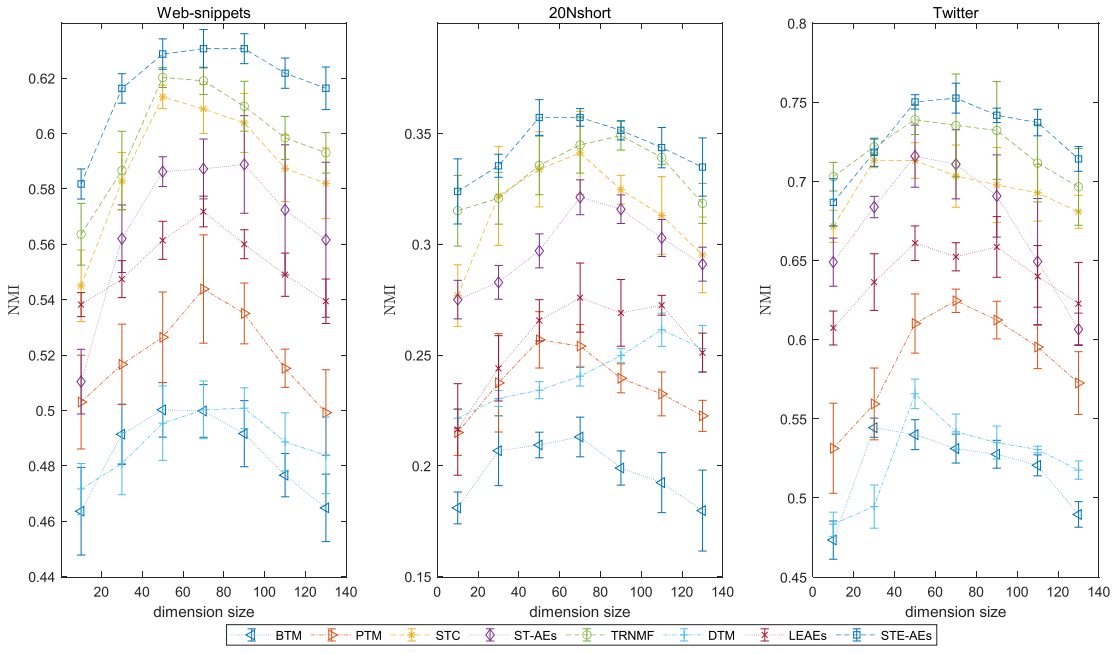

FIGURE 5. The *NMI* of various low-dimensional embeddings in unsupervised settings after 5-CV. The figure from left to right indicates the average performance of embeddings on Web-snippets, 20Nshot, and Twitter, where each point consisting of a mean value as well as standard deviations.

$p(C_i, T_i)$ is the joint probability that $\boldsymbol{x}_i$ belongs to C_i and T_i simultaneously. $p(C_i)$ and $p(T_i)$ denote the probabilities that $\boldsymbol{x}_i$ belongs to C_i and T_i, respectively.

Figure 4 and 5 are the average clustering performance of various low-dimensional embeddings after 5-CV. As shown in the Figure 4 and 5, the mean value (*ACC* and *NMI*) of STE-AEs can consistently outperform the state-of-the-art comparative approaches (TRNMF) over three datasets. In particular, for the Twitter dataset, STE-AEs can achieve the best *ACC* (0.8588 ± 0.0089) in dimension 70. In addition, for three datasets with different sparse scales (more statistical information sees Table 1), the short text embeddings extracted

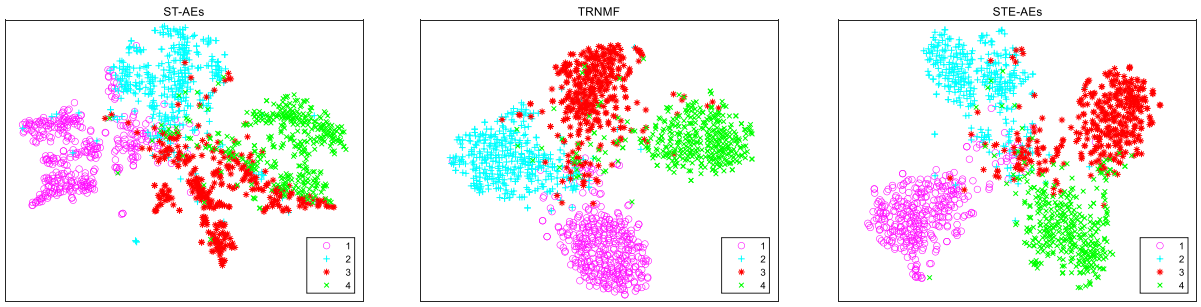

FIGURE 6. The 2D scatter plot 50-dimensional embeddings on Twitter (best performance for visualization).

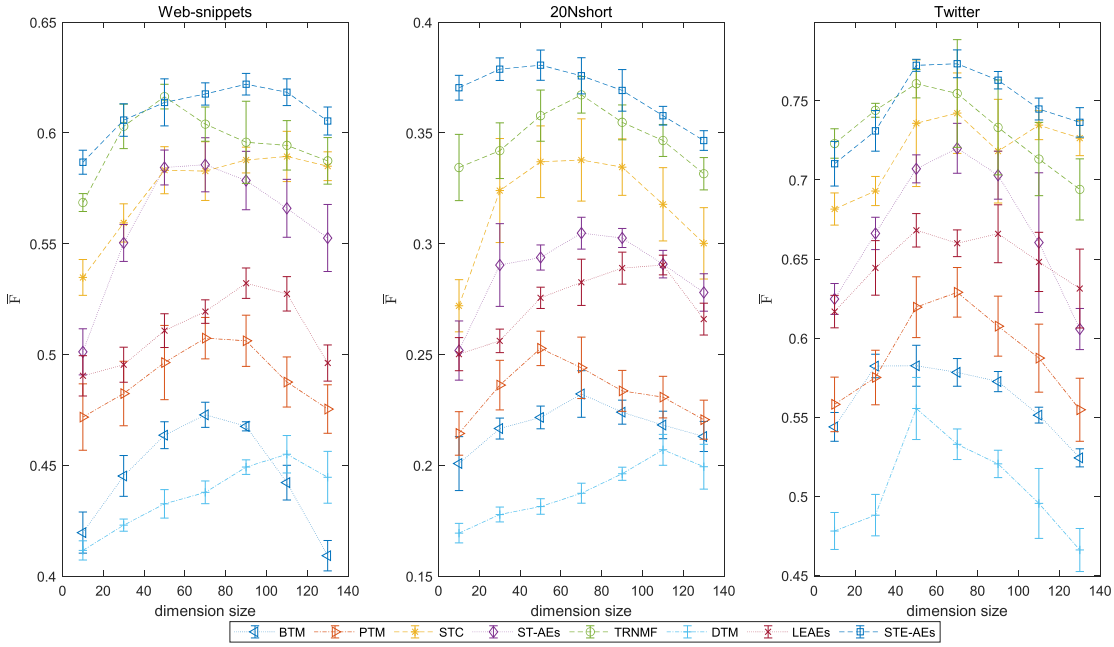

FIGURE 7. Average classification performance of several models with 1-NN for three datasets, where each point consisting of a mean value as well as standard deviations.

by STE-AEs provided smaller standard deviations in various dimensions, which indicates that SE-AEs can achieve more steadier clustering performance. The evidence demonstrates that the preservation of the semantic dependency from the attention-based neighborhood has a positive effect on capturing the intrinsic discriminative explanatory factors. Besides, compared with other manifold-inspired approaches, like DTM and LEAEs, STE-AEs presented a smoother peak, which means that STE-AEs can provide the best clustering performance in a wider range of dimensions. We attribute this evidence mainly to the introduction of sparsity constraints into the extraction process of low-dimensional embeddings, because the fixed sparsity target $\rho = 1/d$ will decrease as the dimension d increases, so STE-AEs still capture the intrinsic discriminative explanatory factors by guiding more hidden unit's activations close to zero, even if the dimension of embeddings is large.

Furthermore, to analyze discriminative performance direct-ly, we adopted a popular tool of data visualization techniques, t-Distributed Stochastic Neighbor Embedding (t-SNE)[11] [51], to visualize low-dimensional embeddings in a 2D scatter plot. The t-SNE utilize a probabilities distribution to measure similarities instead of pairwise distances of objects and minimizes the Kullback-Leibler (KL) divergence between such probabilities in the input and output space, which is conducive to reflect the similarities and dissimilarities over objects. To provide the best visualization, we picked three best performing approaches, STE-AEs, TRNMF and ST-AEs in 50-dimension on Twitter. Figure 7 present scatter diagrams of the 50-dimensional embeddings over test sets. Each dot indicates a short text and each color-shape pair denotes a class. From Figure 7, we can see that STE-AEs and TRNMF present clearer clusters structure than ST-AEs, and comparing with TRNMF, STE-AEs presents clear-cut margins among differ-ent semantic category, which demonstrate that our proposed approach not only preserves inner-class

[11] http://lvdmaaten.github.io/tsne/code/tSNE_matlab.zip

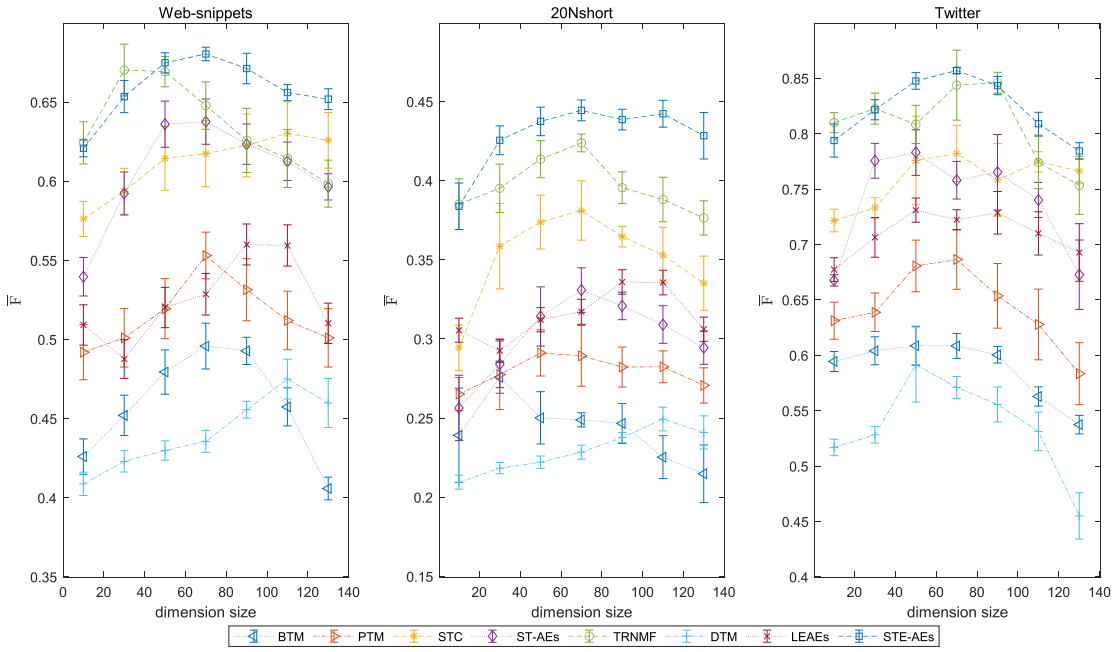

FIGURE 8. Average classification performance of several models with the SVM for three datasets, where each point consisting of a mean value as well as standard deviations.

intrinsic structure but also reduces possible overlap and widens inter-class margins. Intuitively, the evidence shows that our approach provides more separable low-dimensional embeddings than the other methods, confirming the discriminability of our low-dimensional embeddings.

Based on the above evidence, with three measures of *ACC*, *NMI* and t-SNE under three datasets, we can conclude that the proposed approach is effective approach to capture the intrinsic discriminative explanatory factors, improving the performance of short text clustering.

2) DISCRIMINATIVE PERFORMANCE IN SUPERVISED SETTING

In this section, we further compared the influence of discriminative power of various low-dimensional embeddings in a super-vised way. After extracting the low-dimensional embeddings of evaluation subsets, we randomly divided them into 2 equal parts. One is applied for classification test, and the other is used for classifier training of 1-nearest neighbor (1-NN) and support vector machine (SVM),[12] respectively. Since the size of each category is different, we employed the weighted F-measure $\bar{F}$ to estimate the accuracy of the classification model, which is calculated as follows:

$$\bar{F} = \frac{\sum_i c_i F_i}{C} \tag{23}$$

where c_i is the proportion of instances in test set categories i and C is the size of the test set. F_i is the F-measure of categories i, which indicate tradeoff between the precision P_i and recall R_i. The P_i, R_i and F_i are defined as follows:

$$P_i = \frac{|relevant\ texts| \cap |retrieved\ texts|}{|retrieved\ texts|} \tag{24}$$

$$R_i = \frac{|relevant\ texts| \cap |retrieved\ texts|}{|retrieved\ texts|} \tag{25}$$

$$F_i = 2 \cdot \frac{P_i \cdot R_i}{P_i + R_i} \tag{26}$$

Figure 7 and 8 are the average $\bar{F}$ and standard deviations after 5 iterations on 1-NN and the SVM, respectively. From these figures, we can observe a significant improvement and a smoother peak similar to the clustering experiment, which further illustrates the effectiveness of STE-AEs to capture the intrinsic discriminative explanatory factors. In addition, from figure 7 and 8, we see that BTM outperforms DTM in most dimensions, which is different from the performance of *ACC* and *NMI* in Figure 5 and 6. This is mainly because DTM implicitly utilized the valuable class label information to build the similarity matrix [34], and the class label information can give DTM an inherent advantage to improve the discrimenability of low-dimensional embeddings in clustering (unsupervised setting), while for classification (supervised setting), the classifier can naturally utilize the class label information, so DTM's advantage of implicit use of category information may be reduced or even surpassed by other methods, like BTM.

In summary, we conclude that STE-AEs can capture the intrinsic discriminative explanatory factors, improving the performance of short text clustering and classification. The excellent discriminability benefits from the good

[12]We implemented the classification framework via weka. In this paper, we used lazy.IB1 for 1-NN, but for SVM, we adopt LIBSVM java code from github (https://github.com/cjlin1/libsvm), which could be easily executed by weka.

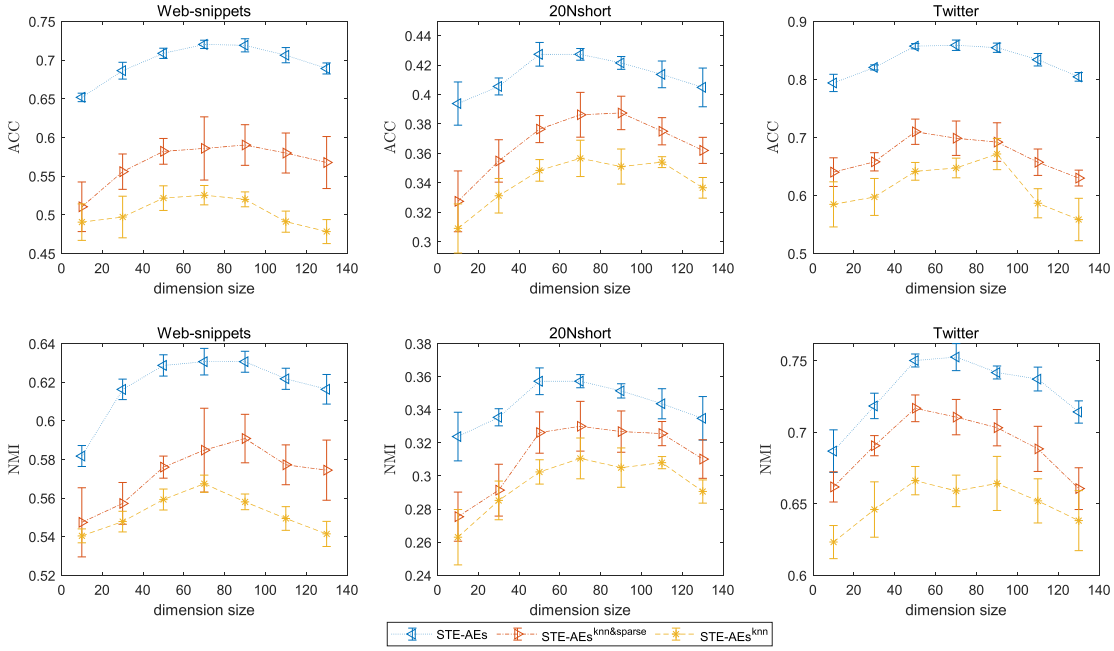

FIGURE 9. Average clustering performance of *ACC* (top) and *NMI* (bottom) for three datasets, where each point consisting of a mean value as well as standard deviations.

expression of low-dimensional embeddings for the inherent similarity of short text. This is mainly because we take a local perspective that the embeddings of each short text are strongly associated with the specific word co-occurrence patterns of itself and its neighbors, ensuring the embeddings are local invariant around the neighborhoods and improve the discriminative effect. Specifically, the minimization of manifold graph regularization will guide the cross-entropy of low-dimensional embeddings to be small, when the AWMD of short text pair is large. In other words, the encoder function tends to assign similar low-dimensional embeddings to nearby short texts or the neighborhood of semantic dependency.

3) ABLATION EXPERIMENT

Additionally, following the above unsupervised and supervised settings, we provided an ablation experiment to evaluate the effect of the AWMD-based manifold graph regularization and the sparsity regularization on the final performance. Specifically, STE-AEs indicates the complete solution incorporating the AWMD-based manifold graph regularization and the sparsity regularization. Comparing with STE-AEs, $\text{STE}-\text{AEs}^{knn\&sparse}$ also incorporates the sparsity regularization, but modify the manifold graph regularization by taking the Euclidean distance as the pairwise similarity distance for KNN construction. Besides, different from $\text{STE}-\text{AEs}^{knn\&sparse}$, $\text{STE}-\text{AEs}^{knn}$ further removes the sparsity regularization from the optimize object function. Figure 9 and 10 are the average clustering and classification performance over three datasets, respectively.

From the comparison between STE-AEs and $\text{STE}-\text{AEs}^{knn\&sparse}$ in Figure 9 and 10, we can see that STE-AEs can consistently outperform $\text{STE}-\text{AEs}^{knn\&sparse}$ in clustering and classification, which demonstrates the AWMD can provide a more reasonable connectivity structure to depict the semantic dependency of neighborhood. This is not only because the AWMD can integrate the inherent semantics of words through word embedding technology, but it also can further achieve good modeling of semantic connections between synonyms through the word weighted matching process based on the attention mechanism. Meanwhile, $\text{STE}-\text{AEs}^{knn}$ presents a worse performance than $\text{STE}-\text{AEs}^{knn\&sparse}$, which indicates the sparsity regularization have a positive effect on improving the discriminability of low-dimensional embeddings. As discussed above, the minimization of sparsity regularization will guide a certain proportion of dimension values that are infinitely close to zero. Therefore, $\text{STE}-\text{AEs}^{knn\&sparse}$ tends to distribute the intrin-sic discriminative explanatory factors in those dimensions with larger values (called discriminative dimensions) in the low-dimensional embeddings, which means those similar low-dimensional embeddings from nearby short texts may have similar discriminative dimensions yet those distinct low-dimensional embeddings from non-neighbor short texts may have distinct discriminative dimensions. In other words, the sparsity regularization helps to increase the combination space of discriminative dimensions, thereby helping to expand the discriminability between low-dimensional embeddings.

4) COMPREHENSION OF LOW-DIMENSIONAL EMBEDDINGS

Finally, we provided understandable keywords to interpret what meaning is implied by each dimension of the low-dimensional embeddings. Specifically, based on

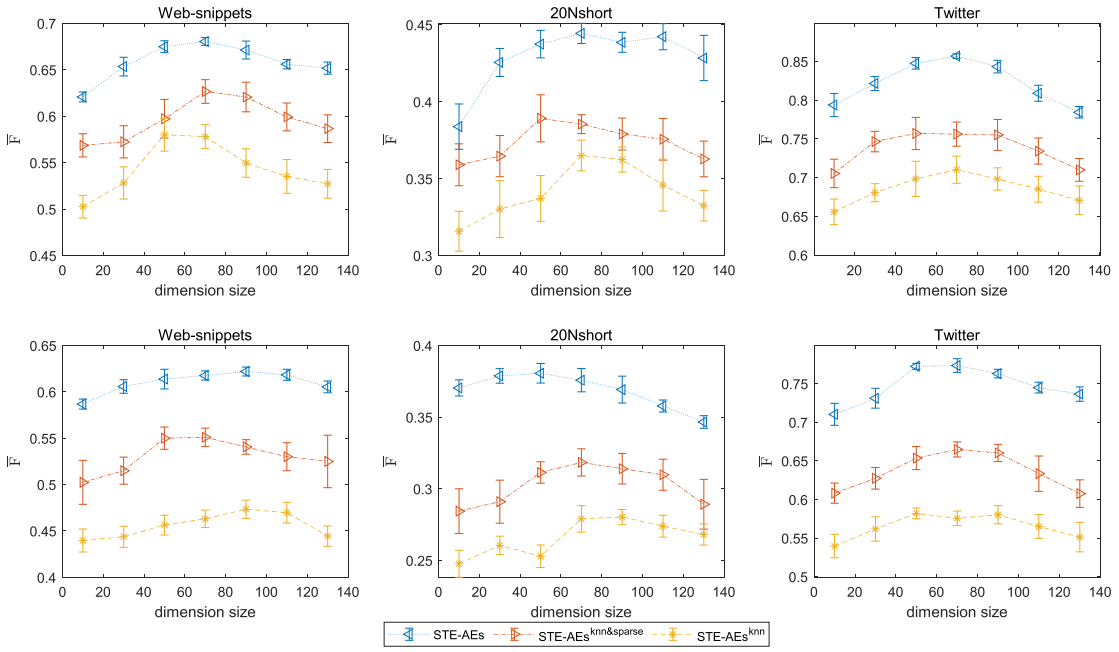

FIGURE 10. Average classification performance of SVM (top) and 1-NN (bottom) for three datasets, where each point consisting of a mean value as well as standard deviations.

TABLE 2. Top 5 word to topic document of 20Nshort and web-snippets (dimension number = 10).

model	Web-snippets				20Nshort				Twitter			
STE-AEs	school game film science graduate	music pop arts album lyrics	navy army weapons war nuclear	market stock finance financial exchange	server install disk window unix	gun israel clinton waco weapon	god jesu soul word bibl	game team player season win	google IT android nexus search	iphone ios store twitter ipad	window s microso ft phone outlook ballmer	icould twitter amaze great apple
TRNMF	sports footbal l fans soccer news	health calorie food preventio n healthy	party democrati c governme nt congress political	computer services market web programmi ng	gov sinc emplo y nasa earth	christia n religio n god atheist belief	mhz cpu clock faster process or	gun weapo n contro l arm firear m	bill ios data ipad phone	microso ft iphone remain store web	android beautifu l James app cloud	ios twitter IT faceboo k apple google
PTM	exporte r pressur e news oversea calorie	gov los court news football	network com xml info software	republic gov united congress gov	claim evid arm game true	church god cathol Christ spirit	season team player year fun	power circuit signal radio audio	twitter microso ft google een window s	cream ballmer iphone twitter steve	nexus volume humble web cloud	amaze apple icloud iphone google
BTM	tennis music ski buy movie	access movies java policy film	online yahoo web directory search	wheels electric cars gear fuel	display color image versio n graphi c	christia n jesu soul word bibl	year player team good biggest	mac apple ntsc intern chip	android phone store nexus google	google ios twitter microso ft cloud	microso ft web outlook phone en	humble twitter apologiz e cloud window s

the one of five iterations estimation performance in the above unsupervised and supervised experiments, we selected 10-dimensional embeddings of training sets of three datasets by the built embedding model. Then, according to the sorting of each dimension of embeddings (the activation of each the hidden neuron), we selected the top-200 short texts to construct interpretation subsets. Next, we built an RF model using one interpretation subset independently and selected the top-5 words (variables) based on the VI to interpret the meaning of each dimension of low-dimensional embeddings.

We compared STE-AEs with other topic-based approaches like TRNMF, PTM and BTM. Table 2 shows part of the selected most important five words over three datasets.

From Table 2, we can see that STE-AEs and TRNMF can provide more informative and understandable words as expected for embeddings interpretation. The words discovered by PTM and BTM are confusing, like ''claim evid arm game true'' of PTM and ''tennis music ski buy movie'' of BTM. Therefore, the evidence demonstrates that the proposed method can alleviate the issue that the neural network-based embedding methods fail to effectively interpret the meaning of the embeddings. Besides, different from the Topic-based methods our method provides a post-processing solution for low-dimensional embeddings interpretation, reducing the complexity of the model and improving practical applications.

V. CONCLUSION

In this paper, we propose a manifold-inspired Short Texts Embedding approach, STE-AEs, which aims to extract low-dimensional embeddings of short texts against data-sparse issue. To avoid the impact of the sparsity on the similarity measurement of nearby short text, STE-AEs develops a robust similarity measurement, AWMD, for KNN construction, then regularizes the training of AEs by imposing an additional minimization of the cross-entropy of nearby texts' embeddings to preserve the local geometrical pattern of neighborhood, which is beneficial to alleviate the data-sparse issue of short text. As a result, under the regularized training framework, STE-AEs can provide an explicit parametrized mapping function between observations and embeddings, ensuring the embeddings are local invariant around the neighborhoods, and improve the discriminative effect. The evidence on three real-world short text corpuses demonstrate that STE-AEs can capture the intrinsic discriminative explanatory factors, improving the performance of short text clustering and classification. Additionally, STE-AEs develops a post-processing solution that build a RF regression model to find some informative and understandable words using the activation values of the encoder and the training set. The exploration of low-dimensional embeddings interpretation yields inspirational results that some informative and understandable words are selected, improving the semantic interpretability of low-dimensional embeddings.

REFERENCES

[1] W. Liang, H. Xie, Y. Rao, R. Y. K. Lau, and F. L. Wang, ''Universal affective model for Readers' emotion classification over short texts,'' *Expert Syst. Appl.*, vol. 114, pp. 322–333, Dec. 2018.
[2] J. Yang, Y. Li, C. Gao, and W. Dong, ''Entity disambiguation with context awareness in user-generated short texts,'' *Expert Syst. Appl.*, vol. 160, Dec. 2020, Art. no. 113652.
[3] W. Gao, M. Peng, H. Wang, Y. Zhang, W. Han, G. Hu, and Q. Xie, ''Generation of topic evolution graphs from short text streams,'' *Neurocomputing*, vol. 383, pp. 282–294, Mar. 2020.
[4] W. Liu, G. Cao, and J. Yin, ''Bi-level attention model for sentiment analysis of short texts,'' *IEEE Access*, vol. 7, pp. 119813–119822, Aug. 2019.
[5] X. Hu, X. Zhang, C. Lu, E. K. Park, and X. Zhou, ''Exploiting Wikipedia as external knowledge for document clustering,'' in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2009, pp. 389–396.
[6] Y. Song, H. Wang, Z. Wang, H. Li, and W. Chen, ''Short text conceptualization using a probabilistic knowledgebase,'' in *Proc. 52nd Int. Conf. Artif. Intell. (IJCAI)*, Jul. 2011, pp. 2330–2336.
[7] H. Huang, Y. Wang, C. Feng, Z. Liu, and Q. Zhou, ''Leveraging conceptualization for short-text embedding,'' *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 7, pp. 1282–1295, Jul. 2018.
[8] Y. Chen, H. Zhang, R. Liu, Z. Ye, and J. Lin, ''Experimental explorations on short text topic mining between LDA and NMF based schemes,'' *Knowl.-Based Syst.*, vol. 163, pp. 1–13, Jan. 2019.
[9] J. Li, G. Huang, J. Chen, and Y. Wang, ''Short text understanding combining text conceptualization and transformer embedding,'' *IEEE Access*, vol. 7, pp. 122183–122191, Aug. 2019.
[10] Y. Wang, H. Zhang, G. Shi, Z. Liu, and Q. Zhou, ''A model of text-enhanced knowledge graph representation learning with mutual attention,'' *IEEE Access*, vol. 8, pp. 52895–52905, Mar. 2020.
[11] Y. Bengio, A. Courville, and P. Vincent, ''Representation learning: A review and new perspectives,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.
[12] T. Hofmann, ''Probabilistic latent semantic indexing,'' in *Proc. 22nd Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 1999, pp. 50–57.
[13] D. M. Blei, A. Y. Ng, and M. I. Jordan, ''Latent Dirichlet allocation,'' *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.
[14] H. Wu, J. Bu, C. Chen, J. Zhu, L. Zhang, H. Liu, C. Wang, and D. Cai, ''Locally discriminative topic modeling,'' *Pattern Recognit.*, vol. 45, no. 1, pp. 617–625, Jan. 2012.
[15] X. Cheng, X. Yan, Y. Lan, and J. Guo, ''BTM: Topic modeling over short texts,'' *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 12, pp. 2928–2941, Dec. 2014.
[16] Y. Zuo, J. Wu, H. Zhang, H. Lin, F. Wang, K. Xu, and H. Xiong, ''Topic modeling of short texts: A pseudo-document view,'' in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 2105–2114.
[17] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, ''Distributed representations of words and phrases and their compositionality,'' in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 26, 2013, pp. 3111–3119.
[18] C. Ma, W. Xu, P. Li, and Y. Yan, ''Distributional representations of words for short text classification,'' in *Proc. 1st Workshop Vector Space Modeling Natural Lang. Process.*, 2015, pp. 33–38.
[19] C. De Boom, S. Van Canneyt, T. Demeester, and B. Dhoedt, ''Representation learning for very short texts using weighted word embedding aggregation,'' *Pattern Recognit. Lett.*, vol. 80, pp. 150–156, Sep. 2016.
[20] J. Xu, Y. Cai, X. Wu, X. Lei, Q. Huang, H.-F. Leung, and Q. Li, ''Incorporating context-relevant concepts into convolutional neural networks for short text classification,'' *Neurocomputing*, vol. 386, pp. 42–53, Apr. 2020.
[21] H. Larochelle and S. Lauly, ''A neural autoregressive topic model,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 2708–2716.
[22] S. T. Roweis, ''Nonlinear dimensionality reduction by locally linear embedding,'' *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.
[23] M. Belkin and P. Niyogi, ''Laplacian eigenmaps and spectral techniques for embedding and clustering,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 585–591.
[24] J. B. Tenenbaum, ''A global geometric framework for nonlinear dimensionality reduction,'' *Science*, vol. 290, no. 5500, pp. 2319–2323, Dec. 2000.
[25] C. Wei, S. Luo, J. Zhang, and L. Pan, ''Short text manifold representation based on AutoEncoder network,'' *J. Zhejiang Univ. (Eng. Sci. Ed.)*, vol. 49, no. 8, pp. 1591–1599, Sep. 2015.
[26] S. Banerjee, K. Ramanathan, and A. Gupta, ''Clustering short texts using Wikipedia,'' in *Proc. 30th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 2007, pp. 787–788.
[27] J. Wang, Z. Wang, D. Zhang, and J. Yan, ''Combining knowledge with deep convolutional neural networks for short text classification,'' in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Melbourne, VIC, Australia, Aug. 2017, pp. 2915–2921.
[28] G. E. Hinton and R. R. Salakhutdinov, ''Replicated softmax: An undirected topic model,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 1607–1614.
[29] H. Larochelle and S. Lauly, ''A neural autoregressive topic model,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 2717–2725.
[30] Y. Wang, H. Huang, C. Feng, Q. Zhou, J. Gu, and X. Gao, ''CSE: Conceptual sentence embeddings based on attention model,'' in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics, Long Papers*, vol. 1, 2016, pp. 505–515.

[31] J. Xu, B. Xu, P. Wang, S. Zheng, G. Tian, J. Zhao, and B. Xu, "Self-taught convolutional neural networks for short text clustering," *Neural Netw.*, vol. 88, pp. 22–31, Apr. 2017.
[32] D. Cai, Q. Mei, J. Han, and C. Zhai, "Modeling hidden topics on document manifold," in *Proc. 17th ACM Conf. Inf. Knowl. Mining (CIKM)*, 2008, pp. 911–920.
[33] D. Cai, X. Wang, and X. He, "Probabilistic dyadic data analysis with local and global consistency," in *Proc. 26th Annu. Int. Conf. Mach. Learn. (ICML)*, 2009, p. 14.
[34] S. Huh and S. E. Fienberg, "Discriminative topic modeling based on manifold learning," *ACM Trans. Knowl. Discovery Data*, vol. 5, no. 4, pp. 1–25, Feb. 2012.
[35] Y. Bengio, "Learning deep architectures for AI," *Found. Trends Mach. Learn.*, vol. 2, no. 1, pp. 1–127, 2009.
[36] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, "Contractive auto-encoders: Explicit invariance during feature extraction," in *Proc. Int. Conf. Mach. Learn.*, 2011, pp. 833–840.
[37] W. Wang, Y. Huang, Y. Wang, and L. Wang, "Generalized autoencoder: A neural network framework for dimensionality reduction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2014, pp. 496–503.
[38] C. Wei, S. Luo, X. Ma, H. Ren, J. Zhang, and L. Pan, "Locally embedding autoencoders: A semi-supervised manifold learning approach of document representation," *PLoS ONE*, vol. 11, no. 1, Jan. 2016, Art. no. e0146672.
[39] H. Narayanan and S. Mitter, "Sample complexity of testing the manifold hypothesis," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 1786–1794.
[40] T. Jebara, J. Wang, and S.-F. Chang, "Graph construction and b-matching for semi-supervised learning," in *Proc. 26th Annu. Int. Conf. Mach. Learn. (ICML)*, 2009, pp. 441–448.
[41] C. Wei, S. Luo, J. Guo, Z. Wu, and L. Pan, "Discriminative locally document embedding: Learning a smooth affine map by approximation of the probabilistic generative structure of subspace," *Knowl.-Based Syst.*, vol. 121, pp. 41–57, Apr. 2017.
[42] F. Yi, B. Jiang, and J. Wu, "Topic modeling for short texts via word embedding and document correlation," *IEEE Access*, vol. 8, pp. 30692–30705, Feb. 2020.
[43] P. Bicalho, M. Pita, G. Pedrosa, A. Lacerda, and G. L. Pappa, "A general framework to expand short text for topic modeling," *Inf. Sci.*, vol. 393, pp. 66–81, Jul. 2017.
[44] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.
[45] J. Cheng, L. Dong, and M. Lapata, "Long short-term memory-networks for machine reading," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 551–561.
[46] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, no. 12, pp. 3371–3408, Dec. 2010. [Online]. Available: http://www.jmlr.org/papers/v11/vincent10a.html.
[47] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layerwise training of deep networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 19, 2007, pp. 153–160.
[48] A. Hadifar, L. Sterckx, T. Demeester, and C. Develder, "A self-training approach for short text clustering," in *Proc. 4th Workshop Represent. Learn. NLP (RepL4NLP)*, 2019, pp. 194–199.
[49] X. H. Phan, L. M. Nguyen, and S. Horiguchi, "Learning to classify short and sparse text & Web with hidden topics from large-scale data collections," in *Proc. 17th Int. Conf.World Wide Web (WWW)*, 2008, pp. 91–100.
[50] L. Lovász and M. D. Plummer, *Matching Theory*. Amsterdam, The Netherlands: North Holland, 1986.
[51] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

CHAO WEI received the Ph.D. degree from the Information System and Security-Countermeasures Experimental Center, School of Information and Electronics, Beijing Institute of Technology, Beijing. He is currently a Postdoctoral Fellow with the National Engineering Research Center of Science and Technology Information, Institute of Scientific and Technical Information of China. His current research interests include representation learning, information retrieval and question answering, information extraction, and knowledge graph.

LIJUN ZHU received the Ph.D. degree in agricultural electrification and automation from the School of Information and Electrical Engineering, China Agricultural University, Beijing. He is currently the Director and a Professor of the National Engineering Research Center of Science and Technology Information, Institute of Scientific and Technical Information of China. His current research interests include semantic Web, Web service and knowledge technology in science and technology information service, knowledge engineering, information engineering, biology and health big data, and information resource management.

JIAOXIANG SHI received the B.M. degree in information management and information system from the Wuhan University of Science and Technology, Wuhan, in 2018. He is currently pursuing the M.S. degree with the National Engineering Research Center of Science and Technology Information, Institute of Scientific and Technical Information of China. His current research interests include knowledge engineering and knowledge discovery, text data visualization, information extraction, and knowledge graph.