An Unsupervised Learning Short Text Clustering Method

Zuhua Dai*, Kelong Li, Hongyi Li and Xiaoting Li
College of Computer Science and Engineering, Northwest Normal University, Lanzhou, 730070, China

*Corresponding author’s e-mail: 2812704000@qq.com

Abstract. Due to the continuous development of Natural Language Processing (NLP), the task of short text categorization has been paid more and more attention. In short text clustering, the high-dimensional sparseness of text representation matrix becomes a challenging problem. This paper proposes a deep embedded method for feature extraction and clustering allocation using auto encoder of sentence distributed embedding. This method maps from data space to low-dimensional feature space and iteratively optimizes clustering targets. Experimental results on three short Chinese text data sets verify the effectiveness of the method. Moreover, it is superior to the existing correlation clustering methods.

1. Introduction
With the rapid development of the Internet, it has become an important way for people to obtain information on the Internet. At the same time, it also gave birth to various short text data such as microblogs, Taobao evaluations, news headlines, etc. However, compared with long text, short text has the problems of data sparsity and irregularity, most words only appear once in short text. Most of the existing clustering methods [1] [2] are based on the traditional vector space model (VSM) to represent text. Yang and Zhang et al.[3] [4] proposed a semantic-based short text clustering method, Hu and Banerjee et al. [5] [6] extended short text using Wikipedia articles, Hu ando and Wei et al.[7][8] proposed different methods using rich ontology text representation. These methods can improve the accuracy of similarity measurement between short texts. However, compared with English short texts, words in Chinese short texts are complex, diverse and large in scale, which leads to poor accuracy and low efficiency in semantic expansion of words.

In view of the problems existing in traditional short texts, deep neural networks have significant advantages in feature extraction. The depth clustering algorithm based on depth neural network feature extraction first embeds high-dimensional data into low-dimensional space, and then adopts the clustering algorithm. Deep Embedded Clustering (DEC) [9] is a pioneer of deep clustering. It uses autoencoder to convert data into low-dimensional space and uses allocation layer to refine potential space for clustering. The self-learning convolution neural network (STC) framework proposed by Xu et al. [10] uses dimension reduction technology to generate the auxiliary targets of the neural network framework, convolution neural network (CNN) learning feature representation to reconstruct these auxiliary targets, and clustering the training representation of CNN using k-means algorithm.

Similar to Xie et al. [9], we use a multi-stage method to train the encoder to convert embedding into potential space before clustering. However, we made two key modifications, which are different from CNN-based encoders [10]. In order to maintain the efficiency and accuracy of short text clustering, we
use BERT [11] to perform vector conversion on demand text. At the beginning of text clustering, we use the self-training of soft cluster classification to fine-tune the encoder. We introduced our model in detail in Section 2 of this article. In Section 3, we use accuracy (ACC) and normalized mutual information (NMI) as evaluation indicators to evaluate our model, and verify it in three Chinese short text data sets. Experiments show that our model has better effect than the more complex neural network architecture.

2. Methodology

2.1. Text preprocessing

In this paper, the obtained Chinese short text data set is processed by NLTK segmentation and removal of stop words, and then the original samples are mapped to the vector space using the word vector tool. In view of the fact that Word2vec training does not consider the impact of word order on the sentence, the sentence cannot have an overall meaning, this paper uses BERT [11] to complete the vector conversion of short text. The text preprocessing model is shown in Figure 1.

![Figure 1. Text Preprocessing](image)

BERT is the first unsupervised deep two-way system used in pre-training NLP. As the Bi-LSTM structure increases with the depth of the model, the entire sentence forms a self-contained cycle of semantics before and after the sentence. The BERT model uses Masked LM to break the one-way limitation of the standard language model, randomly covering the words of the input sentence, and based on the surrounding words Context prediction covering words [11].

Get the maximum length of all short texts in the corpus in the preprocessing of the required text and the number of unique words, and then add each short text to the maximum length to facilitate the batch processing of the neural network. BERT encodes the vocabulary, converts each word into a k-dimensional vector, and then represents each required text as an n*k-dimensional vector. This vector completely retains the semantic information and position information in the text data set, so this paper uses this vector as the original feature vector of the short text data set.

2.2. Self-training model

2.2.1. Auto encoder. Auto-Encoder (AE) [12] is consists of encoder and decoder. After the original feature vector \( x \) obtained by BERT is input into the encoder, nonlinear changes are made, and then a coding result \( y \) is obtained after being processed by the activation function. The calculation formula is shown in (1). The encoding result \( y \) is input into the decoder, and the vector \( z \) is finally obtained after reconstruction. The calculation formula is shown in (2).

\[
y = f_\theta = s(Wx + b)
\]

\[
z = g_\theta(y) = s(W'y + b')
\]
The encoded parameter is \( \theta = \{W, b\} \) and the decoded parameter is \( \theta' = \{W', b'\} \). Where, \( W \) is a weight matrix of \( d' \times d \), and \( W' \) is the transposition of \( W \), that is \( W' = W^T \), \( b \) and \( b' \) are the corresponding bias vectors. The optimization goal is to restore the reconstructed vector \( z \) to \( x \) as soon as possible, which is to reduce the loss caused by the reconstruction as much as possible and obtain the best parameters \( \theta' \) and \( \theta'^* \), as shown in Equation (3):

\[
\theta', \theta'^* = \arg\min_{\theta, \theta'} L(x, z) = \arg\min_{\theta, \theta'} L(x, g_{\theta'}(f_{\theta}(x)))
\]

### 2.2.2. Self-training method

After pre-training with autoencoder, we obtain the feature of nonlinear mapping embedded from BERT to low-dimensional representation, and apply clustering algorithm to the feature. Next, we use the self-training phase to improve clustering: we allocate initial clustering centers and then alternate between two steps: (i) first calculate the probability allocated to each clustering center point; (ii) calculate that auxiliary probability distribution and use it as a target network for the encoder. The network weights and clustering centers are iteratively updated until the convergence criterion is met, as shown in Figure 2.

![Figure 2. self-training model.](image)

For step (i), similar to the t-SNE algorithm [13], the similarity between the points around the center point \( u_j \) and \( u_j \) itself obeys the distribution. Therefore, \( t \)-distribution is used to measure the similarity between the embedded point and the center point, and then all clustering centers are normalized to obtain the soft distribution degree between the embedded points \( z_j \) and the centroids \( u_j \), as shown in Equation (4):

\[
q_{ij} = \frac{(1 + ||z_i - u_j||^2)^{-1}}{\sum_{j'}(1 + ||z_i - u_{j'}||^2)^{-1}}
\]

In step (ii), similar to Xie et al. [9] We use the auxiliary target distribution \( P \), which is higher accuracy than \( q_{ij} \) in Equation (4). Firstly, \( q \) is raised to the second power, which makes the distribution pay more attention to data samples with high confidence. Then normalization is carried out according to the
frequency of each cluster, which can prevent large cluster groups from interfering with the hidden feature space. Finally, the target distribution is normalized so that the probability of the target distribution in $P$ is shown in (5):

$$
p_{ij} = \frac{q_{ij}^2}{\sum_{i} (q_{ij}^2) / \sum_{i} \sum_{j} q_{ij}^2)
$$

(5)

The model learns from this high-confidence distribution, and hopes that the probability distribution $q_{ij}$ of the soft distribution degree is consistent with the auxiliary target distribution $p_{ij}$, so their KL divergence is defined as the loss function as Equation (6):

$$
L = KL(P \| Q) = \sum_{i} \sum_{j} p_{ij} \log \frac{p_{ij}}{q_{ij}}
$$

(6)

The above strategy is a form of self-supervision [15]. Use standard clustering algorithm centroid to initialize the weights of the clustering layer, and then use high-confidence prediction to fine-tune the encoder and centroid. After the process converges, the short text is encoded and the k-means is used for the final clustering allocation.

3. Experimental results and analysis

3.1. Data

We evaluated our model on three Chinese short text clustering data sets. (1) Baidu Q&A Data: Using Python web crawler tool Scrapy to crawl the article reviews of business, entertainment, life, education, health, culture, sports and games (8 categories) in Baidu Q&A as experimental data, each category contains 1000 data sets. (2) Today's Headlines Data: Today's Headline Data Set has 382,688 data sets distributed in 15 categories. Text randomly selects 1,000 data sets from each category for experiments. Table 1 summarizes the types of short text data sets proposed. (3) Sina News Data: Sina News Data contains 740,000 news documents and divided into 14 categories. Text randomly selects 1,000 data sets from each category for experiments. Table 1 summarizes the types of short text data sets proposed.

Table 1. Experimental data set information

| Dataset          | Data type                                                                 |
|------------------|---------------------------------------------------------------------------|
| Baidu Q&A        | Business, Entertainment, Life, Education, Health, Culture, Sports, Games   |
| Today’s Headlines| Stories, Culture, Entertainment, Sports, Finance, Real estate, Automobiles, Education, Technology, Military, Tourism, International, Stocks, Agriculture, Games, |
| Sina dataset     | Finance, Lottery, Real estate, Stocks, Household, Education, Technology, Society, Fashion, Political, Sports, Constellations, Games, Entertainment |

3.2. Experimental Setup

In order to verify the superiority of the algorithm in this paper, the experimental results of the algorithm in this paper are compared with clustering represented by TF-IDF and SIF and $STC^2$ model proposed by Xu et al.[9]. Next, when training data sets from the encoder, the weights are initialized with standard Gaussian distribution random numbers, the network dimension is set to $d$: 500: 500: 2000: 20 for all data sets, where $d$ represents the spatial dimension of the original data. When KL divergence is
minimized, the learning rate of random gradient descent is set to 0.01 and the momentum value is set to 0.9.

In K-means clustering, the initial centroid has a great influence on the accuracy of the experiment. In order to obtain a fair result, we randomly initialize the center of mass and restart K-means 100 times, select the best center of mass from which, and take the average value from 5 experimental results.

3.3. Evaluation Index

We use two widely used performance indexes as evaluation indexes: clustering accuracy (ACC) and normalized mutual information (NMI).

NMI measures the information shared between prediction task $\alpha$ and basic truth task $\beta$. The definition is shown in Equation (7):

$$NMI(\alpha, \beta) = \frac{I(\alpha, \beta)}{\sqrt{H(\alpha)H(\beta)}}$$  \hspace{1cm} (7)

Among them: $I$ is mutual information, $H$ is entropy. The larger the value of NMI, the closer the two divisions are. When the NMI reaches the maximum value, the number of clusters is the optimal number of clusters for the model.

The definition of clustering accuracy is shown in Equation (8):

$$ACC = \frac{\sum_{i=1}^{N} \delta(l_i = map(c_i))}{N}$$  \hspace{1cm} (8)

Where $\delta()$ is the indicator function, $c_i$ is the cluster label of $x_i$, $map()$ uses the Hungarian algorithm to convert the cluster label $c_i$ to its group label, and $y_i$ is the true group label of $x_i$. The NMI results and accuracy of the model and comparison test in this paper are shown in Table 2.

Table 2. Clustering results (accuracy ACC and normalized mutual information NMI) using this method and various methods in three short text sets.

| Method    | Baidu Q&A ACC | Baidu Q&A NMI | Today’s Headlines ACC | Today’s Headlines NMI | Sina dataset ACC | Sina dataset NMI |
|-----------|---------------|---------------|------------------------|-----------------------|------------------|------------------|
| TF-IDF    | 28.3 ± 2.3    | 19.8 ± 2.5    | 24.1 ± 2.9             | 16.5 ± 2.6            | 21.4 ± 3.2       | 18.4 ± 3.2       |
| SIF       | 64.6 ± 1.9    | 53.2 ± 1.2    | 50.7 ± 0.8             | 42.6 ± 0.4            | 54.1 ± 1.3       | 46.3 ± 0.8       |
| STC²      | 78.3 ± 3.1    | 65.3 ± 2.3    | 63.3 ± 1.8             | 53.5 ± 2.4            | 61.2 ± 1.4       | 55.7 ± 0.6       |
| OurMethod | **83.4 ± 1.1**| **61.4 ± 0.8**| **67.9 ± 1.4**         | **58.4 ± 1.8**        | **66.3 ± 2.1**   | **59.3 ± 1.3**   |

General low-dimensional vector representations (such as TF-IDF or SIF embedding) are beneficial in many natural language processing tasks, and for deep embedding clustering, additional fine-tuning and self-training can improve cluster accuracy. The evaluation results show that compared with the STC² model, our method has advantages in all indicators except one indicator, and the improvement of clustering quality is also obvious in Figure 2.
4. Conclusions

This paper first uses the BERT model to obtain word vectors and proposes a multi-stage clustering method. Then, a deep embedded clustering model adopts autoencoder architecture, a self-training method is used to fine-tune the cluster. We verify the accuracy of the method in three Chinese short text data sets.

References

[1] Suo H G and Wang Y W 2018 An improved k-means algorithm for text clustering Journal of Shandong University(Natural Science) 43 (1) 61 - 64.
[2] Zhang X, Wang S Z and Yin Y X 2010 Research of text clustering based on fuzzy granular computing Computer Science 37 (2) 209 - 211.
[3] Yang Z, Wang L T and Lai Y X 2014 Research on Web Comment Clustering Based on Improved Semantic Distance Software Journal 25 (12) 2777 - 2789.
[4] Zhang Q, Wang H J and Wang L W 2016 A Short Text Clustering Algorithm Combined with Context Semantics Computer Science 43 (s2) 443 - 446.
[5] Hu X H, Zhang X D, Lu C M, Eun K. Park and Zhou X H 2009 Exploiting Wikipedia as external knowledge for document clustering In Proceedings of ACM SIGKDD 389 – 396.
[6] Banerjee S, Ramanathan K and Gupta A 2007 Clustering short texts using Wikipedia In Proceedings of ACM SIGIR 787 – 788.
[7] Hotho A, Staab S, and Stumme G 2003 Ontologies improve text document clustering In Proceedings of ICDM 541 – 544.
[8] Wei T T, Lu Y H, Chang H Y and Bao X Y 2015 A semantic approach for text clustering using WordNet and lexical chains Expert Systems with Applications 42 2264 – 2275.
[9] Xie J Y, Girshick R and Farhadi A 2016 Unsupervised deep embedding for clustering analysis In Proceedings of ICML 478 – 487.
[10] Xu J M, Xu B, Wang P, Zheng S C, Tian G H and Zhao J 2017 Self-taught convolutional neural networks for short text clustering Neural Networks 88 22 – 31.
[11] Devlin J, Chang M W, Lee K and Toutanova K 2013 Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding Encyclopedia of Systems Biology
[12] Alain G and Bengio Y 2012 What Regularized Auto-Encoders Learn from the Data Generating Distribution Journal of Machine Learning Research 15 (1) 3563 - 3593.
[13] Maaten L and Hinton G 2008 Visualizing data using t-SNE Journal of machine learning research 9 2579 – 2605.
[14] Maaten L V D 2009 Learning a parametric embedding by preserving local structure Journal of Machine Learning Research 5 384 – 391.
[15] Nigam K and Ghani R 2000 Analyzing the effectiveness and applicability of co-training In Proceedings of CIKM 86 – 93.