Scientific Research Publishing

Journal of Data Analysis and Information Processing, 2024, 12, 597-611
https://www.scirp.org/journal/jdaip
ISSN Online: 2327-7203
ISSN Print: 2327-7211

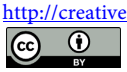
Enhancing BERTopic with Pre-Clustered Knowledge: Reducing Feature Sparsity in Short Text Topic Modeling

Qian Wang, Biao Ma*

Glorious Sun School of Business and Management, Donghua University, Shanghai, China
Email: *mabiao@dhu.edu.cn

How to cite this paper: Wang, Q. and Ma, B. (2024) Enhancing BERTopic with Pre-Clustered Knowledge: Reducing Feature Sparsity in Short Text Topic Modeling. *Journal of Data Analysis and Information Processing*, **12**, 597-611.
https://doi.org/10.4236/jdaip.2024.124032

Received: October 23, 2024
Accepted: November 18, 2024
Published: November 21, 2024



Abstract

Modeling topics in short texts presents significant challenges due to feature sparsity, particularly when analyzing content generated by large-scale online users. This sparsity can substantially impair semantic capture accuracy. We propose a novel approach that incorporates pre-clustered knowledge into the BERTopic model while reducing the l2 norm for low-frequency words. Our method effectively mitigates feature sparsity during cluster mapping. Empirical evaluation on the StackOverflow dataset demonstrates that our approach outperforms baseline models, achieving superior Macro-F1 scores. These results validate the effectiveness of our proposed feature sparsity reduction technique for short-text topic modeling.

Keywords

Topic Model, BERTopic, Short Text, Feature Sparsity, Cluster

1. Introduction

With the rapid development of technology, massive information on the Internet is growing exponentially, and the amount of unstructured text data is increasing. For text data mining, traditional research perspectives are mostly based on long texts, such as news, posts, etc. However, with the functional evolution of platforms and changes in user behavior habits, short text comments can be found everywhere on microblogs, short videos, e-commerce, and other types of platforms. The investigation of topics in short texts is crucial for content analysis; therefore, how to quickly and accurately mine information contained in short texts has been an issue of interest in recent years. The information embedded in the text is an issue

that deserved attention in recent years.

Topic modeling is one of the most important techniques in text mining, which is widely used in information retrieval [1] [2], content recommendation [3] [4], and intelligence analysis [5]. Traditional topic models, such as PLSA and LDA, usually reveal potential topics in text corpora by implicitly capturing document-level word co-occurrence patterns. However, unlike the information-rich long documents traditionally studied, most words in short texts appear only once, which is not enough to support the acquisition of associative information, presenting sparse semantic features and more noise, making the topic modeling problem for this type of text challenging. Therefore, it is an important issue in information processing of short text to effectively mine implicit information so that topic modeling can be better applied to short text and improve the clustering accuracy. For the topic modeling problem of short texts, most of the early research alleviates sparsity by extending short texts, using external knowledge, enhancing topic hypotheses, etc. However, heuristic data aggregation methods are highly dependent on data, and external knowledge is equally dependent on their accuracy. The bi-term co-occurrence model is prone to mixing redundant information and still faces the feature sparsity problem. Self-aggregation-based models, although they can capture topic information to some extent, cannot achieve high accuracy in clustering due to ineffective utilization of contextual information, bags of words may not accurately represent documents, etc.

With wide application of language models, their ability to learn complex linguistic structures and semantic laws is applied to the topic modeling field to make up for shortcomings in semantic captures, such as LDA + SentenceBERT [6], GA + WGSDMM [7], LDA/GSDMM + BERT [8], etc. When using advanced language modeling to process long texts, it can learn long-distance dependencies and capture complex structures and deep semantics through a large number of parameters and deep network structures, thus improving the ability to grasp the text as a whole. This performs very effectively in understanding and analyzing long articles, news, etc. However, when these models are applied to short text processing, the sparsity problem still exists. The lack of contextual information still makes it difficult to provide rich features for model learning, and the brevity of short texts also leads to increased semantic ambiguity. Therefore, comprehension and topic modeling of short texts remains an urgent problem. It is necessary to develop new methods that are more suitable for the characteristics of short texts to improve semantic comprehension and accuracy.

In 2022, BERTopic was proposed, which uses advanced language models and class-based TF-IDF to generate topic representations. It outperforms other classical topic models, such as LDA, in terms of topic consistency, diversity, and runtime. However, BERTopic is oriented to regular text and still adopts regular processing even when dealing with short texts with sparse features, which easily leads to overfitting and affects the accuracy of the topic assignment. Moreover, in the semantic vector space generated using language models, the l2 paradigm

of low-frequency words is larger, *i.e.*, they are farther away from the origin. The sparse distribution makes them show significant anisotropy, and thus, there is still room for improvement in BERTopic with short texts. Based on the aforementioned idea of combining the language model and topic information, this paper tries to improve the clustering process of BERTopic by utilizing pre-clustered knowledge. Adding more features to measure sentence similarity, feature fusion, and dimensionality reduction are performed to mitigate the effect of l2 paradigms for low-frequency words and to improve the accuracy of topic assignments in short texts.

2. Related Work

2.1. Traditional Short-Text Topic Model

In general, topic models are commonly used to extract topic information from text corpora. Traditional topic models use word cooccurrence information to identify topics, such as LSA [9], PLSA [10], etc. Especially LDA [11], which has good generalization ability and extensibility, is widely used to discover hidden topics from text corpora. However, this approach is more suitable for long texts with rich topic matter [12]. Texts from the Internet are usually very short with limited contexts, and most of the above models and their variants do not take into account the specificity of short texts, and thus face serious data sparsity problems when applied to short texts.

A common solution to this problem is to extend short texts and aggregate them into lengthy pseudo-documents to enrich the representations of short texts, e.g., based on user groups [13], based on synonyms [14], based on the time of text publication [15] [16], etc. Alternatively, it is extended using external knowledge, such as WordNet [17], transfer learning with the help of long texts [18], and so on. Other methods are modified in mathematical theory for short texts, such as the bi-term topic model BTM [19], which uses bi-term co-occurrence information to model the corpus generation process.

GSDMM is an unsupervised topic model. It is solved in a similar way to LDA, except that it assumes that each document contains only one potential topic. Such an assumption is more in line with the actual situation of short texts such as comments, headlines, and other types of short texts. The model is based on the Dirichlet Multinominal Mixture Model (DMM), which generates documents and approximates the solution using collapsed Gibbs sampling. GSDMM implements clustering based on the two principles of completeness and consistency, and its performance is significantly better than that of K-means, HAC, and DMAFP. The model assumes that documents are generated according to a mixed polynomial distribution and that topics and documents correspond to one another.

However, these approaches are based on statistical knowledge and cannot solve the ambiguity problem. That is, multiple expressions of a certain meaning may be classified under different topics. Comprehension bias affects the accuracy of the

results, so language models are gradually being used to explore more effective solution strategies.

2.2. Short-Text Topic Models Using Language Models

Effective embeddings can help recognize information at the syntactic and semantic levels [20] and enhance the semantic consistency of topics in short texts, so embedding-based techniques have been gradually used to improve topic modeling in recent years. For example, Fan *et al.* [21] classified short texts based on word embeddings with bi-directional GRU, WETM [22] and JEA-LDA [23] discovered structural information about topics and words based on word embeddings, and Jia *et al.* [24] used sentence embeddings and CNN to cluster textual features. In addition, embedding information can be combined with methods for extending text. For example, VAETM [25] inferred potential representations of topic distributions of short texts by combining word embeddings with external knowledge; ETM [26] inferred topics by aggregating short texts into pseudo-documents using word embeddings and applying Markov Random Field Regularization models.

Language models facilitate the development of topic modeling [27]. BERTopic [28] is a topic model that performs clustering based on embeddings. It transforms documents into dense vector representations using the pre-trained language model SentenceBERT [29], dimensionality reduction using the UMAP algorithm [30], clustering with HDBSCAN [31], and finally, applying a class-based variant of TF-IDF to extract the topic representation of each cluster. However, when dealing with a large number of short texts, sparse features introduce bias to clusters exclusively by the similarity of sentence embeddings, which makes it difficult for the language model in BERTopic to fully understand semantics.

Some studies introduce topic knowledge into advanced language models, where learning multi-dimensional features contributes to a comprehensive text understanding of the complex properties. Nguyen *et al.* [6] used LDA to obtain probabilistic topic vectors for comment texts and integrated sentence embeddings from SentenceBERT to identify topics; Agarwal *et al.* [7] used GA algorithm to improve WGSDMM for web service clustering; Peinelt *et al.* [8] proposed tBERT model to improve the performance of BERT in semantic similarity detection using topic models such as LDA and GSDMM. Liu *et al.* [32] combined GSDMM with SBERT to effectively extract topics and overall semantic features of short texts. Performs targeted classification tasks that require manual labeling and training. Although the proposed method in this paper also introduces pre-clustered topic knowledge, the splicing is different and it performs entirely in an unsupervised manner without prior labeled data.

3. Methodology

The vectors extracted by BERTopic only consider semantic similarity, so in this paper, we use pre-clustered topic knowledge to supplement the original algorithm. This makes the topic assignment result more reasonable and enhances the ability

to extract more accurate topics. In the following, we first analyze the existing problems of BERTopic and put forward the motivation, then introduce the new method of similarity computation that introduces pre-clustered knowledge. Finally, the specific process of the improved algorithm is described in detail.

3.1. Motivation

The purpose of topic modeling is to cluster texts based on their semantic relationships. Given a document collection $\vec{d}$ containing D short texts, perform clustering to get the result $Z=\{z_1,\cdots,z_D\}$, where $z_d \in \{1,2,\cdots,K\}$ is the topic to which document d belongs and K is the number of topics. Finally, the topic-word distribution representation is constructed based on the importance of the word w in topic z. There are some problems with clustering using text similarity in the clustering process. This paper aims to improve the vector representation used for similarity computation. We use pre-clustered knowledge to reduce the intra-cluster distance and increase the inter-cluster distance, to improve the objective, the optimization problem of clustering accuracy, which is measured by Macro-F1.

Specifically, short texts have more low-frequency words, and their vectors are characterized by large l2 norms, making the distribution sparse. Therefore, in the automatic clustering phase of BERTopic, the density-based HDBSCAN algorithm can discover more and finer-grained clusters. When the number of automatic clusters is larger, it needs to be mapped to a pre-specified number of clusters, a process that requires cosine similarity. However, the characteristics of low-frequency words make the vector space present a narrow cone, showing anisotropy in the spatial distribution, which interferes with the calculation of cosine similarity. The short text has more low-frequency words, which leads to bias in topic vectors due to the uneven distribution of word frequencies. This affects the accuracy of the mapping results based on cosine similarity, so there is room for improvement of the native BERTopic. Moreover, it is challenging to mitigate the effect of anisotropy while maintaining semantic relevance.

3.2. Similarity Calculation Method with Pre-Clustered Knowledge

Cosine similarity is a common method for calculating semantic similarity. For embeddings A and B, the cosine similarity is calculated as shown in Equation (1). The denominator represents the modulus of two embeddings, which can also be regarded as the l2 norm, *i.e.*, the distance from the origin.

$$\text{sim}_{\cos} = \frac{A \cdot B}{\|A\|\|B\|} \tag{1}$$

Specifically, according to Li *et al.* [33], language models can use word co-occurrence as a proxy for semantic similarity. Given the context c, the modeling process of language models can be simplified as:

$$p(x \mid c) = \frac{\exp\left(h_c^{\mathrm{T}} W_x\right)}{\sum_{x'} h_c^{\mathrm{T}} W_{x'}} \tag{2}$$

The similarity of sentence embeddings can be regarded as the similarity of context embeddings $h_c^{\mathrm{T}} h_{c'}$, and the dot product of context embeddings h_c and word embeddings w_x is used as a proxy in Equation (2). According to Li *et al.* [33], in a well-trained language model, $h_c^{\mathrm{T}} W_x$ can be approximately decomposed as:

$$h_c^{\mathrm{T}} W_x = \mathrm{PMI}(x,c) + \log p(x) + \lambda_c \tag{3}$$

where PMI(x,c) denotes pointwise mutual information between x and c. This metric is semantically meaningful, so context vectors imply semantically relevant interactions due to higher-order word co-occurrence relations. Thus the cosine similarity can represent the semantic relations of sentences in the vector space. However, log$p$(x) represents the probability of the occurrence of word x. Low-frequency words have a larger l2 norm and are sparsely distributed, making the shape of vectors in the embedding space a narrow cone, characterized by anisotropy. In short texts, the occurrence probability of low-frequency words is larger, so it will cause bias in the semantic information captured by context vectors in higher-order space because of the distributional characteristics of word frequency.

Considering fewer short text features, this paper introduces pre-clustered knowledge to weaken the influence of word frequency through interaction between different features. In the pre-clustering phase, several potential problems may arise when the external corpus is used. On the one hand, pre-clustering is an unsupervised process and cannot adjust the learning direction of the model in contrast to supervised learning with predefined labels. Using an external corpus may introduce noise and data matching between two different data sources can complicate simple problems. On the other hand, the original corpus and external corpus may have large differences in feature space, which is not comparable in feature representation, and it is difficult to judge algorithmic effects with l2 paradigm. Based on the above considerations, this paper uses the original corpus as the object of pre-clustering to maintain simplicity and effectiveness. It is used as an improvement strategy to combine the topic knowledge from classical short-text clustering algorithms with semantic embeddings, but this also destroys the original semantic space and leads to inconsistency. The feature matrix being too large will also lead to the problem of large computation and long model fitting time, so the dimensionality reduction process is necessary before fusion to reduce the noise in data and prevent overfitting. The specific steps are as follows:

1) Get topic vectors by a short-text clustering algorithm, and reduce dimension;

2) Get semantic embeddings by a language model, and reduce dimension;

3) Splice embeddings, and compute similarity using cosine.

Taking vectors incorporating pre-clustered knowledge as the final representation of each document, the l2 paradigm of vectors under the same topic is reduced, and the spatial distribution is denser which alleviates anisotropy. In this way, the cosine similarity measure is more reasonable, which can improve the accuracy of the clustering results.

3.3. G-BERTopic Algorithm

In brief, get topic vectors by a classical short-text clustering algorithm, and splice them and semantic embeddings in the cluster remapping phase of BERTopic, so that short texts on the same topic are more likely to be assigned into the same cluster. Figure 1 is the overall structure of BERTopic with pre-clustered knowledge.

When it is desired to cluster documents into class K, the process is shown below:

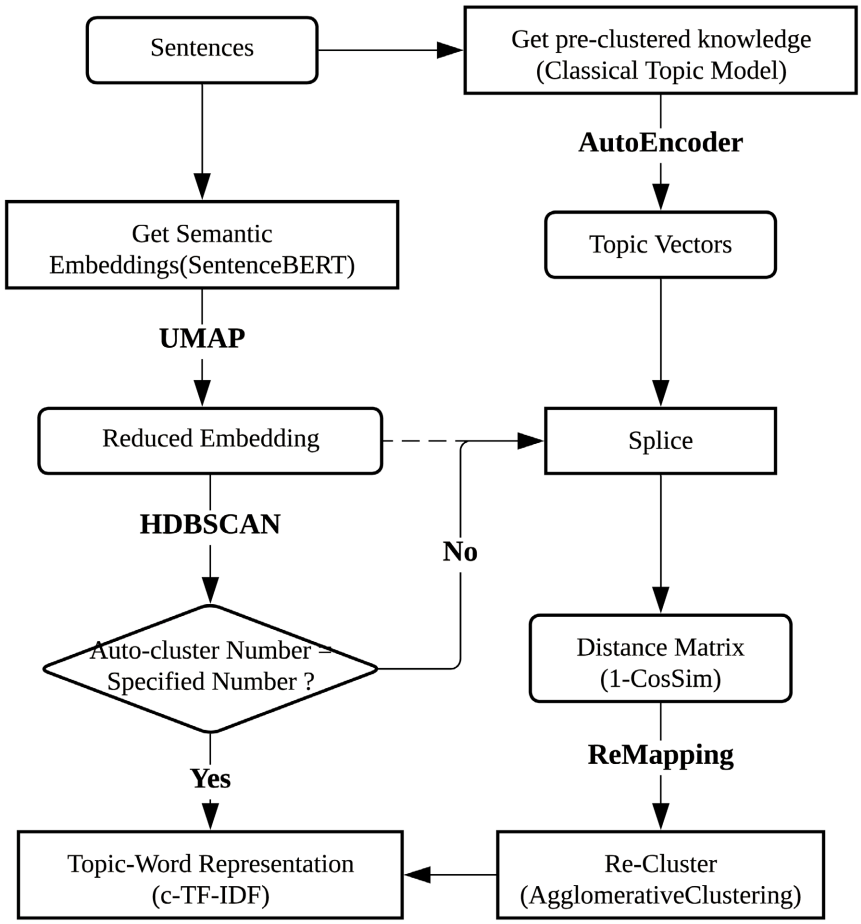


Figure 1. Algorithm flow chart.

1) Get Embeddings

Use a classical short-text topic model to obtain the vector matrix of the topic[1] and denote it as $g \in R^{D\times K}$. The topic vectors are generally sparse, so to minimize redundancy, use an auto-encoder to map it to lower dimensionality, and denote it as $\boldsymbol{g}^{\boldsymbol{r}} \in R^{D\times(K/2)}$:

$$\boldsymbol{g}^{\boldsymbol{r}} = \text{AutoEncoder}(\boldsymbol{g}) \tag{4}$$

Sentences are fed into a pre-trained model SentenceBERT to generate semantic

[1]The pre-clustering scheme used in this paper is GSDMM: https://github.com/rwalk/gsdmm.

embeddings $\boldsymbol{e}$. Use UMAP to reduce dimension, preserving the global structure while retaining local semantic features, and denote it as $\boldsymbol{e}^r \in R^{D\times 128}$:

$$\boldsymbol{e} = \text{SentenceBERT}\left(\vec{d}\right) \tag{5}$$

$$e^r = UMAP(e) \tag{6}$$

2) Cluster

The optimal clustering results are automatically generated using the HDBSCAN algorithm on semantic embeddings $\boldsymbol{e}^r$, where $\boldsymbol{T} \in R^{D\times 1}$ denotes corresponding labels after automatic clustering.

$$T = \text{HDBSCAN}\left(e^r\right) \tag{7}$$

When $|T| > K$, clusters mapping is required. Splice dimensionality reduced semantic embeddings with topic vectors as $\boldsymbol{e}'$, and takes the mean value under the same topic as the representation $\boldsymbol{t_k}$ for each topic:

$$e' = \text{concat}\left(e^r, g^r\right) \tag{8}$$

$$e^r = \text{UMAP}(e) \tag{9}$$

The distance matrix is computed through cosine similarity among topic representations and fed into *Agglomerative Clustering* algorithm for further clustering, to get new cluster labels $\boldsymbol{T}'$:

$$\text{distance} = 1 - \text{sim}_{\cos}(t) \tag{10}$$

$$e^r = \text{UMAP}(e) \tag{11}$$

where the diagonal of distance is filled using zeros. Mapping the labels yields the final clustering result $\boldsymbol{Z}$. Update the representation of each topic.

3) Topic Words

Generalize the classical TF-IDF as c-TF-IDF for the clustering granularity. It treats sentences under the same cluster as a whole and extracts important words to construct the topic-word representation. The importance score is shown in the following equation:

$$W_{w,z} = tf_{w,z} \cdot \log\left(1 + \frac{A}{tf_w}\right) \tag{12}$$

where $tf_{w,z}$ denotes the co-occurrence frequency of word w in topic z, and A denotes the average number of words across all topics.

4. Experiment

In this section, we will launch experiments on the StackOverflow dataset. GSDMM is chosen as the pre-clustered algorithm, and the improved model G- BERTopic is compared with baselines on metrics such as Macro-F1, topic coherence, DBI, etc. Experimental results demonstrate the superiority of BERTopic introducing pre-clustered knowledge on Macro-F1.

4.1. Basic Settings

Dataset: StackOverflow is a well-known community for programmers to learn and share. The dataset collects post-question titles on different topics. For each topic, 100 pieces of data are extracted, satisfying lengths of 5 to 40. After conversion to lowercase, stemming extraction and lexical reduction, the basic attributes are shown in Table 1.

Baseline: To evaluate the effectiveness of G-BERTopic, the following baselines are selected for comparison in this paper.

- **BERTopic**: A topic model based on the BERT pre-trained model, using SentenceBERT for clustering and c-TF-IDF for topic representation.
- **BTM**: One of the classic topic models for short text. It is based on bi-term co-occurrence patterns and has stable performance. The number of iterations is set to 50, alpha to 0.1, and beta to 0.01.
- **LDA**: The most widely used topic model. Assuming that document topics are in the form of probability distributions, the Dirichlet distribution is used as the prior distribution of model parameters, and topics are sampled using Gibbs Sampling. The number of iterations is 50.

Table 1. Dataset description.

Number of Topics	Total Number	Max Length	Min Length	Avg Length
20	2000	16	3	6

4.2. Evaluation Metrics

The model proposed in this paper consists of two parts: cluster generation and topic representation, so the evaluation metrics are selected for these two aspects. For cluster quality, it includes internal and external evaluation metrics, where internal quality evaluates the cluster structure and external quality relies on the accuracy of topic assignment accuracy [34]. Topic representation is evaluated by choosing semantic coherence in this paper.

In this paper, the following evaluation metrics were chosen and the results are the mean values of 10 experiments.

Macro-F1 is an external evaluation metric for datasets that are not affected by data imbalance. We take the same number of samples for each category of the StackOverflow dataset. It is defined as:

$$\mathrm{F1}_{\mathrm{macro}} = 2 \cdot \frac{\mathrm{Precision}_{\mathrm{macro}} \cdot \mathrm{Recall}_{\mathrm{macro}}}{\mathrm{Precision}_{\mathrm{macro}} + \mathrm{Recall}_{\mathrm{macro}}} \tag{12}$$

where $\mathrm{Precision}_{\mathrm{macro}}$ and $\mathrm{Recall}_{\mathrm{macro}}$ are the average values of $\mathrm{Precision}_i$ and Recall_i for each category. $\mathrm{F1}_{\mathrm{macro}}$ takes the value in the range of $[0,1]$. The higher the score, the better the clustering quality.

NMI(Normalized Mutual Information) is a normalization of MI used to measure the degree of consistency of two clusters, and belongs to external evaluation metrics. It is defined as:

$$\mathrm{NMI}(X,Y)=\frac{\mathrm{MI}(X,Y)}{F(H(X),H(Y))} \tag{13}$$

where $X=\{x_1,x_2,\cdots,x_k\}$ is assigned topics, $Y=\{y_1,y_2,\cdots,y_k\}$ is true topics, and $\mathrm{MI}(X,Y)$ is the mutual information of X and Y, reflecting the degree of correlation of the two. $H(\cdot)$ the entrop and $F(\cdot)$ is a normalization function. NMI takes values in the range of $[0,1]$. The higher the score, the better the clustering quality.

DB (Davies-Bouldin) is an internal evaluation index, which measures differences between clusters and closeness within clusters. It is defined as:

$$\mathrm{DB}(k)=\frac{\sum_{i=1}^{k}\max_{j.j\neq i}\left\{\frac{S_i+S_j}{d_{ij}}\right\}}{k} \tag{14}$$

where $S_i=(1/n)\sum_{x\in C_i}\|x-z_i\|$ measures tightness in cluster C_i, $d_{ij}=\|z_i-z_j\|$ measures the dispersion between C_i and C_j. n_i denotes the number of samples in C_i, z_i is the center of mass of C_i, and $\|\cdots\|$ denotes Euclidean distance. The smaller the value, the better the quality of clustering.

Coherence measures the coherence of the topic. In this paper, we use the idea proposed by Röder *et al.* [35], which utilizes the topic representation and reference corpus to determine the contextual features of each topic word. The higher the value, the better the topic modeling results.

4.3. Results

The comparison results are shown in Table 2. Note that the external evaluation metrics need to compare the original topics with the predicted topics, so we use the Hungarian Algorithm for matching. Experiments show that G-BERTopic has superior metrics on the StackOverflow dataset, with Macro-F1 of 0.661, which is 0.16 better than the original BERTopic and maintains the best or second best in other metrics. The results show that the incorporation of pre-clustered topic knowledge improves accuracy while ensuring semantic coherence, proving effectiveness of G-BERTopic.

Table 2. Comparison of indicators.

	Macro-F1	NMI	DBI	Coherence
LDA	0.126	0.084	12.063	**0.377**
BTM	0.231	0.235	8.060	0.301
BERTopic	<u>0.496</u>	<u>0.616</u>	**3.167**	0.344
G-BERTopic	**0.661**	**0.662**	<u>3.263</u>	<u>0.365</u>

Experiments were conducted by incorporating pre-clustered topic knowledge at different phases of BERTopic, and the results are shown in Table 3. The result shows that the highest accuracy rate is after clustering, which justifies the

algorithm, that is, the first clustering phase should not be destructed. The dimension reduction phase is a key step that affects the clustering results by performing a low-dimensional dense vectorized representation of documents. Therefore, adding topic information in the first two cases, whether before dimension reduction or after dimension reduction before clustering, affects the intuition of clustering based on semantic similarity, *i.e.*, adding other features instead blurs the semantic understanding. The purpose of generating topic vectors after the first clustering is to obtain a comprehensive representation, so splicing topic-related vectors from GSDMM in the second clustering phase can add more effective knowledge to improve the accuracy of clustering.

Since word embeddings and sentence embeddings share the same high-dimensional space and short texts are characterized by more low-frequency words, word embeddings are used here instead of sentence embeddings to verify whether the addition of pre-clustered knowledge improves the anisotropy. In this paper, l2 norm of different word frequencies is calculated to compare the distance from the origin before and after the improvement, as shown in Table 4. From the results, it can be seen that word frequency brings bias to the semantic space, because low-frequency words are distributed farther than high-frequency words. After splicing the topic knowledge, l2 norms are all reduced to different degrees, indicating that this approach makes the vector distribution more tightly, which alleviates that bias to some extent.

Table 3. The effect of incorporating pre-clustered knowledge at different stages.

Splice Position	Macro-F1	NMI
Before dimension reduction	0.515	0.545
After dimension reduction before clustering	0.626	0.651
After clustering	**0.659**	**0.662**

Table 4. SSE of different word frequency groups.

Word Frequency	(0, 100)	[100, 500)	[500, 1000)	[1000, 2790)
$l2\text{-norm}_{SBERT}$	4.53	9.07	10.14	19.18
$l2\text{-norm}_{SBERT+GSDMM}$	4.21	8.41	9.40	17.79
$3\text{-NN inertia}_{SBERT}$	1.05	3.92	5.66	25.09
$3\text{-NN inertia}_{SBERT+GSDMM}$	1.03	3.85	5.54	24.36
diff	0.02	0.07	0.12	0.73

For different word frequencies, different components are grouped into 3 clusters by k-means, and SSE is calculated as the degree of closeness in the vector space. From the results, high-frequency words are denser than low-frequency words. So, the sparse distribution of low-frequency words will lead to some gaps in the semantic space, making the similarity of embeddings not accurately

represent the semantics of sentences. The method in this paper reduces the sum of squares of the distances of word embeddings to cluster centroids in different degrees, and the lower the frequency, the better the effect. This proves that the method proposed in this paper can alleviate this sparsity and make the vector representation semantically smoother, which in turn improves the accuracy of clustering.

In conclusion, the experiment proves that adding pre-clustered knowledge in the process of secondary clustering can improve the clustering accuracy of BERTopic.

4.4. Discussion

BERTopic separates clustering and topic representation, where the language model can be replaced with the state-of-the-art. Similarly, G-BERTopic can combine BERTopic with different classical topic models. The results in Table 5 show that Macro-F1 is significantly improved after the combination compared to the native algorithm, and it remains comparable in NMI and Coherence. For the internal metric DBI, it is worse because the addition of features affects semantic space, but it has been shown in Section 4.3 that adding new features helps mitigate anisotropy and can improve clustering accuracy, so such a loss is worthwhile. It proves that incorporating pre-clustered knowledge in BERTopic can improve the clustering effect. The reason for choosing GSDMM is that the two are more consistent in their assumptions, *i.e.*, they both assume that a document contains only one topic. In addition, due to portability, the effect of topic assignment can be further improved by replacing the self-aggregating topic model with a more advanced one.

Table 5. Comparison of BERTopic combined with different pre-clustered models.

Model	Macro-F1	NMI	DBI	Coherence
BERTopic	0.496	0.616	**3.167**	0.344
BERTopic$_{+GSDMM}$	0.661	**0.662**	3.263	**0.365**
BERTopic$_{+BTM}$	**0.666**	0.659	3.250	**0.365**
BERTopic$_{+LDA}$	0.663	0.661	3.221	0.360

5. Conclusions

To alleviate the significant anisotropy in BERTopic, this paper proposes the G-BERTopic, which adds topic features to semantic embeddings from pre-clustered knowledge. It improves the topic assignment process to make the vector space denser, by reducing the l2 norm of low-frequency words. Through comparative experiments on the StackOverflow dataset, Macro-F1 of G-BERTopic is improved by 1.5%, which outperforms other baselines, demonstrating its effectiveness in extracting topic information. Based on the algorithm in this paper, valuable information can be extracted from massive short texts for opinion mining, social media

monitoring, public opinion management, etc. It can efficiently identify core viewpoints and emotional tendencies, such as comments, and timely understand the direction of public emotions, which can help enterprises and the government to understand feedback and adjust their strategies on time.

The method proposed in this paper can effectively improve clustering quality. However, this model still has limitations. For example, the internal metrics are not large compared with the original BERTopic. In addition, the introduction of external knowledge inevitably destroys the original semantic space and increases complexity, which can be further improved by expression consistency, or adopting graph alignment. Or, introduce more specialized domain knowledge to explore whether the effect will be improved in domain-specific topic modeling.

Acknowledgements

The authors thank the anonymous reviewers for their helpful comments.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Li, X., Mao, J., Ma, W., Liu, Y., Zhang, M., Ma, S., *et al.* (2021) Topic-Enhanced Knowledge-Aware Retrieval Model for Diverse Relevance Estimation. *Proceedings of the Web Conference* 2021, Ljubljana, 19-23 April 2021, 756-767. https://doi.org/10.1145/3442381.3449943

[2] Adlakha, V., Dhuliawala, S., Suleman, K., de Vries, H. and Reddy, S. (2022) TopiOCQA: Open-Domain Conversational Question Answering with Topic Switching. *Transactions of the Association for Computational Linguistics*, **10**, 468-483. https://doi.org/10.1162/tacl_a_00471

[3] Sejwal, V.K. and Abulaish, M. (2022) A Hybrid Recommendation Technique Using Topic Embedding for Rating Prediction and to Handle Cold-Start Problem. *Expert Systems with Applications*, **209**, Article 118307. https://doi.org/10.1016/j.eswa.2022.118307

[4] Cao, B., Xiao, Q., Zhang, X. and Liu, J. (2019) An API Service Recommendation Method via Combining Self-Organization Map-Based Functionality Clustering and Deep Factorization Machine-Based Quality Prediction. *Chinese Journal of Computers*, **42**, 1367-1383.

[5] Xi, X., Guo, Y., Song, X. and Wang, J. (2021) Research on the Technical Similarity Visualization Based on Word2vec and LDA Topic Model. *Journal of the China Society for Scientific and Technical Information*, **40**, 974-983.

[6] Ruan, G. and Huang, Y. (2023) Topic Recognition of Comment Text Based on Sense Bert and LDA. *Journal of Modern Information*, **43**, 46-53.

[7] Agarwal, N., Sikka, G. and Awasthi, L.K. (2023) WGSDMM+GA: A Genetic Algorithm-Based Service Clustering Methodology Assimilating Dirichlet Multinomial Mixture Model with Word Embedding. *Future Generation Computer Systems*, **145**, 254-266. https://doi.org/10.1016/j.future.2023.03.028

[8] Peinelt, N., Nguyen, D. and Liakata, M. (2020) tBERT: Topic Models and BERT Joining Forces for Semantic Similarity Detection. *Proceedings of the* 58*th Annual Meeting of*

the Association for Computational Linguistics, Online, 5-10 July 2020, 7047-7055. https://doi.org/10.18653/v1/2020.acl-main.630

[9] Salton, G., Wong, A. and Yang, C.S. (1975) A Vector Space Model for Automatic Indexing. *Communications of the ACM*, **18**, 613-620. https://doi.org/10.1145/361219.361220

[10] Hofmann, T. (1999) Probabilistic Latent Semantic Analysis. *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, Stockholm, 30 July-1 August 1999, 289-296.

[11] Blei, D.M., Ng, A. and Jordan, M.I. (2003) Latent Dirichlet Allocation. *Journal of Machine Learning Research*, **3**, 993-1022.

[12] Yu, H., Fan, S., Wu, L. and Ma, Z. (2022) Review Inquiry and IPO Information Disclosure under the Registration System of Science and Technology Board—Text Analysis Based on LDA Subject Model. *Journal of Management Sciences in China*, **25**, 45-62. https://doi.org/10.19920/j.cnki.jmsc.2022.08.003

[13] Feng, J., Rao, Y., Xie, H., Wang, F.L. and Li, Q. (2019) User Group Based Emotion Detection and Topic Discovery over Short Text. *World Wide Web*, **23**, 1553-1587. https://doi.org/10.1007/s11280-019-00760-3

[14] Hong, L. and Davison, B.D. (2010) Empirical Study of Topic Modeling in Twitter. *Proceedings of the First Workshop on Social Media Analytics*, Washington DC, 25-28 July 2010, 80-88. https://doi.org/10.1145/1964858.1964870

[15] Diao, Q., Jiang, J., Zhu, F. and Lim, E.P. (2012) Finding Bursty Topics from Microblogs. *Proceedings of the* 50*th Annual Meeting of the Association for Computational Linguistics*, Jeju Island, 8-14 July 2012, 536-544.

[16] Zhang, Q., Gong, Y., Sun, X. and Huang, X. (2014) Time-Aware Personalized Hashtag Recommendation on Social Media. *Proceedings of COLING* 2014, *the* 25*th International Conference on Computational Linguistics: Technical Papers*, Dublin, 23-29 August 2014, 203-212.

[17] Nguyen, H.T., Duong, P.H. and Cambria, E. (2019) Learning Short-Text Semantic Similarity with Word Embeddings and External Knowledge Sources. *Knowledge-Based Systems*, **182**, Article 104842. https://doi.org/10.1016/j.knosys.2019.07.013

[18] Jin, O., Liu, N.N., Zhao, K., Yu, Y. and Yang, Q. (2011) Transferring Topical Knowledge from Auxiliary Long Texts for Short Text Clustering. *Proceedings of the* 20*th ACM International Conference on Information and Knowledge Management*, Glasgow, 24-28 October 2011, 775-784. https://doi.org/10.1145/2063576.2063689

[19] Yan, X., Guo, J., Lan, Y. and Cheng, X. (2013) A Biterm Topic Model for Short Texts. *Proceedings of the* 22*nd international conference on World Wide Web*, Rio de Janeiro, 13-17 May 2013, 1445-1456. https://doi.org/10.1145/2488388.2488514

[20] Yang, D., Li, N., Zou, L. and Ma, H. (2022) Lexical Semantics Enhanced Neural Word Embeddings. *Knowledge-Based Systems*, **252**, Article 109298. https://doi.org/10.1016/j.knosys.2022.109298

[21] Fan, H. and Li, P. (2021) Sentiment Analysis of Short Text Based on FastText Word Vector and Bidirectional GRU Recurrent Neural Network. *Information Science*, **39**, 15-22. https://doi.org/10.13833/j.issn.1007-7634.2021.04.003

[22] Rashid, J., Kim, J., Hussain, A. and Naseem, U. (2023) WETM: A Word Embedding-Based Topic Model with Modified Collapsed Gibbs Sampling for Short Text. *Pattern Recognition Letters*, **172**, 158-164. https://doi.org/10.1016/j.patrec.2023.06.007

[23] Qin, T., Liu, Z. and Chen, K. (2020) Topic Model Combining Topic Word Embedding and Attention Mechanism. *Computer Engineering*, **46**, 104-108.

https://doi.org/10.19678/j.issn.1000-3428.0055952

[24] Jia, J., Wang, H., Ren, K. and Kang, W. (2022) Research on Text Clustering Based on Sentence Vector and Convolutional Neural Network. *Computer Engineering and Applications*, **58**, 123-128.

[25] Zhao, X., Wang, D., Zhao, Z., Liu, W., Lu, C. and Zhuang, F. (2021) A Neural Topic Model with Word Vectors and Entity Vectors for Short Texts. *Information Processing & Management*, **58**, Article 102455. https://doi.org/10.1016/j.ipm.2020.102455

[26] Qiang, J., Chen, P., Wang, T. and Wu, X. (2017) Topic Modeling over Short Texts by Incorporating Word Embeddings. *Advances in Knowledge Discovery and Data Mining*, Jeju, 23-26 May 2017, 363-374. https://doi.org/10.1007/978-3-319-57529-2_29

[27] Bianchi, F., Terragni, S. and Hovy, D. (2021) Pre-Training Is a Hot Topic: Contextualized Document Embeddings Improve Topic Coherence. *Proceedings of the* 59*th Annual Meeting of the Association for Computational Linguistics and the* 11*th International Joint Conference on Natural Language Processing* (*Volume* 2: *Short Papers*), Online, 1-6 August 2021, 759-766. https://doi.org/10.18653/v1/2021.acl-short.96

[28] Grootendorst, M. (2022) BERTopic: Neural Topic Modeling with a Class-Based TF-IDF Procedure. arXiv:2203.05794.

[29] Reimers, N. and Gurevych, I. (2019) Sentence-BERT: Sentence Embeddings Using Siamese BERT-Networks. *Proceedings of the* 2019 *Conference on Empirical Methods in Natural Language Processing and the* 9*th International Joint Conference on Natural Language Processing* (*EMNLP-IJCNLP*), Hong Kong, 3-7 November 2019, 3982-3992. https://doi.org/10.18653/v1/d19-1410

[30] McInnes, L., Healy, J., Saul, N. and Großberger, L. (2018) UMAP: Uniform Manifold Approximation and Projection. *Journal of Open Source Software*, **3**, Article 861. https://doi.org/10.21105/joss.00861

[31] McInnes, L., Healy, J. and Astels, S. (2017) Hdbscan: Hierarchical Density Based Clustering. *The Journal of Open Source Software*, **2**, Article 205. https://doi.org/10.21105/joss.00205

[32] Liu, H. and Wang, Y. (2022) Clustering Short Text Classification Based on Fusion of BERT and GSDMM. *Computer Systems & Applications*, **31**, 267-272. https://doi.org/10.15888/j.cnki.csa.008307

[33] Li, B., Zhou, H., He, J., Wang, M., Yang, Y. and Li, L. (2020) On the Sentence Embeddings from Pre-Trained Language Models. *Proceedings of the* 2020 *Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), Online, 16-20 November 2020, 9119-9130. https://doi.org/10.18653/v1/2020.emnlp-main.733

[34] Hu, Q., Shen, J., Wang, K., Du, J. and Du, Y. (2022) A Web Service Clustering Method Based on Topic Enhanced Gibbs Sampling Algorithm for the Dirichlet Multinomial Mixture Model and Service Collaboration Graph. *Information Sciences*, **586**, 239-260. https://doi.org/10.1016/j.ins.2021.11.087

[35] Röder, M., Both, A. and Hinneburg, A. (2015) Exploring the Space of Topic Coherence Measures. *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, Shanghai, 2-6 February 2015, 399-408. https://doi.org/10.1145/2684822.2685324