LLM-Guided Semantic-Aware Clustering for Topic Modeling

Jianghan Liu[1*], **Ziyu Shang**[2*], **Wenjun Ke**[2,3†], **Peng Wang**[2,3],
Zhizhao Luo[4], **Jiajun Liu**[2], **Guozheng Li**[2], **Yining Li**[1],
[1]College of Software Engineering, Southeast University
[2]School of Computer Science and Engineering, Southeast University
[3]Key Laboratory of New Generation Artificial Intelligence Technology and Its
Interdisciplinary Applications (Southeast University), Ministry of Education, China
[4]Beijing Institute of Technology, Zhuhai
{liujianghan,ziyus1999,kewenjun,pwang,jiajliu,gzli,liyining}@seu.edu.cn
zzluo@bit.edu.cn

Abstract

Topic modeling aims to discover the distribution of topics within a corpus. The advanced comprehension and generative capabilities of large language models (LLMs) have introduced new avenues for topic modeling, particularly by prompting LLMs to generate topics and refine them by merging similar ones. However, this approach necessitates that LLMs generate topics with consistent granularity, thus relying on the exceptional instruction-following capabilities of closed-source LLMs (such as GPT-4) or requiring additional training. Moreover, merging based only on topic words and neglecting the fine-grained semantics within documents might fail to fully uncover the underlying topic structure. In this work, we propose a semi-supervised topic modeling method, LiSA, that combines LLMs with clustering to improve topic generation and distribution. Specifically, we begin with prompting LLMs to generate a candidate topic word for each document, thereby constructing a topic-level semantic space. To further utilize the mutual complementarity between them, we first cluster documents and candidate topic words, and then establish a mapping from document to topic in the LLM-guided assignment stage. Subsequently, we introduce a collaborative enhancement strategy to align the two semantic spaces and establish a better topic distribution. Experimental results demonstrate that LiSA outperforms state-of-the-art methods that utilize GPT-4 on topic alignment, and exhibits competitive performance compared to Neural Topic Models on topic quality. The codes are available at https://github.com/ljh986/LiSA.

1 Introduction

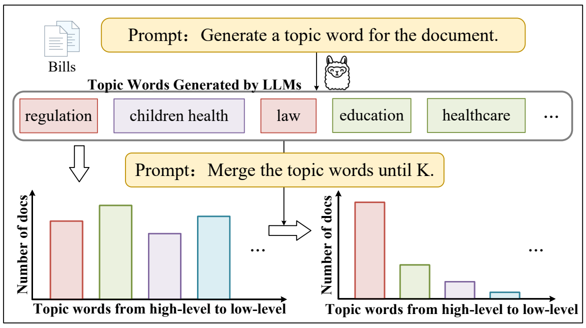

Bills
Prompt：Generate a topic word for the document.
Topic Words Generated by LLMs
regulation
children health
law
education
healthcare
···
Prompt：Merge the topic words until K.
Number of docs
···
Topic words from high-level to low-level
Number of docs
···
Topic words from high-level to low-level

Figure 1: Cluster imbalance issues caused by topic words of various granularities on bills from U.S. congresses. Red color denotes topic words that are too coarse-grained.

Topic modeling is a key technique in text analysis, aiming at uncovering underlying themes or topics within a collection of documents, allowing for the automatic categorization of text based on these themes (Abdelrazek et al., 2023). Conventional topic models, such as Latent Dirichlet Allocation (LDA) (Blei et al., 2003), model the topic-word distribution and the document-topic distribution in an unsupervised manner. BERTopic (Grootendorst, 2022), a cluster-based topic model, clusters document embeddings derived from pre-trained models on a large corpus, such as BERT variants (Devlin et al., 2019), to identify topics. However, these methods represent topics as word combinations and often require domain expert annotations for users to interpret them directly.

To address the above shortcuts, several topic models based on large language models (LLMs) have emerged, such as TopicGPT (Pham et al., 2024) and PromptTopic (Wang et al., 2023). These methods first rely on the language understanding and generation capabilities of large language models (LLMs) (Radford et al.) to achieve comprehensive semantic analysis of documents and generate semantically consistent topic words. Then, they simulate the clustering process by instructing LLMs to classify a large number of topic words into fewer themes, thereby refining and merging the topics. However, due to the limited instruction-following ability of LLMs, the consistency in the

[*] Equal contribution.
[†] Corresponding author.

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 18420–18435
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

granularity of all topics during the topic generation phase could not be guaranteed (Mu et al., 2024a), as represented by the different colored blocks for various levels of topics in Figure 1. Consequently, LLMs are inclined to incorporate lower-level topics into a few high-level topics, resulting in **obvious long tail distribution after the topic refining phase**, where the topics are too coarse-grained to mine latent information in the document or provide meaningful topics for subsequent applications. Taking an example of PromptTopic (Wang et al., 2023) as shown in Figure 1, when dealing with Bills dataset, it tends to refine most topic words to the *law* or *regulation* theme, resulting in most documents concentrated in high-level topics. Moreover, relying on LLMs to merge similar topic words from a large set introduces significant randomness (the order of topic words in the prompts heavily influences performance (Saito et al., 2025)), further undermining the robustness of these methods. As for TopicGPT, it directly instructs GPT-4 (Achiam et al., 2023) to generate topics of consistent granularity with seed topics, alleviating long-tail distribution during the refinement stage. However, it may require substantial resources and rely heavily on seed topics provided by humans.

Additionally, since each topic word is derived from summarizing the topic-level information of the documents, these words often lose the semantic details contained in the original documents. Consequently, refining topic words only based on their intrinsic semantic information **fails to fully uncover the underlying topic structure** within the document. Moreover, when overly coarse-grained topic words are present, directly assigning documents to the cluster corresponding to its topic words, which are formed during refinement, could **lead to inaccurate cluster assignments for documents**. As present in Figure 1, if LLMs generate a broad topic word *law* for a document with main content about *education*, this document would be assigned to the wrong cluster that is semantically related to *law*.

To address the above issues, we propose LiSA, a novel topic modeling method that collaborates LLMs and clustering to obtain topics, and further integrates neighboring semantic information from topic-level and document-level to achieve a better topic distribution. To fully mine the topics under the corpus, we first construct a topic-level semantic space by iteratively instructing LLMs to generate candidate topic words along with descriptions. Then, we perform clustering on both candidate topic words and documents to identify inconsistencies between the semantic space of document-level and topic-level, as well as reduce the randomness of LLMs in merging candidate topic words. Subsequently, recognizing the complementarity between the two semantic spaces, we establish a mapping between them by assigning a sample from topic-level to document-level and utilize LLMs to aid in checking the low-confidence points, thereby initiating the alignment between the two semantic spaces from a localized, single-point perspective. Furthermore, we design a collaborative enhancement strategy that trains two topic prediction networks with a sophisticated loss function to achieve alignment from a global perspective, as well as obtain the topic distribution of documents.

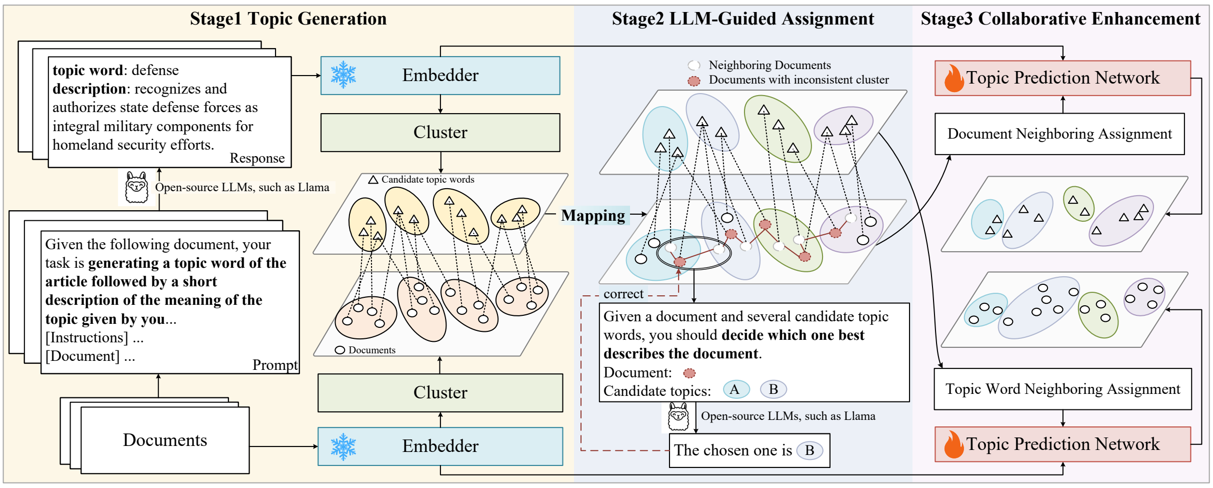
Our key contributions are as follows:

- We propose LiSA, a novel topic model that utilizes LLMs to construct a topic-level semantic space, which provides strong guidance in modeling the topic distribution of documents.
- LiSA identify inconsistencies between the topic-level and document-level semantic spaces and then align them from a local to global perspective, thereby reducing the impact of topic words with inconsistent granularities.
- We conduct extensive experiments and demonstrate that our method outperforms state-of-the-art method that utilizes GPT-4 in clustering performance, *i.e.*, an average of 1.5%, 6.0%, 2.0% improvement on Bills and Wiki dataset for P_1, ARI, and NMI, respectively.

2 Related work

Traditional Topic Models and Neural Topic Models To mine the topics within the corpus, conventional topic models (Steyvers and Griffiths, 2007; Larochelle and Lauly, 2012; Shi et al., 2018), such as LDA (Blei et al., 2003), model the topic-word distribution and the document-topic distribution in an unsupervised manner. Neural Topic Models directly optimize parameters without requiring model-specific derivations, achieving better scalability and flexibility.

Cluster-based Topic Models Sia et al. propose to cluster pre-trained word embeddings and rerank top words, which is straightforward to implement, and feasible for regular-length documents.



Stage1 Topic Generation
Stage2 LLM-Guided Assignment
Stage3 Collaborative Enhancement
Response
Open-source LLMs, such as Llama
Prompt
Documents
Embedder
Cluster
Candidate topic words
Documents
Cluster
Embedder
Mapping
Neighboring Documents
Documents with inconsistent cluster
correct
Open-source LLMs, such as Llama
Topic Prediction Network
Document Neighboring Assignment
Topic Word Neighboring Assignment
Topic Prediction Network

topic word: defense
description: recognizes and authorizes state defense forces as integral military components for homeland security efforts.

Given the following document, your task is **generating a topic word of the article followed by a short description of the meaning of the topic given by you**...
[Instructions] ...
[Document] ...

Given a document and several candidate topic words, you should **decide which one best describes the document**.
Document: ●
Candidate topics: (A) (B)

The chosen one is (B)

Figure 2: An overview of our proposed LiSA.

BERTopic (Grootendorst, 2022) follows this way and utilizes c-TF-IDF to obtain topic words. Subsequently, some methods directly leverage LLMs to generate topic words for documents (Wang et al., 2023; Mu et al., 2024b), and PromptTopic (Wang et al., 2023) simulates clustering by merging these topic words to derive the topic distribution of documents. As for topic-word distribution, it follows BERTopic to generate the bag-of-words representation. Similarly, TopicGPT (Pham et al., 2024) leverages GPT-4 to generate and merge topic words, directly producing descriptions and bypassing the cumbersome process of creating a topic-word distribution followed by summarizing top words. Recently, Mu et al. challenge the issues of topic granularity and hallucination in the above work. They address these problems by fine-tuning LLMs using Direct Preference Optimization (Rafailov et al., 2024), enabling the generation of topic words with consistent granularity. However, they concentrate on fine-tuning LLMs to consistently generate granular topic words but overlook the topic merging stage, failing to produce the topic distribution of documents. In this work, we follow cluster-based methods and explore how to obtain the topic distribution using the topic words generated by open-source LLMs, *i.e.*, Llama-3 (Meta, 2024) and Mistral-2 (Jiang et al., 2023), without fine-tuning.

3 Methodology

Figure 2 illustrates LiSA's overall architecture. In the first stage, an LLM generates a candidate topic word along with a description for each document. Subsequently, the candidate topics and documents are separately organized into K clusters. Since each document has a unique candidate topic, this stage assigns two categories to each document. The first category reflects the fine-grained semantics of the document itself, while the other is the topic-level semantics of the document. In the second stage, we introduce an LLM-guided assignment strategy designed to first establish a mapping from document-level to topic-level and then utilize LLMs to aid in checking the low-confidence points. In the third stage, we propose a collaborative enhancement strategy that leverages neighboring information to improve the alignment between topic-level and document-level semantic spaces, resulting in a more optimal topic distribution.

3.1 Problem Statement and Notations

Consider a collection of documents $D = \{d_1, \cdots, d_{|D|}\}$, the conventional methods (Blei et al., 2003) model the word distribution for topics as well as the topic distribution for documents. The former is to represent the topic with words or some other observations, and the latter is to indicate the probabilities of a document belonging to each topic. Unlike conventional topic models, LLMs can generate semantically coherent topic words, allowing for the replacement of word distribution with natural language (Pham et al., 2024; Wang et al., 2023), referred to here as topic representation. Therefore, the objective of this work is to leverage candidate topic words generated by LLMs to establish the topic representation and subsequently determine the topic distribution of documents.

3.2 Topic Generation

In this stage, we prompt an LLM to generate a candidate topic word for each document, accompanied by a description that aids in understanding topic word's meaning. Specifically, for documents $D = \{d_1, d_2, \cdots, d_{|D|}\}$, candidate topic words

$T = \{t_1, t_2, \cdots, t_{|T|}\}$ (where $|T| \leq |D|$) and topic descriptions $TDs = \{td_1, td_2, \cdots, td_{|D|}\}$ are generated. The complete prompt template is shown in Appendix Table D1. In this way, a one-to-one mapping ($\mathcal{M}_1 : D \rightarrow T$) from documents to candidate topic words, as well as a one-to-many mapping ($\mathcal{M}_2 : T \rightarrow \mathcal{S}(TDs)$, where $\mathcal{S}(TDs)$ is the power set of TDs) between candidate topic words and descriptions has been established. By generating candidate topic words and descriptions, we effectively establish a topic-level semantic space. Each document retains its document-level semantics while also being endowed with topic-level semantics. Subsequently, we mine the topic distribution of documents based on the above two semantic spaces.

Previous LLM-based methods (Wang et al., 2023; Pham et al., 2024) often rely on LLMs to merge semantically redundant topic words, but this process tends to be unstable, mainly because LLMs are very sensitive to the positioning of prompts (Saito et al., 2025). Additionally, the topics of documents are prone to a long-tailed distribution, meaning that the majority of documents are assigned to several high-level topics. To enhance the stability of topic merging, we design a cluster-based merging strategy. Specifically, for each candidate topic word $t \in T$, we leverage all relevant descriptions to generate the representation of candidate topic words R_T as follows:

$$R_T(t) = \frac{1}{|\mathcal{M}_2(t)|} \sum_{j=1}^{|\mathcal{M}_2(t)|} embedder(td_j) \quad (1)$$

where $td_j \in \mathcal{M}_2(t)$ represents the j_{th} description of candidate topic word t, and $embedder$ is the embedding model[1]. To obtain document representations R_D, we similarly input each document d into the same $embedder$ as $R_D(d) = embedder(d)$. Then, we cluster both candidate topic representation R_T and document representation R_D into K categories[2]. Let C_D and C_T represent the cluster of documents and candidate topic words, respectively. In this way, semantically redundant candidate topic words are merged into K distinct categories. Let $\mathcal{M}_3 : D \rightarrow C_D$ represents the mapping between D and C_D. Using the categories of candidate topic words as the initial topics, we obtain the topic representation by prompting an LLM to generate a new summarizing topic word ω that can describe as many candidate topic words as possible within each topic cluster from C_T. Finally, a set of summarizing topic words $\Omega = \{\omega_1, \cdots, \omega_K\}$ is composed, thus obtaining the mapping $\mathcal{M}_\Omega : C_T \rightarrow \Omega$.

Algorithm 1 Find Mapping

```
Input: Clustering results $C_D$, $C_T$
Output: Mapping of clusters $\mathcal{M}_4$
Parameters: $U_D$ is unique labels in $C_D$; $U_T$ is unique labels in $C_T$; J is Confusion matrix;
 1: $U_D \leftarrow$ unique($C_D$)
 2: $U_T \leftarrow$ unique($C_T$)
 3: J $\leftarrow$ initialize_matrix(length($U_D$), length($U_T$))
 4: for each $u_d$ in $U_D$ do
 5:     for each $u_t$ in $U_T$ do
 6:         J$[u_d][u_t] \leftarrow$
            count_common_elements$(C_D, u_d, C_T, u_t)$
 7:     end for
 8: end for
 9: $r, c \leftarrow$ hungarian_algorithm$(M)$
10: $\mathcal{M}_4 \leftarrow$ initialize_mapping()
11: for each $(row, col)$ in $(r, c)$ do
12:     $\mathcal{M}_4[U_D[row]] \leftarrow U_T[col]$
13: end for
14: return $\mathcal{M}_4$
```

3.3 LLM-Guided Assignment

Although candidate topic words for documents can assist in identifying initial topics, their accuracy is significantly constrained by the reliance on topic-level semantic information alone. Therefore, we propose an LLM-guided assignment strategy, which integrates both topic-level and document-level semantic information to establish a mapping between two semantic spaces. Broadly, we match a summarizing topic word to the document's clustering category C_D. For each document, if topic-level and document-level summarizing topic words are consistent, the document is assigned to that topic. In cases of inconsistency, an LLM is prompted to select the most appropriate candidate topic.

As illustrated in Algorithm 1, we first construct a confusion matrix $\mathbf{J}$, where $\mathbf{J}[i][j]$ represents the number of common sample in C_D^i and C_T^j and $i, j \in [1, K]$ (Lines 1-8). Then, the Hungarian algorithm (Kuhn, 1955) is utilized to establish the mapping ($\mathcal{M}_4 : C_D \rightarrow C_T$) between C_D and C_T (Lines 9-14). However, there are documents with conflict summarizing topic words, *i.e.*, $D_w = \{d' | \mathcal{M}_1(d') \notin \mathcal{M}_4(\mathcal{M}_3(d')) \text{ and } d' \in D\}$. Therefore, for each $d'_i \in D_w$, we retrieve the nearest λ neighbors from D between their document

[1] In this work, we adopt sentence bert, which can be accessed from https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

[2] In the main experiment, we adopted KMeans, and experiments on various clustering methods are shown in Appendix E.1.

representation:

$$\text{dis}_{i,j} = \|R_D(d'_i) - R_D(d_j)\|_2,\ j \in [1, |D|], d'_i \neq d_j \tag{2}$$

where $\text{dis}_{i,j}$ represents the Euclidean Distance between document d'_i and d_j. Then, we filter λ nearest neighbors $N'_i = \{d_j | \text{dis}_{i,j} \leq \hat{\text{dis}}_i\}$, where

$$\hat{\text{dis}}_i = \text{sort}\{\text{dis}_{i,1}, \cdots, \text{dis}_{i,|D|}\}[\lambda] \tag{3}$$

Subsequently, for each $d'_i \in D_w$ and calculated corresponding neighbor set N'_i, we prompt LLM to choose the most suitable topic words from $\mathcal{M}_\Omega(\mathcal{M}_4((\mathcal{M}_3(N'_i))))$. Based on the response of LLMs, we assign the document to the chosen topic. Detailed prompt is presented in Appendix D.

3.4 Collaborative Enhancement

In the previous stage, for each inconsistent document in D_w, we search for neighboring sample points within the document-level semantic space, disregarding the neighboring information embedded in the topic-level semantic space. Therefore, to further align the two semantics from a global perspective, we propose a collaborative enhancement strategy. Specifically, we first design two topic prediction networks (TPN), *i.e.*, $\mathcal{P_D} : \mathcal{R}_D(d_i) \rightarrow p_{i,*} \in \mathbb{R}^K$ and $\mathcal{P_T} : \mathcal{R}_T(t_i) \rightarrow q_{i,*} \in \mathbb{R}^K$, to predict the soft cluster distribution of a document and a topic word, respectively, which can be formulated as follows:

$$\mathcal{P_D}(D) = \begin{bmatrix} p_{1,1} & \cdots & p_{1,K} \\ \vdots & \ddots & \vdots \\ p_{|D|,1} & \cdots & p_{|D|,K} \end{bmatrix}, \mathcal{P_T}(\mathcal{M}'_1(D)) = \begin{bmatrix} q_{1,1} & \cdots & q_{1,K} \\ \vdots & \ddots & \vdots \\ q_{|D|,1} & \cdots & q_{|D|,K} \end{bmatrix} \tag{4}$$

where p_{ij} and q_{ij} indicates the probability of document d_i and topic word t_i of d_i belonging to j_{th} cluster, respectively.

Then, to make TPN align the clustering of each document d_i with its topic word, we introduce the consistent loss function $\mathcal{L}_1$ defined as:

$$\mathcal{L}_1 = -log \frac{1}{|D|} \sum_{i=1}^{|D|} < p_{i,*}, q_{i,*} > \tag{5}$$

where $< \cdot, \cdot >$ denotes the dot product operator. $\mathcal{L}_1$ encourages the prediction result of TPN on d_i and its corresponding topic word to be consistent. Meanwhile, it guides TPN to produce more distinct prediction results, *i.e.*, both $p_{i,*}$ and $q_{i,*}$ to become one-hot vectors.

Next, consider the following observations: a single document can be associated with multiple topic words, and a single topic word can correspond to multiple documents. Additionally, the fundamental characteristic of clustering is that samples within the same cluster are typically close to each other in the feature space. Based on these considerations, we design a nearest-neighbor matching loss function. First, for each document $d_i \in D$, we search the corresponding μ nearest neighbors $\mathcal{N}_D^i$ by calculating the Euclidean distance between d_i and other documents in the document clustering space R_D. Similarly, for each topic word $M'_1(d_i)$, we also search the corresponding μ nearest neighbors $\mathcal{N}_T^i$ in the topic clustering space R_T. During the training, for each document d_i and its corresponding topic word $M'_1(d_i)$, in each epoch, a nearest neighbor is randomly selected from the neighbor sets $\mathcal{N}_D^i$ and $\mathcal{N}_T^i$, respectively, to construct the neighbor clustering probability matrix. The formal representation is shown as follows:

$$\mathcal{P_D}(\mathcal{N}_D) = \begin{bmatrix} \hat{p}_{1,1} & \cdots & \hat{p}_{1,K} \\ \vdots & \ddots & \vdots \\ \hat{p}_{|D|,1} & \cdots & \hat{p}_{|D|,K} \end{bmatrix}, \mathcal{P_T}(\mathcal{N}_T) = \begin{bmatrix} \hat{q}_{1,1} & \cdots & \hat{q}_{1,K} \\ \vdots & \ddots & \vdots \\ \hat{q}_{|D|,1} & \cdots & \hat{q}_{|D|,k} \end{bmatrix} \tag{6}$$

where i_{th} row of $\mathcal{P_D}(\mathcal{N}_D)$ and $\mathcal{P_T}(\mathcal{N}_T)$ represents the clustering probability of randomly chosen nearest neighbor of d_i and $M'_1(d_i)$.

Subsequently, the nearest-neighbor matching loss function $\mathcal{L}_2$ can be defined as:

$$\mathcal{L}_2 = \mathcal{L}_{TD} + \mathcal{L}_{DT} \tag{7}$$

$$\mathcal{L}_{TD} = -\sum_{i=1}^{K} \log \frac{e^{(<p_{*i}, \hat{q}_{*i}>/\tau)}}{\sum\limits_{j=1}^{K} e^{(<p_{*i}, \hat{q}_{*j}>/\tau)} + \sum\limits_{l \neq i} e^{(<\hat{q}_{*i}, \hat{q}_{*l}>/\tau)}} \tag{8}$$

$$\mathcal{L}_{DT} = -\sum_{i=1}^{K} \log \frac{e^{(<q_{*i}, \hat{p}_{*i}>/\tau)}}{\sum\limits_{j=1}^{K} e^{(<q_{*i}, \hat{p}_{*j}>/\tau)} + \sum\limits_{l \neq i} e^{(<\hat{p}_{*i}, \hat{p}_{*l}>/\tau)}} \tag{9}$$

where τ is the temperature of softmax. The purposes of $\mathcal{L}_2$ are: (1) By enhancing the similarity between a document and the nearest topic words surrounding its candidate topic word, $\mathcal{L}_{TD}$ realizes that a document can be described by multiple topic words, enabling the semantic features of several similar topic words to be perceived by the same document. This also increases the distinction between different clusters in topic word clustering. (2) Similarly, by enhancing the similarity between a candidate topic word and the nearest documents surrounding its documents, $\mathcal{L}_{DT}$ realizes that a topic word can describe multiple different documents, enabling the semantic features of several similar documents to be perceived by the same topic word. Meanwhile, this increases the distinction between different clusters in document clustering. Finally, to prevent TPN from learning the trivial solution that assigns a majority of samples into a minority of clusters, we design a balanced cluster loss function to avoid the issue, which can be defined as follows:

$$\mathcal{L}_3 = -\sum_{i=1}^{K} x_i \log x_i - \sum_{i=1}^{K} y_i \log y_i \tag{10}$$

where

$$x_i = \frac{1}{|D|}\sum_{j=1}^{|D|} p_{ji}, \quad y_i = \frac{1}{|D|}\sum_{j=1}^{|D|} q_{ji}$$

Finally, the overall loss function can be formulated as follows:

$$\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2 - \alpha \times \mathcal{L}_3 \tag{11}$$

where α is a hyperparameter that regulates the weights.

4 Experiments

In this section, we first evaluate LiSA on three widely-used datasets, assessing its performance in terms of topic alignment, topic quality, and through human evaluation. Subsequently, we present ablation studies to investigate the impact of different loss functions on LiSA. Finally, additional ablation experiments on hyperparameters and clustering methods can be found in Appendix B and Appendix E.1, respectively.

4.1 Experimental Setup

4.1.1 Dataset

We select three widely used datasets to evaluate our LiSA, including Bills, Wiki, and Twitter. The Bills dataset is derived from bill summaries from the 110th to 114th U.S. Congresses. Hoyle et al. (2022) collected 32,661 bills and manually annotated each document with topics, which include 21 high-level and 114 low-level labels. Similarly, Hoyle et al. (2022) selected 14,290 "good" articles from Wikipedia (Merity et al., 2018) and manually annotated each document with topics, which consist of 15 high-level and 279 low-level labels. Antypas et al. (2022) collected short tweets from September 2019 to August 2021 and annotated each tweet with a high-level topic label. The Twitter dataset contains 11,171 documents, encompassing 6 high-level labels. The statistical information of the datasets is shown in Appendix A.

4.1.2 Evaluation Metrics

The consistency between quantitative evaluation metrics for topic models and human evaluation has long been a highly debated issue. Hoyle et al. demonstrated that NPMI, a widely used automatic evaluation metric for measuring Topic Coherence, exaggerates differences between models relative to human judgments. To address this issue, Hoyle et al. proposed datasets with human annotations to measure the alignment between topic models and human-labeled topics from the perspective of content analysis. Therefore, we evaluate the performance of LiSA from two aspects: Topic Alignment and Topic Quality, which is also adopted by Wu et al..

Topic Alignment Following previous work (Hoyle et al., 2022; Pham et al., 2024; Wu et al., 2024b), we adopt the harmonic mean of purity (P_1) (Amigó et al., 2009), Adjusted Rand Index (ARI), and Normalized Mutual Information (NMI) to evaluate the performance of clustering. Metrics are detailed in Appendix C.2.

Topic Quality Since our model did not generate the bag-of-words topic representation, we employed c-TF-IDF scores to compute each cluster's most representative words for evaluation (Wang et al., 2023; Grootendorst, 2022). **(1) Topic Coherence:** We employ the widely-used metric, Coherence Value (C_V), which has been empirically shown to outperform the traditional metrics, NPMI, UCI, and UMass (Röder et al., 2015; Wu et al., 2023). **(2) Topic Diversity:** We utilize the Topic Diversity metric (TD, Dieng et al. 2020) to evaluate the differences between discovered topics.

4.1.3 Baselines

In this work, we evaluate our LiSA against traditional topic models (LDA (Blei et al., 2003), NMF (Févotte and Idier, 2011), CTM (Song et al., 2020)), neural topic models (FASTopic (Wu et al., 2024b)) and cluster-based topic models (BERTopic (Grootendorst, 2022), PromptTopic (Wang et al., 2023), TopicGPT (Pham et al., 2024)). We directly adopt the results of EDTM (Dhanania et al., 2024) and TopicGPT, and for the rest, we reproduce their results.

4.2 Main Results

4.2.1 Topic Alignment

The main results of topical alignment with ground truth are shown in Table 1. It can be concluded

Table 1: Experimental results of our LiSA and baselines. The best results are denoted in **bold** and the second-best results are marked by $\underline{\text{underline}}$. It is worth noting that we additionally provide the MI score for the Twitter dataset to achieve a fair comparison to TopicGPT. For all baselines except TopicGPT, we set k=21,15,6 on Bills, Wiki, and Twitter, respectively. The only exception is that we report results on Twitter when $k = 10$ because the K of TopicGPT is the same as the ground truth.

Datasets	Bills			Wiki			Twitter			
	P_1	ARI	NMI	P_1	ARI	NMI	P_1	ARI	MI	NMI
LDA (Blei et al., 2003)	0.52	0.31	0.42	0.72	0.63	$\underline{0.71}$	0.50	0.21	0.34	0.22
BERTopic (Grootendorst, 2022)	0.41	0.18	0.37	0.62	0.50	0.59	0.53	0.18	0.31	0.23
NMF (Févotte and Idier, 2011)	0.24	0.04	0.15	0.58	0.38	0.55	0.29	0.02	0.03	0.02
CTM (Song et al., 2020)	0.37	0.19	0.32	0.61	0.46	0.62	0.45	0.11	0.23	0.14
PromptTopic (Wang et al., 2023)	0.36	0.17	0.28	0.62	0.46	0.59	0.66	0.45	0.59	0.41
EDTM (Dhanania et al., 2024)	$\underline{0.58}$	-	-	0.65	-	-	0.72	-	0.70	-
TopicGPT (GPT-4) (Pham et al., 2024)	0.57	0.40	$\underline{0.49}$	$\underline{0.74}$	0.60	0.70	0.75	-	0.70	-
FASTopic (Wu et al., 2024b)	0.40	0.30	0.33	0.65	0.56	0.60	0.59	0.30	0.53	0.34
	K=24			*K=22*			*K=6*			
LiSA (Llama-3)	**0.59**	0.42	**0.51**	0.73	$\underline{0.66}$	$\underline{0.71}$	**0.78**	**0.65**	$\underline{0.71}$	$\underline{0.46}$
LiSA (Mistral-2)	$\underline{0.58}$	**0.45**	$\underline{0.49}$	**0.75**	**0.67**	**0.72**	$\underline{0.77}$	0.63	**0.72**	**0.47**
	K=21			*K=15*			*K=10*			
LiSA (Llama-3)	0.57	0.40	**0.51**	$\underline{0.74}$	$\underline{0.66}$	$\underline{0.71}$	$\underline{0.77}$	$\underline{0.64}$	0.70	$\underline{0.46}$
LiSA (Mistral-2)	$\underline{0.58}$	$\underline{0.43}$	**0.51**	$\underline{0.74}$	$\underline{0.66}$	**0.72**	0.76	$\underline{0.64}$	0.70	0.45

that collaborating LLMs with clustering methods is most effective, achieving state-of-the-art results and outperforms the GPT-4-based method, TopicGPT (Pham et al., 2024), on all datasets. Additionally, LiSA achieves better topic distribution of documents by incorporating neighboring information. Across all datasets, it achieves average improvements of 1.7%, 6.0%, and 2.0% over the state-of-the-art in P_1, ARI, and NMI, respectively.

The conventional topic model, LDA, demonstrated superior performance compared to language model-based approaches, such as BERTopic and PromptTopic, on Bills and Wiki. However, on Twitter, LDA's performance was inferior to PromptTopic and comparable to BERTopic. This limitation highlights the challenge traditional methods face in short text scenarios (Qiang et al., 2020), where the insufficient number of words in documents significantly constrains their effectiveness. PromptTopic exhibits the poorest performance on Bills and Wiki, primarily due to issues related to inconsistent topic granularity during the topic merging stage. This shortcut was particularly evident on the more complex Bills dataset, where the ARI was 28% lower than LiSA.

To provide a clear overview of the clustering results, we visualize the features of documents obtained by sentence-bert and LiSA in Figure 3. It can be observed that the topic prediction network has learned more effective document representations by better integrating the semantic information of topic words and documents, thereby achieving improved topic clusters. Specifically, Figure 3 (a) shows the results of k-means clustering on the sentence-bert embeddings of the documents. It can be seen that the distribution of documents with the same topic is not sufficiently concentrated, while the distinction between topics is not sufficiently clear. In contrast, Figure 3 (b) illustrates that our topic prediction network generates more distinct document representations while ensuring greater similarity among documents within the same topic cluster. This enhancement is attributed to the incorporation

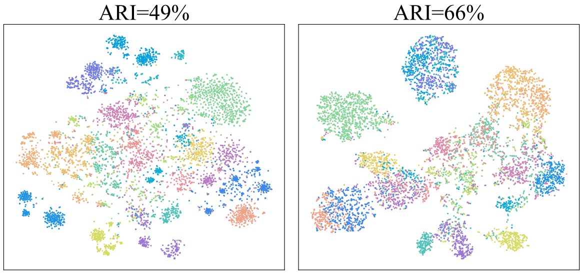


Figure 3: Clustering results of documents. (a) We directly perform K-Means (Macqueen, 1967) clustering on document embeddings obtained by Sentence-BERT. (b) We report the result of TPN's topic prediction and embeddings derived from TPN. For Figure (a), clustering is performed using K-Means. For Figure (b), document features are extracted from the final layer of TPN, and the clustering results from TPN are depicted in different colors.

of neighboring information from both topic words and documents.

Table 2: Topic Quality. We realize LDA by MALLET. For NMF and CTM, we utilize OCTIS to evaluate.

Datasets	Bills		Wiki		Twitter	
	C_V	TD	C_V	TD	C_V	TD
LDA	0.52	0.63	0.61	0.83	0.46	0.39
BERTopic	0.45	0.46	0.37	0.31	0.44	0.38
NMF	0.49	0.55	0.62	0.77	**0.48**	0.52
CTM	0.58	0.93	0.72	0.96	0.44	**1.00**
FASTopic	0.47	**1.00**	0.51	**1.00**	0.47	**1.00**
LiSA	**0.70**	0.96	**0.79**	0.97	**0.48**	0.91

4.2.2 Topic Quality

The comparison between LiSA and traditional topic models, as well as neural topic models, is shown in Table 2. We see that our LiSA surpasses all traditional topic models, as well as reaches competitive performance against strong neural topic models (FASTopic). Recent studies have highlighted that variations in dataset pre-processing settings, including factors such as minimum and maximum document frequency, vocabulary size constraints, and the use of stop word sets, significantly influence the outcomes of topic modeling (Wu et al., 2023, 2024a).

4.2.3 Human Evaluation on Granularity

To evaluate whether the generated topic labels (*i.e.*, summarizing topic words) are consistent with the human-annotated ground truth labels, we invited three experts to assess the quality and granularity of the topics. For topic quality, we first measured the proportion of missing topics by having the experts report the number of true labels that could not be found in the topic labels generated by our model as semantically equivalent. Similarly, we measured whether the generated labels exhibited semantic redundancy, with experts identifying pairs of semantically equivalent topic labels produced by the model. For topic granularity, we predefined 1-2 overly broad and overly narrow topic words for each dataset. For example, in the Bills dataset, an overly broad topic word is "Law," while an overly narrow topic word is "radiological materials security." The experts were then asked to report the number of topic labels generated by the model that matched the given word granularity. The evaluation results are shown in Table 3. It can be observed that LiSA successfully uncovers most of the topics, demonstrating the effectiveness of our approach in topic mining. Additionally, the granularity of the topics generated by LiSA is more consistent, with significantly fewer overly broad topics compared to PromptTopic. The summarizing topic words along with a case study can be found in the Appendix E.2. More details about human evaluation are listed in Appendix C.1.

Table 3: Human evaluation results.

	PromptTopic	LiSA
undetected	33%	23%
repeated	12	5
broad	11	4
narrow	1	0

4.3 Implementation Details

Following previous work (Pham et al., 2024; Grootendorst, 2022), we take sentence-bert (Reimers and Gurevych, 2019) as our $embedder$. Our experiments were conducted on a single A100 GPU using Llama-3-8B-Instruct and Mistral-7B-Instruct-v0.2. In the topic generation stage, we truncate the document if the prompt exceeds the context window length. Our TPN is an MLP network with a dimension of 256-256-K and produces a soft cluster assignment of each document. For evaluation, we choose the topic with the highest probability. We train TPN by AdaGrad (Duchi et al., 2011) optimizer with an initial learning rate of 1e-3 with a batch size of 128. We set $\tau = 1.5$ and $\alpha = 0.8$ for all datasets, and the training lasts 30 epochs on Bills, and 20 on Wiki and Twitter. For all baselines except TopicGPT, we set the number of topics (K) the same as that of the human-labeled ground truth. To achieve a fair comparison, we also report the performance of LiSA with the same number of topics as TopicGPT. For all datasets, we set the number of neighbors in the LLM-guided assignment stage (λ) as 5. As for the number of neighbors in the collaborative enhancement stage (μ), we set $\mu = 15$ for the Bills dataset, and $\mu = 20$ for the Wiki and Twitter datasets.

4.4 Ablation Study

4.4.1 Stages

Firstly, to provide a comprehensive understanding of the effectiveness of stages in our LiSA, we evaluate LiSA with different stages removed, *i.e.*, the collaborative enhancement stage and the LLMs-guided assignment stage. As shown in Table 4, the performance of LiSA without the collaborative

Table 4: The performance of our LiSA with different stages: without Collaborative Enhancement (w/o train); without check in LLMs-Guided Assignment stage (w/o check); without both of them (w/o train, check).

Dataset	Bills (K=21)			Wiki (K=15)		
	P_1	ARI	NMI	P_1	ARI	NMI
LiSA	**0.57**	**0.40**	**0.51**	**0.74**	**0.66**	**0.71**
w/o train	0.55	0.38	0.50	0.67	0.54	0.60
w/o check	0.56	**0.40**	0.48	0.70	0.56	0.61
w/o train, check	0.51	0.33	0.47	0.65	0.54	0.65

enhancement stage suffers a slight drop on Bills and a more significant decrease on Wiki, indicating the success of this stage in collaborating different levels of semantic granularity. Furthermore, after removing the LLM-guided assignment stage, our model's performance on the Bills dataset exhibited a slight decline, suggesting that it effectively leverages the language understanding capabilities of LLMs to rectify erroneous topic distributions.

4.4.2 Loss Function

Furthermore, we conduct an ablation experiment with various combinations of our loss functions $\mathcal{L}_1$, $\mathcal{L}_2$, and $\mathcal{L}_3$ to evaluate the effectiveness of each part, shown in Table 5. We can draw the following conclusion: **(1)** Among $\mathcal{L}_1$, $\mathcal{L}_2$, and $\mathcal{L}_3$, $\mathcal{L}_1$ plays a role in establishing a performance baseline. The primary function of $\mathcal{L}_1$ is to ensure the consistency between each document and its topic word. Optimizing $\mathcal{L}_1$ results in model performance converging to that of KMeans (Macqueen, 1967) on document embeddings.

(2) $\mathcal{L}_2$ further leverages the features of the neighbors surrounding the documents and topic words to enhance the performance. While the removal of L_2 did not affect the performance on the Wiki dataset, for the more challenging Bills dataset, utilizing the neighbor information of both topic words and documents is necessary. **(3)** We can find that $\mathcal{L}_3$ successfully prevents the predictions into the trivial solution that assigns most samples into a single cluster. Without $\mathcal{L}_3$, the model assigns most documents to only a few clusters, leading to poor clustering performance on both datasets.

Table 5: The performance of our LiSA with different combinations of loss functions $\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3$.

Dataset	Bills			Wiki		
	P_1	ARI	NMI	P_1	ARI	NMI
$\mathcal{L}$	**0.57**	**0.40**	**0.51**	**0.74**	**0.66**	**0.71**
$\mathcal{L}_2 + \mathcal{L}_3$	0.49	0.31	0.48	0.60	0.46	0.63
$\mathcal{L}_1 + \mathcal{L}_3$	0.56	0.39	0.50	**0.74**	**0.66**	**0.71**
$\mathcal{L}_1 + \mathcal{L}_2$	0.54	0.36	0.49	0.70	0.63	0.69
$\mathcal{L}_1$	0.44	0.25	0.43	0.62	0.48	0.66
$\mathcal{L}_2$	0.52	0.29	0.49	0.62	0.49	0.65
$\mathcal{L}_3$	0.17	0.03	0.09	0.30	0.13	0.17

5 Conclusion

In this work, we propose LiSA to integrate clustering and LLMs for topic modeling. To address the instability issues in the topic refinement stage caused by topic words of varying granularities, LiSA construct a topic-level semantic space and then establish a mapping from documents to topics after clustering. Additionally, we address the inconsistencies between topic-level and document-level semantic spaces by training TPNs to learn neighboring information. Experimental results demonstrate that LiSA consistently outperforms the method based on GPT-4 with respect to the clustering metrics reflecting alignment with human-labeled ground truth, and shows competitive performance against strong Neural Topic Models on topic quality.

Limitations

Contexts Limits One limitation of our current method is the need to truncate documents to fit the context length limitations of LLMs. While truncation was necessary in our experiments, we do not consider it an ideal solution. Future work could explore the use of LLMs with longer context windows or generate topic words iteratively for documents that exceed the context window size.

Multilinguality We did not evaluate LiSA on non-English datasets, partly because LLaMA and Mistral were primarily pre-trained and fine-tuned on English language data. As a result, the instruction-following ability of these models significantly decreases for non-English languages. We look forward to assessing the performance of LiSA on multilingual models in the future.

Acknowledgment

This work was supported by National Science Foundation of China (Grant Nos.62376057) and the Start-up Research Fund of Southeast University (RF1028623234).

References

Aly Abdelrazek, Yomna Eid, Eman Gawish, Walaa Medhat, and Ahmed Hassan. 2023. Topic modeling algorithms and applications: A survey. *Information Systems*, 112:102131.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. 2009. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information retrieval*, 12:461–486.

Dimosthenis Antypas, Asahi Ushio, Jose Camacho-Collados, Vítor Silva, Leonardo Neves, and Francesco Barbieri. 2022. Twitter topic classification. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3386–3400.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*.

Garima Dhanania, Sheshera Mysore, Chau Minh Pham, Mohit Iyyer, Hamed Zamani, and Andrew McCallum. 2024. Interactive topic models with optimal transport. *arXiv preprint arXiv:2406.19928*.

Adji B Dieng, Francisco JR Ruiz, and David M Blei. 2020. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7).

Cédric Févotte and Jérôme Idier. 2011. Algorithms for nonnegative matrix factorization with the β-divergence. *Neural computation*, 23(9):2421–2456.

K Chidananda Gowda and GJPR Krishna. 1978. Agglomerative clustering using the concept of mutual nearest neighbourhood. *Pattern recognition*, 10(2):105–112.

Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.

Alexander Hoyle, Pranav Goel, Andrew Hian-Cheong, Denis Peskov, Jordan Boyd-Graber, and Philip Resnik. 2021. Is automated topic model evaluation broken? the incoherence of coherence. *Advances in neural information processing systems*, 34:2018–2033.

Alexander Miserlis Hoyle, Pranav Goel, Rupak Sarkar, and Philip Resnik. 2022. Are neural topic models broken? In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5321–5344.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Harold W Kuhn. 1955. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97.

Hugo Larochelle and Stanislas Lauly. 2012. A neural autoregressive topic model. *Advances in Neural Information Processing Systems*, 25.

J Macqueen. 1967. *Some methods for classification and analysis of multivariate observations*. University of California Press.

Leland McInnes, John Healy, Steve Astels, et al. 2017. hdbscan: Hierarchical density based clustering. *J. Open Source Softw.*, 2(11):205.

Stephen Merity, Nitish Shirish Keskar, and Richard Socher. 2018. Regularizing and optimizing lstm language models. In *International Conference on Learning Representations*.

Meta. 2024. Introducing meta llama 3: The most capable openly available llm to date. https://ai.meta.com/blog/meta-llama-3/. Accessed: 2024-05-19.

Yida Mu, Peizhen Bai, Kalina Bontcheva, and Xingyi Song. 2024a. Addressing topic granularity and hallucination in large language models for topic modelling. *arXiv preprint arXiv:2405.00611*.

Yida Mu, Chun Dong, Kalina Bontcheva, and Xingyi Song. 2024b. Large language models offer an alternative to the traditional approach of topic modelling. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 10160–10171, Torino, Italia. ELRA and ICCL.

Andrew Ng, Michael Jordan, and Yair Weiss. 2001. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 14.

Chau Pham, Alexander Hoyle, Simeng Sun, Philip Resnik, and Mohit Iyyer. 2024. Topicgpt: A prompt-based topic modeling framework. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2956–2984.

Jipeng Qiang, Zhenyu Qian, Yun Li, Yunhao Yuan, and Xindong Wu. 2020. Short text topic modeling techniques, applications, and performance: a survey. *IEEE Transactions on Knowledge and Data Engineering*, 34(3):1427–1445.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992.

Michael Röder, Andreas Both, and Alexander Hinneburg. 2015. Exploring the space of topic coherence measures. In *Proceedings of the eighth ACM international conference on Web search and data mining*, pages 399–408.

Kuniaki Saito, Chen-Yu Lee, Kihyuk Sohn, and Yoshitaka Ushiku. 2025. Where is the answer? an empirical study of positional bias for parametric knowledge extraction in language model. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1252–1269.

Tian Shi, Kyeongpil Kang, Jaegul Choo, and Chandan K Reddy. 2018. Short-text topic modeling via non-negative matrix factorization enriched with local word-context correlations. In *Proceedings of the 2018 world wide web conference*, pages 1105–1114.

Suzanna Sia, Ayush Dalmia, and Sabrina J. Mielke. 2020. Tired of topic models? clusters of pretrained word embeddings make for fast and good topics too! In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1728–1736, Online. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pretraining for language understanding. *Advances in neural information processing systems*, 33:16857–16867.

Mark Steyvers and Tom Griffiths. 2007. Probabilistic topic models. In *Handbook of latent semantic analysis*, pages 439–460. Psychology Press.

Han Wang, Nirmalendu Prakash, Nguyen Khoi Hoang, Ming Shan Hee, Usman Naseem, and Roy Ka-Wei Lee. 2023. Prompting large language models for topic modeling. In *2023 IEEE International Conference on Big Data (BigData)*, pages 1236–1241. IEEE.

Xiaobao Wu, Xinshuai Dong, Thong Thanh Nguyen, and Anh Tuan Luu. 2023. Effective neural topic modeling with embedding clustering regularization. In *International Conference on Machine Learning*, pages 37335–37357. PMLR.

Xiaobao Wu, Thong Nguyen, and Anh Tuan Luu. 2024a. A survey on neural topic models: methods, applications, and challenges. *Artificial Intelligence Review*, 57(2):18.

Xiaobao Wu, Thong Nguyen, Delvin Zhang, William Yang Wang, and Anh Tuan Luu. 2024b. Fastopic: Pretrained transformer is a fast, adaptive, stable, and transferable topic model. *Advances in Neural Information Processing Systems*, 37:84447–84481.

A Dataset Details

The statistical information of our chosen datasets is shown in Table A1. The average number of tokens per document was calculated using Llama-3's tokenizer[3] (Meta, 2024).

Table A1: Statistical information of datasets: number of documents in the train split (# Train), average number of tokens per document in train (Doc length), number of documents in the test split (# Test), average number of tokens per document in test (Doc length), number of ground truth classes/labels.

Dataset	# Train	Doc length	# Test	Doc length	# Classes
Bills	32,661	259	15,242	407	21
Wiki	14,290	3,406	8,024	3,869	15
Twitter	6,798	46	4,373	45	6

B Parameter Analysis

B.1 Number of topic clusters (K)

It can be seen from Figure B1 (a) that the P_1 score of LiSA remains relatively stable across different values of K, indicating the robustness of our method. Moreover, ARI and NMI exhibit contrasting trends as K varies: as the number of clusters deviates from the ground truth, ARI generally shows a declining trend. This behavior can be attributed to ARI's heightened sensitivity to cluster number, which favors alignment with the ground truth. In contrast, NMI exhibits an increasing trend as the K value rises.

B.2 Number of Neighbors (λ)

As illustrated in Figure B1 (b), LiSA demonstrates robust stability in performance with respect to changes in the number of neighbors in the LLM-guided assignment stage. When λ varies within a smaller range, LiSA's performance remains relatively consistent across all datasets. However, as λ increases to 15, there is a noticeable decline on the Bills dataset, contrasting with the stable trend observed on the Wiki dataset. This discrepancy may be attributed to the presence of topics in the Bills dataset that contain relatively few documents. In this case, neighbors may introduce erroneous information, thereby increasing noise and reducing the accuracy of LLMs' responses.

B.3 Number of Neighbors (μ)

Figure B1 (c) shows that LiSA demonstrates strong stability concerning the number of neighbors in the collaborative enhancement stage. While smaller values of μ may hinder the model's ability to fully utilize neighbors, resulting in a slight performance decline, the performance of LiSA stabilizes once μ exceeds 15.

C Implementation Details

Following previous work (Pham et al., 2024; Grootendorst, 2022), we take sentence-bert (Reimers and Gurevych, 2019) as our $embedder$. Our experiments were conducted on a single A100 GPU using Llama-3-8B-Instruct and Mistral-7B-Instruct-v0.2. In the topic generation stage, we truncate the document if the prompt exceeds the context window length. Our TPN is an MLP network with a dimension of 256-256-K and produces a soft cluster assignment of each document. For evaluation, we choose the topic with the highest probability. We train TPN by AdaGrad (Duchi et al., 2011) optimizer with an initial learning rate of 1e-3 with a batch size of 128. We set $\tau = 1.5$ and $\alpha = 0.8$ for all datasets, and the training lasts 30 epochs on Bills, and 20 on Wiki and Twitter. For all baselines except TopicGPT, we set the number of topics (K) the same as that of the human-labeled ground truth. To achieve a fair comparison, we also report the performance of LiSA with the same number of topics as TopicGPT.

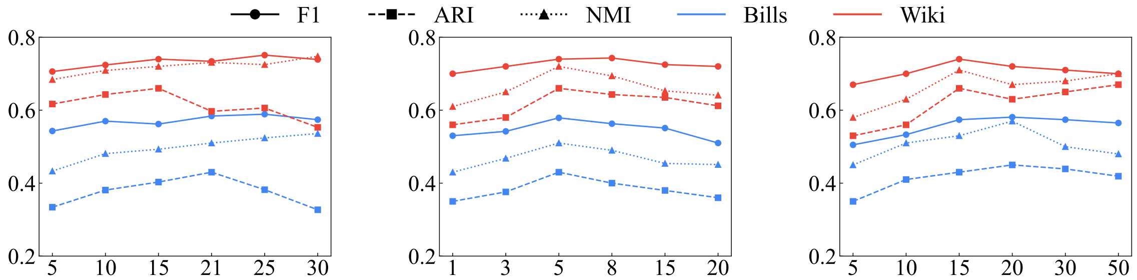
C.1 Human Evaluation Details

In this section, since our dataset involves the fields of legal, linguistic, and social networks, we invited three experts from law, linguistics, and the school of computer science at the author's institution to evaluate the generated topic words. The evaluation involved comparing the topic words generated by (Wang et al., 2023), as well as LiSA, with the standard labels annotated in the original dataset (denoted as A). Importantly, they did not know that the comparison context was topic modeling in advance and only evaluated the generated topic words based on their own knowledge. Detailed evaluation steps are as follows:

We initially categorized topic words along four dimensions: *undetected*, *repeated*, *broad*, and *narrow*. Then, for each dimension, we set specific requirements:

a) For the *undetected* dimension, we randomly sampled one topic from A and asked experts to determine whether this topic shared the same meaning with any topic generated by topic models. Thus,

[3] https://huggingface.co/meta-llama/Meta-Llama-3-8B



(b) Number of clusters K (b) Number of nearest neighbors during correction λ (c) Number of nearest neighbors during training μ

Figure B1: Parameter analysis results on three hyperparameters.

for each topic, this dimension had two possible options: *appeared* or *not appeared*.

b) For the *repeated* dimension, we paired the topics generated by the same topic model and asked the experts whether they shared the same meaning. For instance, Solicitor and Lawyer, Car and Automobile, Computer and Magazine. Thus, for each pair of topics, this dimension had two options: *repeated* or *not repeated*.

c) For the *broad* dimension, we predefined an upper-level topic granularity for each dataset. Examples include Law for the Bills dataset, Knowledge for the Wiki dataset, and Post for the Twitter dataset. We asked the experts to assess whether a generated topic surpassed the predefined level of topic granularity. If it exceeded or matched this granularity level, the dimensional value would be set to not broader. Thus, for each topic, this dimension had two options: *broader* or *not broader*.

d) For the *narrow* dimension, we predefined a lower-level topic granularity for each dataset. For example, for the Bills dataset, the lower-level category topic is "radiological materials security"; for the Wiki dataset, the lower-level category topic is "intellectually disabled services"; and for the Twitter dataset, the lower-level category topic is "DC extended universe". We asked experts to assess whether the generated topic word was at or below the predefined level of topic granularity. If so, the dimension value would be set to narrower. Thus, for each topic, this dimension had two options: *narrower* or *not narrower*.

In the human evaluation, we requested three experts to evaluate three datasets within a specified time frame. The post-processing steps were as follows: For each of the above four dimensions, the score of the three experts were aggregated using a voting mechanism—if two or more experts chose the same option, the corresponding topic would be classified under that option. The calculation methods for the results reported in this paper are defined as follows:

a) For the *undetected* dimension, the result is calculated as:

$$\frac{|\text{Not appeared}|}{|A|} \times 100$$

b) For the *repeated* dimension, the result is represented as: $|\text{Repeated}|$.

c) For the *broad* dimension, the result is represented as $| \geq |$.

d) For the *narrow* dimension, the result is represented as $| \leq |$.

Each expert assessed a total of 150 samples from Wiki (15 topics), 273 samples from Bills (21 topics), and 33 samples from Twitter (6 topics), resulting in a total of 456 evaluation samples. The evaluation was carried out over five days, with a total compensation of 22.8 USD, calculated based on a rate of 0.05 USD per sample ($456 \times 0.05 = 22.8$).

C.2 Evaluation Metric Details

Given a set of ground truth classes $X = \{x_1, \ldots, x_J\}$ and a set of predicted assignment clusters $Y = \{y_1, \ldots, y_K\}$, we evaluated the alignment between Y and X using external evaluation metrics for clustering, as detailed below. P_1 measures clustering purity, and its values lie between 0 and 1. ARI evaluates the agreement between clustering results and the ground truth, and its possible values range from -1 to 1. NMI measures the correlation between two clusters, with values spanning from 0 to 1. Higher values of these metrics indicate better clustering performance.

C.2.1 P_1

The Harmonic Mean of Purity (Amigó et al., 2009) is a clustering evaluation metric that balances purity and inverse purity. It ensures clusters are both homogeneous and comprehensive, providing a single measure that accounts for both cluster accuracy and completeness:

$$P_1 = \sum_k \frac{|y_k|}{N} \max_j F(y_j, x_k) \tag{12}$$

where

$$F(y_k, x_j) = \frac{2 \cdot \text{Precision}(y_k, x_j) \cdot \text{Recall}(y_k, x_j)}{\text{Precision}(y_k, x_j) + \text{Recall}(y_k, x_j)} \tag{13}$$

and

$$\text{Precision}(x_k, y_j) = \frac{|x_k \cap y_j|}{|x_k|} \tag{14}$$

$$\text{Recall}(\mathbf{X}, \mathbf{Y}) = \text{Precision}(\mathbf{Y}, \mathbf{X}) \tag{15}$$

C.2.2 ARI

Rand Index (RI) calculates the proportion of pairs of elements that are consistently clustered together or consistently separated in both the true clustering and the predicted clustering. ARI adjusts the Rand Index for the chance grouping of elements. It compares the actual Rand Index to the expected Rand Index (under random labeling), taking into account the possibility that clusters could be matched by chance:

$$ARI(X, Y) = \frac{\text{RI} - \mathbb{E}[\text{RI}]}{\max(\text{RI}) - \mathbb{E}[\text{RI}]} \tag{16}$$

where

$$RI(X, Y) = \frac{TP + TN}{TP + FP + FN + TN} \tag{17}$$

C.2.3 NMI and MI

Mutual Information (MI) is a measure of the mutual dependence between the ground truth labels (X) and the predicted clusters (Y). It quantifies the amount of information obtained about one random variable through the other. MI is calculated by summing the joint probability of X and Y over all possible values, weighted by the logarithm of the ratio between the joint probability and the product of the individual probabilities.

$$MI(X, Y) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \log \left(\frac{p(x, y)}{p(x)p(y)} \right) \tag{18}$$

where $p(x, y)$ is the joint probability distribution of clusters x in set X and y in set Y. $p(x)$ and $p(y)$ are the marginal probability distributions of clusters x and y respectively.

Normalized Mutual Information (NMI) is a metric used to measure the similarity between the ground truth labels (X) and the predicted clusters (Y). It quantifies how much information is shared between X and Y, normalized by the average entropy of both. This normalization ensures that NMI values range between 0 and 1, where 1 indicates perfect correlation between the clustering results and the true labels, while 0 indicates no correlation. NMI is particularly useful in scenarios where the number of clusters may differ from the number of true classes, making it a robust measure for comparing clustering algorithms across various datasets.

$$NMI(Y, X) = \frac{MI(Y, X)}{\left[\frac{H(Y)+H(X)}{2}\right]} \tag{19}$$

D Prompt Templates

In this section, we provide the prompt template for Llama-3 used in topic generation in Table D1, summarizing topic word generation in Table D2, and the LLM-guided assignment in Table D3. To better regulate the output from Mistral, we configure it to return all responses in JSON format, compensating for its relatively lower ability in following instructions in Table D2 and Table D3.

Table D1: Prompt template for topic generation, where **{text}** serves as a placeholder to be replaced by different document content.

Prompt Template for Topic Generation
Given the following document, your task is to generate a topic word for the article, followed by a short description of the meaning of the topic given by you. Your response should follow the JSON format, with the first key being 'topic_word', the second key being 'description'. The value corresponding to the first key is the topic word, and the value corresponding to the second key is the description of the topic. The description should be no more than two sentences. Return only the JSON data without any explanation. [Instructions] - The topic should be a single word or a short phrase of 2-3 words. [Document] **{text}** [Your response]

Table D2: Prompt template for summarizing topic word generation, where **{topic_list}** serves as a placeholder to be replaced by topic words from the same cluster.

Prompt Template for Summarizing Topic Word
Given a group of topic words, your task is to generate one general topic word that can describe as many topic words from this group as possible. Just tell me the topic word without any explanation or context. Topic words: **{topic_list}** Your response:

Table D3: Prompt template for LLM-guided assignment, where **{text}** serves as a placeholder to be replaced by different document content from D_w, and **{topic_list}** represents the corresponding topic words from its λ neighbors.

Prompt Template for Topic Generation

Given a document and several candidate topics, you should decide which topic the document belongs to. If the document is not related to any of the candidate topics, you should respond "None". Return the topic that best describes the document. Do not provide any explanation or context.
Document: **{text}**
Candidate topics: **{topic_list}**
Your chosen topic:

E Additional Experimental Results

In this section, we experiment with different clustering methods for our LiSA. Also, the summarizing topic words generated on Wiki (Hoyle et al., 2022) and Twitter (Antypas et al., 2022) are shown in Table E2. In all experiments, the hyperparameter K was set to match the number of ground truth labels, and all experiments were conducted using llama-3 (Meta, 2024).

E.1 Clustering Methods

In the main experiment, we chose K-Means (Macqueen, 1967) as it is the most widely used clustering method. To explore whether our method is applicable to different clustering algorithms, we performed experiments using three additional clustering techniques, including Spectral Clustering (Ng et al., 2001), Agglomerative Clustering (Gowda and Krishna, 1978), and HDBSCAN, which represent soft and hard clustering approaches. We chose HDBSCAN (McInnes et al., 2017) because it is also the clustering method adopted by BERTopic (Grootendorst, 2022). Table E1 illustrates the strong robustness of our model across various clustering algorithms. This robustness stems from our method's ability to improve model performance through the mutual optimization of both topic word clustering and document clustering. Notably, K-Means and Spectral Clustering demonstrate superior performance compared to the other algorithms. In contrast, HDBSCAN's performance is comparatively weaker. This discrepancy arises because HDBSCAN is a soft clustering method, which limits our ability to control the number of clusters K. Consequently, we utilized the official BERTopic package for clustering, merging the results into K clusters by adjusting the 'nr_topics' hyperparameter. Despite the slight performance drop observed with HDBSCAN, it still outperforms other clustering algorithms on certain metrics, further confirming the robustness and stability of our model.

E.2 Case Study on All Datasets

It can be observed from Table E2 that the topic words generated by our model exhibit a high level of consistency with the ground truth labels in the dataset. Additionally, we present the number of documents assigned to each topic, ranking them from highest to lowest. Our topic partitioning on the Wiki and Twitter datasets aligns more closely with the ground truth, particularly for topics containing a larger number of documents, where the model-generated topic words are more accurate. Conversely, for topics with many fewer documents, our method is less likely to produce consistent topic words. Additionally, the consistency of the Ω of the Bills dataset (Hoyle et al., 2022) with the ground truth is lower compared to the other two datasets. This discrepancy arises because the Bills test set contains documents from only 19 topics, rather than the 21 topics present in the training set. Consequently, our model produced finer-grained topics, potentially decomposing some of the ground truth topics into multiple subtopics.

Table E1: Experimental results of our LiSA with different clustering methods. The best results are denoted in **bold**.

Clustering Methods	Bills			Wiki			Twitter			
	P_1	ARI	NMI	P_1	ARI	NMI	P_1	ARI	MI	NMI
HDBSCAN	0.56	**0.42**	0.50	0.73	0.66	**0.71**	0.76	**0.65**	**0.71**	0.46
Agglomerative Clustering	**0.57**	**0.42**	0.50	0.73	**0.67**	0.70	**0.77**	0.64	**0.71**	**0.47**
Spectral Clustering	**0.57**	**0.42**	**0.51**	**0.75**	**0.67**	**0.71**	0.76	**0.65**	**0.71**	0.46
K-Means	**0.57**	0.40	**0.51**	0.74	0.66	**0.71**	**0.77**	0.64	0.70	0.46

Table E2: Summarizing topic words generated by LiSA and the corresponding ground truth labels.

Dataset	Topic Words		
Bills	Ground Truth		
	health: 2316	government operations: 1937	domestic commerce: 1323
	Defense: 1297	Public Lands: 1232	Law and Crime: 967
	Environment: 692	Transportation: 663	Energy: 661
	Macroeconomics: 610	International Affairs: 554	Labor: 493
	Foreign Trade: 471	Education: 448	Social Welfare: 436
	Civil Rights: 353	Technology: 321	Housing: 306
	Agriculture: 162		
	LiSA		
	healthcare :1336	government :1207	national heritage: 1161
	human services: 1086	defense :1034	policy reform: 942
	veterans: 850	energy: 779	disaster relief: 755
	transportation safety: 669	employee benefits: 669	water: 665
	education: 648	trade: 628	medical: 584
	civil rights: 565	criminal justice: 450	immigration: 428
	chemicals regulation: 311	reform: 247	regulation: 228
Wiki	Ground Truth		
	Media and drama: 1217	Sports and recreation: 1018	Warfare: 978
	Music: 976	Natural sciences: 952	Engineering and technology: 615
	Social sciences and society: 589	History: 446	Video games: 362
	Geography and places: 276	Language and literature: 235	Art and architecture: 199
	Philosophy and religion: 114	Agriculture, food, and drink: 38	Mathematics: 9
	LiSA		
	sports: 969	music: 887	television: 862
	transportation: 759	education: 702	science: 597
	warships: 476	biography: 462	war: 447
	weather: 424	mythology: 395	gaming: 356
	biology: 257	government: 235	history: 196
Twitter	Ground Truth		
	pop_culture: 1705	sports__gaming: 1528	daily_life: 647
	science_ _technology: 209	business__entrepreneurs: 195	arts__culture: 90
	LiSA		
	sports: 1479	music: 1233	technology: 514
	environment: 505	celebrity and culture: 403	health: 240