MIST: Mutual Information Maximization for Short Text Clustering

Anonymous ACL submission

Abstract

Short text clustering poses substantial challenges due to the limited amount of information provided by each text sample. Previous efforts based on dense representations are still inadequate since texts from different clusters are not sufficiently segregated in the embedding space prior to the clustering stage. Even though the state-of-the-art technique integrated contrastive learning with a soft clustering objective to address this issue, the process of summarizing all local tokens to form a sequence representation for the whole text may include noise that obscures the key information. We propose a framework called MIST: **M**utual **I**nformation Maximization for **S**hort **T**ext Clustering, which overcomes the information limitation by maximizing the mutual information between texts on both sequence and token levels. We assess the performance of our proposed method on eight standard short text datasets. Experimental results show that MIST outperforms the state-of-the-art methods in terms of Accuracy or Normalized Mutual Information in most cases.

1 Introduction

Text clustering is a crucial task for a wide range of downstream applications. It aims to partition texts into groups of similar categories in an unsupervised manner. The growth of social media, discussion forums and news aggregator websites has led to a large number of short-length texts being produced daily. Therefore, clustering short texts is gaining more attention and becoming a vital step for many real-world applications ranging from recommendation to text retrieval (Yohannes and Assabie, 2021).

In short texts, words and phrases that are most representative of the text content usually appear only once. This exacerbates the sparsity problem, posing an additional hurdle for clustering short texts. Traditional methods, such as BoW and TF-IDF, provide relatively sparse representation vectors with limited descriptive power. Hence, they perform poorly when clustered using a standard distance-based clustering algorithm, such as the k-means algorithm (Hadifar et al., 2019).

To address this problem, deep neural networks have been employed to map high-dimensional data into meaningful dense representations in a lower-dimensional space. Most recent deep clustering methods utilize a multi-stage scheme in which the clustering process is performed after learning feature representations (Xu et al., 2017; Hadifar et al., 2019; Yin et al., 2021). Unfortunately, the clustering performance of these methods remains unsatisfactory. One plausible explanation is that, texts still have a lot of overlap among categories in the latent space before clustering (Zhang et al., 2021).

Alternatively, an end-to-end clustering strategy simultaneously optimizes representation learning and clustering objectives (Zhang et al., 2021; Xie et al., 2016). To achieve desirable outcomes, Zhang et al. (2021) propose a method that adopts contrastive representation learning, which has been successful in self-supervised learning and can help spread out overlapping categories so that effective representations can be acquired, and optimize it together with a soft clustering objective.

As shown in Zhang et al. (2021), improving representation is crucial for enhancing the clustering performance. Nevertheless, the contrastive learning method used in Zhang et al. (2021) only considers whole text representations while optimizing a contrastive objective. In particular, these representations are formed by summarizing all token representations in each text instance via mean pooling, including uninformative noise. We conjecture that this may allow creating a representation in which important information used to describe the text content is obscured by noise, potentially affecting the clustering performance. Hence, there is still a gap that needs to be explored in order to construct an efficient representation for short text clustering that emphasizes on maintaining informative terms.

In this paper, we introduce the **M**utual **I**nformation Maximization Framework for **S**hort **T**ext Clustering (MIST), which adopts a multi-stage approach. We focus primarily on improving the representation learning stage that learns textual representations by incorporating two contrastive learning objectives together with an auxiliary clustering objective. The crux of our framework lies in this contrastive learning procedure based on mutual information (MI) maximization. This mechanism facilitates us in comparing the semantic similarity across different hierarchical levels to achieve multiple purposes. First, we learn distinct text representations by maximizing MI at the sequence-level between entire text representations. Second, we also attempt to preserve important information by enforcing each text representation to extract information that is shared across all of its individual words in the text. To accomplish this, we design an additional learning objective to maximize the MI between each sequence-level representation and all local tokens in the sequence. For the clustering stage, we apply the k-means algorithm at inference time to get the final clusters.

MIST handles the substantial challenge of short text clustering, and our contributions are as follows:

- We propose a novel representation learning technique for short text clustering. In particular, our representation learning solution integrates two MI maximization objectives: sequence-level and token-level MI maximization to learn short text representations.
- To balance both MI maximization objectives in the learning stage, we introduce a simple dynamic weighting function that adjusts the ratio of the objectives in accordance with the length of local tokens in each minibatch.
- We conduct extensive experimental studies to evaluate the performance of MIST over eight standard benchmarks for short text clustering. MIST improves the clustering performance in terms of Accuracy and NMI for most cases compared to the current state-of-the-art.

2 Related Work

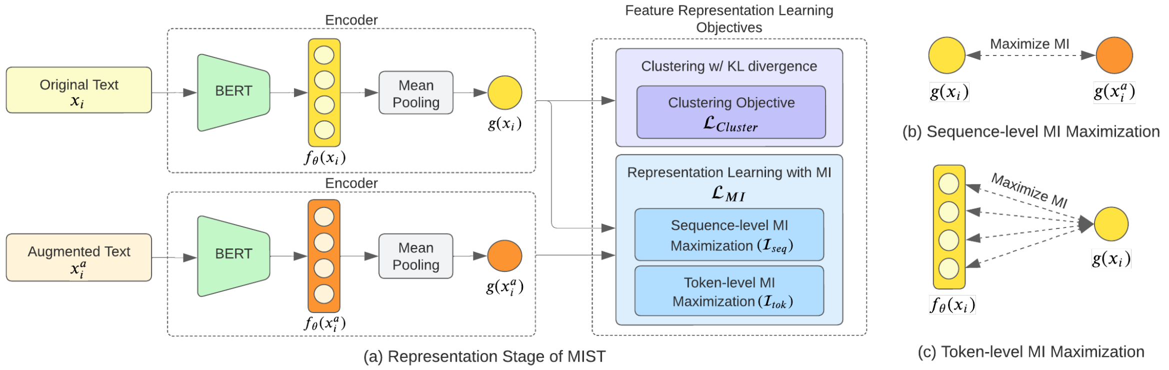
Short Text Clustering. There are several approaches to overcome the sparsity of short text representations, such as (1) multi-stage approaches which break down the clustering framework into multiple stages, (2) clustering enhancement algorithms that apply outlier removal, and (3) a joint framework that simultaneously optimizes both representation learning and clustering objectives.

Several multi-stage works perform clustering after learning feature representations. Pretrained-word embeddings (Mikolov et al., 2013a,b; Pennington et al., 2014) and neural networks are adopted to transform data into meaningful representations. Xu et al. (2015, 2017) use a convolutional neural network to learn non-biased representations by fitting the output units with pretrained-binary codes from a dimensionality reduction method. Hadifar et al. (2019) utilize Smooth Inverse Frequency (SIF) (Arora et al., 2017) to obtain weighted word embeddings. During training, they enrich discriminative features by tuning an autoencoder with soft clustering assignments from a clustering objective. For the aforementioned works, k-means clustering is then employed on trained representations to get final clusters.

Another approach is to enhance the quality of the initial clustering with an iterative classification algorithm. Rakib et al. (2020) proposed the ECIC algorithm to detect and remove outliers in each iteration. Moreover, they make use of word embeddings by averaging them to represent each text, and combine the ECIC algorithm with hierarchical clustering. To boost the clustering quality further, Pugachev and Burtsev (2021) exploit deep sentence representations (Cer et al., 2018) and make modifications to the ECIC algorithm.

The recent state-of-the-art, SCCL (Zhang et al., 2021), leverages contrastive learning to encourage greater separation between overlapped categories in the original data space. By jointly optimizing a contrastive loss and a clustering objective (Reimers and Gurevych, 2019a), SCCL outperforms prior works and yields cutting-edge results. In addition, other constrastive learning methods have also been experimented on short text clustering, such as using entities for contrastive learning to provide supervision signals for their related sentences (Nishikawa et al., 2022), and using virtual augmentation for contrastive learning to circumvent the discrete nature of language (Zhang et al., 2022). However, these methods do not outperform SCCL on short text clustering.

Self-Supervised Learning. Self-supervision has gained popularity and become a common technique in unsupervised representation learning for a variety of downstream purposes. Many recent accomplishments have been based on contrastive repre-


Encoder
Original Text x_i
BERT
Mean Pooling
$f_\theta(x_i)$
$g(x_i)$
Encoder
Augmented Text x_i^a
BERT
Mean Pooling
$f_\theta(x_i^a)$
$g(x_i^a)$
Feature Representation Learning Objectives
Clustering w/ KL divergence
Clustering Objective $\mathcal{L}_{Cluster}$
Representation Learning with MI $\mathcal{L}_{MI}$
Sequence-level MI Maximization ($\mathcal{I}_{seq}$)
Token-level MI Maximization ($\mathcal{I}_{tok}$)
Maximize MI
$g(x_i)$
$g(x_i^a)$
Maximize MI
$f_\theta(x_i)$
$g(x_i)$

(a) Representation Stage of MIST

(b) Sequence-level MI Maximization

(c) Token-level MI Maximization

Figure 1: (a) Representation Learning Stage Overview. MIST considers all pairs of original text x_i, and its augmented version x_i^a as positives. MIST jointly optimizes the clustering objective $\mathcal{L}_{\text{Cluster}}$, and the MI objective $\mathcal{L}_{\text{MI}}$. The $\mathcal{L}_{\text{MI}}$ comprises (b) a sequence-level MI maximization objective $\mathcal{I}_{\text{seq}}$, which attempts to maximize MI between sequence representations of x_i and x_i^a, along with (c) a token-level MI maximization objective $\mathcal{I}_{\text{tok}}$ that maximizes MI between a sequence representation (of both x_i and x_i^a) and its tokens ($f_\theta(x_i)$ and $f_\theta(x_i^a)$).

sentation learning (Chen et al., 2020; He et al., 2020; Caron et al., 2020; Grill et al., 2020).

Learning meaningful representations by estimating and maximizing MI is one of the prominent contrastive learning strategies. Its effectiveness has been demonstrated in both vision (Hjelm et al., 2019; Bachman et al., 2019; Sordoni et al., 2021) and text domains (Kong et al., 2020; Caron et al., 2020; Giorgi et al., 2021). Deep Infomax (DIM) (Hjelm et al., 2019) introduces global and local MI maximization objectives for learning image representations. Each objective is then used separately according to the task. The authors also find success in optimizing local MI maximization objective by maximizing MI between local features and global features. Inspired by local Deep InfoMax, Zhang et al. (2020) proposes a sentence representation learning method that maximizes the MI between the sentence-level representation and its CNN-based n-gram contextual dependencies.

In this work, we leverage the MI maximization strategy to learn textual representations for short text clustering. We also introduce a weighting function for appropriately balancing two MI maximization objectives to improve clustering results.

3 Proposed Method: MIST

We propose a short text clustering framework consisting of two steps: (1) we first train a model using feature representation learning objectives as illustrated in Figure 1 and (2) then apply the k-means algorithm on the trained representations at inference time to obtain the final clusters. This section primarily focuses on the representation learning stage. The main idea of our solution lies in the proposed objective function $\mathcal{L}$ that takes into account an MI objective $\mathcal{L}_{\text{MI}}$, which preserves a local invariance for each sample, and an unsupervised clustering objective $\mathcal{L}_{\text{Cluster}}$ to capture categorical structure.

$$\mathcal{L} = \beta\mathcal{L}_{\text{MI}} + \eta\mathcal{L}_{\text{Cluster}}, \tag{1}$$

where β and η represent the trade-off between $\mathcal{L}_{\text{MI}}$, and $\mathcal{L}_{\text{Cluster}}$. In our experiments, we set β to 1 and η to 2 to provide more weight to $\mathcal{L}_{\text{Cluster}}$.

We describe our proposed method in the following subsections. Section 3.1 provides a description of the MI maximization learning procedure, including (1) sequence-level and (2) token-level MI maximization objectives, along with a weighting function for balancing them. Note that the MI maximization framework takes both original and augmented texts as inputs. Section 3.2 presents the auxiliary clustering objective that enforces the encoder to create a suitable representation space for clustering.

3.1 Representation Learning with MI Maximization

One strategy to improve the clustering performance is to use representation learning to build an embedding space that minimizes local invariance for each individual sample. A prominent method for constructing such embedding space is contrastive learning which relies on contrasting representations throughout the whole context. Short text inputs are varied in terms of their lengths across different datasets. There are short texts with smaller sizes (e.g., 6-8 words) and longer texts (e.g., 22-28 words). The latter tends to contain more words that may not be beneficial for providing high-level semantics useful for clustering. Consequently, optimizing solely the global-level objective, as commonly done in contrastive learning, may not be sufficient to train effective representations for short texts with weak signals problem.

To prevent any local information from being obscured, we adopt an additional learning objective to constrain the representation of the entire text (global feature) to contain high MI with each of its token embeddings (local features). In this investigation, we refer to the global and local features as *sequence* and *token* representations, respectively. Therefore, we build our learning framework based on MI maximization strategy to reduce discrepancy between sequence- and token-level representations via their relative ability to predict each other across the representation levels.

Computing the MI Objective. As shown in Figure 1, the objective $\mathcal{L}_{\text{MI}}$ consists of two components: (1) sequence-level MI maximization $\mathcal{I}_{\text{seq}}$, and (2) token-level MI maximization $\mathcal{I}_{\text{tok}}$.

$$\mathcal{L}_{\text{MI}} = -(1-\lambda)\mathcal{I}_{\text{seq}} - \lambda\mathcal{I}_{\text{tok}}, \tag{2}$$

where λ corresponds to the balancing weight for $\mathcal{I}_{\text{seq}}$ and $\mathcal{I}_{\text{tok}}$ objectives.

Let us now consider the λ value, which is the weight for the $\mathcal{I}_{\text{tok}}$ objective. We want λ to be substantial when the total number of tokens in the text is large. This is owing to the fact that when the text is lengthy, the process of summarizing all the tokens in the text to form a sequence representation could include noise that obscures important keywords, which usually appear only once in short texts. In this situation, it is desirable to preserve the crucial information by optimizing the $\mathcal{I}_{\text{tok}}$ objective. In particular, λ for each minibatch of size N is calculated as follows.

$$\lambda = \max\left(0, \left\lfloor \frac{0.1}{N}\sum_{i=1}^{N} l_i \right\rceil - 1\right), \tag{3}$$

and l_i denotes the number of tokens in a text x_i. Further discussion can be found in Section 4.3.1.

In the representation learning stage, we first randomly sample a minibatch $X^o = x_1^o, ..., x_N^o$ of N original texts with empirical probability distribution $\mathbb{P}$. Then, we generate an augmented version for each text to obtain an augmented batch $X^a = x_1^a, ..., x_N^a$, where X^o and X^a are of identical size. The encoder is a pretrained transformer network f_θ that encodes an input text x into a sequence of contextualized token embeddings with length l, $f_\theta(x) := \{f_\theta^{(i)}(x) \in \mathbb{R}^d\}_{i=1}^{l}$, where i is the token index and d is the number of dimension. These token representations are then subsequently averaged by mean pooling operation $m(f_\theta(x))$ to generate a sequence representation denoted as $g(x) = m(f_\theta(x)) \in \mathbb{R}^d$.

Computing the Sequence-level MI. The first learning objective, $\mathcal{I}_{seq}$, aims to learn a representation that captures the entire context by maximizing MI between the original sample and its augmented version at the sequence-level. According to Tian et al. (2020), contrastive learning is equivalent to maximizing the lower bound of MI between the representations of two texts. By treating each original text $g(x^o)$ and its augmentation $g(x^a)$ as positive samples, we can define $\mathcal{I}_{seq}$ over the whole minibatch as follows.

$$\mathcal{I}_{seq} = \frac{1}{N}\left(\sum_{x\in X}\widehat{\mathcal{I}}^{JSD}(g(x^o); g(x^a))\right) \tag{4}$$

We adopt a Jensen-Shannon estimator (Nowozin et al., 2016; Hjelm et al., 2019) to estimate a lower bound of MI, $\widehat{\mathcal{I}}_\theta^{JSD}$:

$$\begin{aligned}\widehat{\mathcal{I}}_\theta^{JSD}&(g(x^o); g(x^a)) :=\\ &E_{\mathbb{P}}\left[-sp(-g(x^o)\cdot g(x^a))\right]\\ &- E_{\mathbb{P}\times\tilde{\mathbb{P}}}\left[sp(g(x^o)\cdot g(\tilde{x}^a))\right],\end{aligned} \tag{5}$$

where $\tilde{x}^a$ is a negative augmented textual input sampled from distribution $\tilde{\mathbb{P}} = \mathbb{P}$, and $sp(z) = \log(1+e^z)$ is the softplus function.

Computing the Token-level MI. To further enrich text representations, we include a second learning objective to MIST. Inspired by Zhang et al. (2020), this learning objective encourages a text representation to incorporate and preserve local information shared across all contextualized tokens. In particular, we attempt to maximize the average MI between a sequence representation and all of its token representations, while minimizing MI with the tokens of other texts. Conceptually, this reflects how much more precisely we can determine the representation of a token given a sequence representation compared to when we are unaware of the sequence representation (Bachman et al., 2019). We now define $\mathcal{I}_{tok}$ for each minibatch as

$$\begin{aligned}\mathcal{I}_{tok} = \frac{1}{2N}\Big(&\sum_{x^o\in X^o}\sum_{i=1}^{l_{x^o}}\widehat{\mathcal{I}}^{JSD}(g(x^o); f_\theta^{(i)}(x^o))\\ &+\sum_{x^a\in X^a}\sum_{i=1}^{l_{x^a}}\widehat{\mathcal{I}}^{JSD}(g(x^a); f_\theta^{(i)}(x^a)).\end{aligned} \tag{6}$$

An estimated MI for each sequence $g(x)$ and token representations $f_\theta^{(i)}(x)$ is as follows:

$$\begin{aligned}&\widehat{\mathcal{I}}_\theta^{JSD}(g(x); f_\theta^{(i)}(x)) := \\ &\quad E_{\mathbb{P}}[-sp(-g(x)\cdot f_\theta^{(i)}(x))] \\ &\quad - E_{\mathbb{P}\times\tilde{\mathbb{P}}}[sp(g(x)\cdot f_\theta^{(i)}(\tilde{x}))],\end{aligned} \tag{7}$$

where $\tilde{x}$ is a different text on the minibatch.

3.2 Clustering with KL Divergence

In addition to the MI objective, we employ an auxiliary clustering objective $\mathcal{L}_{\text{Cluster}}$ to encourage the coalescence of samples that are most likely to belong to the same cluster. We follow the clustering method proposed by Xie et al. (2016), which are also used by Hadifar et al. (2019); Yin et al. (2021) and Zhang et al. (2021). This method involves computing soft cluster assignments, and formulating the clustering objective using KL divergence.

For the first step, we follow Xie et al. (2016) using the Student's t-distribution Q to compute a soft cluster assignment for each text instance $x_j \in X$ and the centroid μ_k where $\mu_k \in \{1, ..., K\}$ for the dataset with K-clusters. Specifically, we compute the probability q_{jk} of assigning a text x_j to a cluster μ_k as follows.

$$q_{jk} = \frac{(1 + \|g(x_j) - \mu_k\|_2^2/\alpha)^{-\frac{\alpha+1}{2}}}{\sum_{k'=1}^{K}(1 + \|g(x_j) - \mu_{k'}\|_2^2/\alpha)^{-\frac{\alpha+1}{2}}} \tag{8}$$

The α symbol represents the degree of freedom of the distribution, and we set α to 1. Following Zhang et al. (2021), each centroid μ_k is approximated by the linear clustering head c_θ.

The second step is calculating an auxiliary target distribution P and utilizing it to assist in refining clusters' centroids. The main idea is to give more importance towards text samples with high clustering confidence. The probability $p_{jk} \in P$ is calculated as follows.

$$p_{jk} = \frac{q_{jk}^2/\sum_{j'} q_{j'k}}{\sum_{k'}(q_{jk'}^2/\sum_{j'} q_{j'k'})} \tag{9}$$

To match the soft cluster assigments to the target distribution, the KL-divergence between probability distributions P and Q is computed as follows.

$$\ell_j^C = KL\left[p_j || q_j\right] = \sum_{k=1}^{K} p_{jk} \log \frac{p_{jk}}{q_{jk}} \tag{10}$$

We then formulate it as a clustering objective for each minibatch of size N as

$$\mathcal{L}_{\text{Cluster}} = \sum_{j=1}^{N} \ell_j^C/N. \tag{11}$$

4 Experiments

4.1 Experimental Setup

Datasets. Following previous works (Rakib et al., 2020; Zhang et al., 2021; Pugachev and Burtsev, 2021), we conduct experiments and evaluate the performance of our method, MIST, on the eight standard short text clustering datasets: AgNews, SearchSnippets, StackOverflow, Biomedical, Tweet, GoogleNews-TS, GoogleNews-T and GoogleNews-S. The descriptions and statistics of the datasets are provided in Appendix A.1

Implementation. We implement our model in PyTorch (Paszke et al., 2017) and use the *paraphrase-mpnet-base-v2* in Sentence Transformers library (Reimers and Gurevych, 2019b) as the encoder, with a linear clustering head following Zhang et al. (2021). The encoder is trained for 1,200 iterations for all datasets and we use Adam optimizer with a batch size of 256. The learning rate of the encoder and the clustering head are set to 6e−6 and 6e−5, respectively. We follow Xu et al. (2017) and Hadifar et al. (2019) by randomly selecting 10% of data as the validation set. Furthermore, we follow Zhang et al. (2021) by not performing any preprocessing operations on all eight datasets. Although some of the existing works preprocess the texts by removing symbols, stop words, and punctuation, or converting them to lowercase.

In the training stage, the original and augmented texts are taken into consideration as inputs for the MI objective $\mathcal{L}_{\text{MI}}$, since we found that they are more effective than employing two augmented pairs. We follow Zhang et al. (2021) and utilize *Contextual Augmenter* (Kobayashi, 2018; Ma, 2019) to generate augmented samples for each text instance as it was demonstrated to produce the best outcomes in their study. Additionally, we choose BERT with 20% word substitution ratio for *Contextual Augmenter* which provides the best results as shown in Appendix A.5. To assess clustering performance, we use the same two standard metrics—Accuracy (ACC) and Normalized Mutual Information (NMI)— as used in all key existing methods (Xu et al., 2017; Hadifar et al., 2019; Rakib et al., 2020; Zhang et al., 2021). The Accuracy is calculated via the Hungarian algorithm, and NMI measures the information shared between the ground truth assignments and the predicted assignments. The results are averaged over five trials.

	AgNews		SearchSnippets		StackOverflow		Biomedical	
	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
Reported in the References								
BoW†	27.6	2.6	24.3	9.3	18.5	14.0	14.3	9.2
TF-IDF†	34.5	11.9	31.5	19.2	58.4	58.7	28.3	23.2
Skip-Thought‡	-	-	33.6	13.8	9.3	2.7	16.3	10.7
STCC	-	-	77.09	63.16	51.13	49.03	43.62	38.05
Self-Train‡	-	-	77.1	56.7	59.8	54.8	**54.8**	**47.1**
SCA-AE	68.36	34.14	68.71	50.26	76.55	65.99	40.25	33.29
HAC-SD	81.84	54.57	82.69	63.76	64.80	59.48	40.13	33.51
SCCL†	88.2	68.2	**85.2**	**71.1**	75.5	74.5	46.2	41.5
Reimplementation								
SCCL w/ BERT 20%	87.10	67.18	84.78	70.02	49.48	47.50	44.90	39.73
SCCL-Multi w/ BERT 20%	86.95	67.06	83.88	69.50	53.56	46.99	44.70	39.65
Proposed Method								
MIST	**89.47***	**70.25***	76.72	67.69	**78.74***	**77.59***	39.15	34.66

	Tweet		GoogleNews-TS		GoogleNews-T		GoogleNews-S	
	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
Reported in the references								
BoW†	49.7	73.6	57.5	81.9	49.8	73.2	49.0	73.5
TF-IDF†	57.0	80.7	68.0	88.9	58.9	79.3	61.9	83.0
Skip-Thought‡	-	-	-	-	-	-	-	-
STCC	-	-	-	-	-	-	-	-
Self-Train‡	-	-	-	-	-	-	-	-
SCA-AE	84.85	89.19	-	-	-	-	-	-
HAC-SD	89.62	85.20	85.76	88.00	**81.75**	84.20	80.63	83.50
SCCL†	78.2	89.2	89.8	94.9	75.8	88.3	**83.1**	90.4
Reimplementation								
SCCL w/ BERT 20%	55.98	82.12	75.35	90.96	62.53	81.95	67.88	86.07
SCCL-Multi w/ BERT 20%	79.05	89.59	88.83	94.69	76.20	87.89	82.25	90.01
Proposed Method								
MIST	**91.75***	**95.12***	**89.93***	**95.47***	75.97	**88.97***	81.91	**90.79***

Table 1: Experimental results on eight short text clustering datasets. † and ‡ refer to results taken from Zhang et al. (2021) and Hadifar et al. (2019), respectively; both originally present their results in one decimal place. * denotes that MIST is significantly better than both reimplemented versions of SCCL. In order to statistically compare models, we use Almost Stochastic Dominance test (Dror et al., 2019) with the significant level of 0.05.

4.2 Experimental Results

We compare the performance of our proposed framework, MIST, with state-of-the-art methods including STCC (Xu et al., 2017), Self-Train (Hadifar et al., 2019), HAC-SD (Rakib et al., 2020), SCA-AE (Yin et al., 2021) and SCCL (Zhang et al., 2021). As demonstrated in Table 1, MIST achieves state-of-the-art results for most cases in terms of Accuracy and NMI across eight benchmark datasets. In addition to the results reported in the reference papers, we further compare our method with SCCL, the state-of-the-art model that also employs contrastive learning for short text clustering, by reproducing SCCL in an end-to-end (original) version and a multi-stage version analogous to our architecture for a fair comparison. The reimplemented versions of SCCL use the same augmentation setting as our model. We refer to these models as SCCL w/ BERT 20% and SCCL-Multi w/ BERT 20%, respectively. The comparative results in Table 1 show that MIST outperforms SCCL, SCCL w/ BERT 20%, and SCCL-Multi w/ BERT 20% in 11, 12, and 10 cases, respectively.

For datasets with a small number of clusters, Search Snippets and Biomedical, MIST does not yield competitive results. We obtain an inferior result on Biomedical, since the dataset used to pretrain our encoder is a general domain one. On the other hand, Hadifar et al. (2019) produces the best result using pre-trained embeddings learned from a large in-domain biomedical corpus. For the SearchSnippets dataset, MIST also obtains a poorer result. One probable explanation is that snippets are typically composed of content words, as well as the dataset has been automatically crawled and preprocessed further by Phan et al. (2008), the preprocessing steps include removing stop and rare words. The length and incoherency of each text in this dataset has made our algorithm dependent on keywords rather than contextual information. This is particularly evident when the algorithm performs the token-level MI maximization objective during the representation learning stage, which enforces similarity between each contextualized token representation and the sequence representation of the incoherent text sequence. This can be even more problematic when the same keywords also appear in different clusters.

For datasets with a large number of clusters, such as GoogleNews, it is more likely that texts in different clusters may share similar content due to fine-grained categorization, hence inducing ambiguity. This ambiguity in textual data and ground truths is one of the factors that lead to erroneous predictions. As GoogleNews-T only contains news headlines, which are relatively short with few keywords. It presents a challenge for clustering these texts into a large number of clusters. For example, "`liam adam sentenced abuse daughter`" is a news headline in a cluster of news related to Gerry Adams, an IRA activist and the former president of Sinn Féin. This sample contains the same keywords as another cluster of articles regarding domestic violence. Moreover, another cause of inaccuracy is when the content of texts in one cluster is a subtopic of the content in another cluster.

We hypothesize that Rakib et al. (2020), which employs hierarchical clustering and outlier removal algorithms, can better deal with the hierarchical nature of data. Contrarily, both our method and SCCL are primarily focused on improving representations through the use of the MI maximization strategies and contrastive learning, which are equivalent. Thus, Rakib et al. (2020) outperforms our method and SCCL on this dataset in terms of Accuracy. MIST also has lower Accuracy on GoogleNews-T and GoogleNews-S than the reported result of SCCL in the reference paper and SCCL-Multi w/ BERT 20%, respectively. Notably, we collected the experimental results of SCCL w/ BERT 20% and SCCL-Multi w/ BERT 20% from the best iteration for each dataset instead of using a stopping criterion, which is not indicated in the publication.

Although GoogleNews-S and GoogleNews-TS share the same challenges as GoogleNews-T, clustering texts in both datasets is more accurate due to the benefit of additional context and information in the texts themselves. GoogleNews-S contains snippets of news, and GoogleNews-TS includes both titles and snippets. Consequently, MIST can derive a very strong and comparable Accuracy to SCCL on GoogleNews-S and outperforms SCCL on GoogleNews-TS.

Additional details and the results of SCCL in both reproduced versions with other augmentation settings can be found in Appendix A.6, which shows that MIST still outperforms SCCL in both end-to-end and multi-stage settings in 11 cases.

4.3 Ablation Study

To better understand the impact of each component in our training procedure on the clustering performance, we conduct additional experiments by varying the trade-off between sequence- and token-level MI maximization objectives in the MI loss $\mathcal{L}_{\text{MI}}$, as well as the clustering objective $\mathcal{L}_{\text{Cluster}}$.

4.3.1 The Impact of Sequence- and Token-MI Maximization Objectives

We report the clustering performance of MIST in four different ratios by setting λ in Eq.2 to 1, 0.5, 0, and by assigning λ using Eq. 3. In this section, we refer to the MIST model with a sequence-only ($\lambda = 0$) and a token-only ($\lambda = 1$) MI maximization objectives as MIST-seq and MIST-tok, respectively. As demonstrated in Figure 2, MIST with the ratio set according to Eq.3 yields the best performance in terms of Accuracy for most datasets. NMI tends to follow the same trend as Accuracy, as presented in Appendix A.2. This demonstrates that the length of texts, i.e., the numbers of tokens, is a crucial factor in determining the appropriate ratio for both MI maximization objectives. In addition, we also investigate the situation in which the MI objective is absent ($\beta = 0$). The ablation results show when the MI objective is removed, the performance on all datasets suffers drastically. This implies that the MI objective is essential for performance gain.

For datasets with long-length texts, such as GoogleNews-TS, we discovered that MIST produces the best outcomes when sequence-level and token-level MI maximization objectives are weighted using λ calculated by Eq. 3. Note that this setting also outperforms the scenario when both objectives are assigned equal weight ($\lambda = 0.5$). Remarkably, MIST-tok always outperforms MIST-seq. This shows that if the text is lengthy, MIST-seq is insufficient. This is because when generating a sequence representation, informative terms of the text are averaged with other non-informative terms via mean pooling. Then, important information in the text is more likely to be obscured. However, this is-

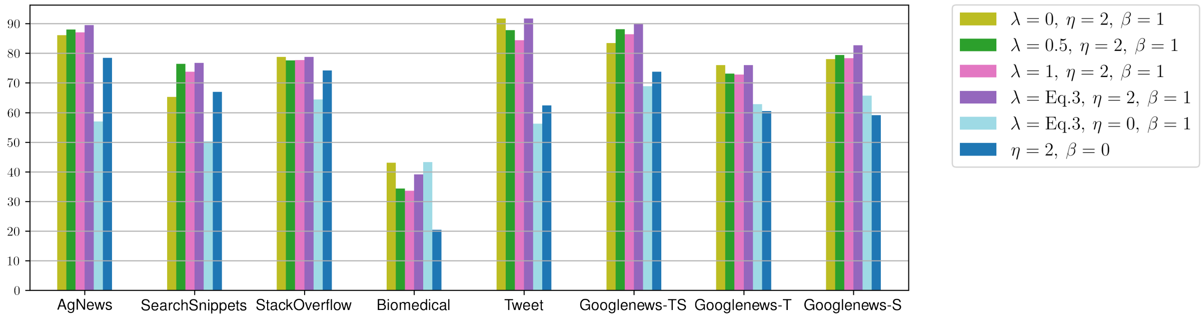


Figure 2: Accuracy for six different settings including four different weighting ratios between sequence-level and token-level MI maximization objectives. As well as, a setting where the clustering objective is absent ($\eta = 0$), and a setting where the MI objective is absent ($\beta = 0$). Note that when we set β to 0, λ has no effect.

sue can be mitigated by performing the local-level MI maximization.

For datasets with very short-length texts, such as StackOverflow and Tweet, the weighting ratio based on Eq. 3 is equivalent to setting λ to 0. For this setting, MIST is identical to MIST-seq. Interestingly, MIST-seq outperforms all other settings, followed by MIST with both the sequence-level and token-level MI maximization objectives combined, which consistently outperformed MIST-tok. For instance, texts in Tweet dataset are relatively short and contains only of content words rather than coherent texts. As a result, MIST-tok and MIST with the combination of both MI maximization objectives, might emphasize on keywords that could also appear in multiple clusters, causing ambiguity.

4.3.2 The Impact of the Clustering Objective

As shown in Figure 2, the clustering performance drops drastically when we remove the clustering objective ($\eta = 0$). This demonstrates that the categorical structure imposed by jointly optimizing the clustering objective with the MI objective is a crucial component that boosts performance. However, this trend holds true for all datasets, except for Biomedical. A possible reason is that, the encoder was not pretrained with textual data suitable for its specific domain, the clustering objective does not benefit the efficiency of our model as much as the MI objective.

Furthermore, we observe that as the weight of clustering objective increases, the performance continuously improves until it reaches saturation as η, the clustering weight, approaches 2. As depicted in Figure 3, the Accuracy and NMI of AgNews both improve as we gradually increase the clustering weight until it reaches the appropriate value, which is 2 in our experiment. The supplementary experimental results can be found in Appendix A.4.

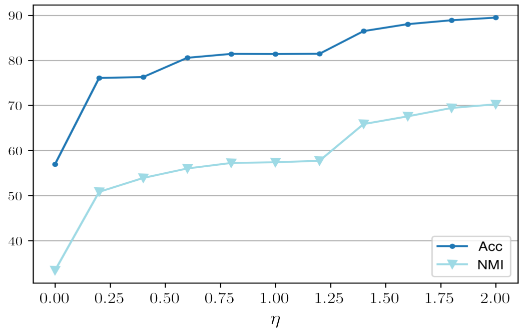


Figure 3: The clustering performance on AgNews based on the strength of the clustering objective, while the ratio of the MI objective is kept constant based on Eq 3.

5 Conclusion

We propose a novel multi-stage framework that employs two MI maximization objectives to produce effective representations for short text clustering. To learn distinct text representations,we first maximize MI between original texts and their augmentations at the sequence-level. While the second objective maximizes MI between sequence representations and their local tokens. Additionally, we introduce a preliminary weighting function for properly balancing the two MI maximization objectives during training stage.

We have conducted extensive experiments across eight benchmark datasets for short text to study the effectiveness of our method. Our model outperforms state-of-the-art methods in most cases on Accuracy and NMI. This demonstrates that utilizing the MI maximization strategy during the representation learning stage could potentially be a promising tactic. Further study would be worthwhile since it might enhance the quality of textual representations for other tasks

References

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2017. A simple but tough-to-beat baseline for sentence embeddings. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Philip Bachman, R. Devon Hjelm, and William Buchwalter. 2019. Learning representations by maximizing mutual information across views. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 15509–15519.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. 2020. Unsupervised learning of visual features by contrasting cluster assignments. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder. *CoRR*, abs/1803.11175.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR.

Rotem Dror, Segev Shlomov, and Roi Reichart. 2019. Deep dominance - how to properly compare deep neural models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2773–2785, Florence, Italy. Association for Computational Linguistics.

John M. Giorgi, Osvald Nitski, Bo Wang, and Gary D. Bader. 2021. Declutr: Deep contrastive learning for unsupervised textual representations. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 879–895. Association for Computational Linguistics.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. 2020. Bootstrap your own latent - A new approach to self-supervised learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Amir Hadifar, Lucas Sterckx, Thomas Demeester, and Chris Develder. 2019. A self-training approach for short text clustering. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*. Association for Computational Linguistics.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 9726–9735. Computer Vision Foundation / IEEE.

R. Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Philip Bachman, Adam Trischler, and Yoshua Bengio. 2019. Learning deep representations by mutual information estimation and maximization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Sosuke Kobayashi. 2018. Contextual augmentation: Data augmentation by words with paradigmatic relations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 452–457. Association for Computational Linguistics.

Lingpeng Kong, Cyprien de Masson d'Autume, Lei Yu, Wang Ling, Zihang Dai, and Dani Yogatama. 2020. A mutual information maximization perspective of language representation learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Edward Ma. 2019. Nlp augmentation. https://github.com/makcedward/nlpaug.

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119.

Sosuke Nishikawa, Ryokan Ri, Ikuya Yamada, Yoshimasa Tsuruoka, and Isao Echizen. 2022. EASE: Entity-aware contrastive learning of sentence embedding. In *Proceedings of the 2022 Conference* *of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3870–3885, Seattle, United States. Association for Computational Linguistics.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. 2016. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 271–279.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch. In *NIPS 2017 Workshop on Autodiff*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543. ACL.

Xuan Hieu Phan, Minh Le Nguyen, and Susumu Horiguchi. 2008. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceedings of the 17th International Conference on World Wide Web, WWW 2008, Beijing, China, April 21-25, 2008*, pages 91–100. ACM.

Leonid Pugachev and Mikhail S. Burtsev. 2021. Short text clustering with transformers. *CoRR*, abs/2102.00541.

Md. Rashadul Hasan Rakib, Norbert Zeh, Magdalena Jankowska, and Evangelos E. Milios. 2020. Enhancement of short text clustering by iterative classification. In *Natural Language Processing and Information Systems - 25th International Conference on Applications of Natural Language to Information Systems, NLDB 2020, Saarbrücken, Germany, June 24-26, 2020, Proceedings*, volume 12089 of *Lecture Notes in Computer Science*, pages 105–117. Springer.

Nils Reimers and Iryna Gurevych. 2019a. Sentence-bert: Sentence embeddings using siamese bert-networks. *CoRR*, abs/1908.10084.

Nils Reimers and Iryna Gurevych. 2019b. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics.

Alessandro Sordoni, Nouha Dziri, Hannes Schulz, Geoffrey J. Gordon, Philip Bachman, and Remi Tachet des Combes. 2021. Decomposed mutual information estimation for contrastive representation learning. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 9859–9869. PMLR.

Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. 2020. What makes for good views for contrastive learning? In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Junyuan Xie, Ross B. Girshick, and Ali Farhadi. 2016. Unsupervised deep embedding for clustering analysis. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 478–487. JMLR.org.

Jiaming Xu, Peng Wang, Guanhua Tian, Bo Xu, Jun Zhao, Fangyuan Wang, and Hongwei Hao. 2015. Short text clustering via convolutional neural networks. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing, VS@NAACL-HLT 2015, June 5, 2015, Denver, Colorado, USA*, pages 62–69. The Association for Computational Linguistics.

Jiaming Xu, Bo Xu, Peng Wang, Suncong Zheng, Guanhua Tian, Jun Zhao, and Bo Xu. 2017. Self-taught convolutional neural networks for short text clustering. *Neural Networks*, 88:22–31.

Hui Yin, Xiangyu Song, Shuiqiao Yang, Guangyan Huang, and Jianxin Li. 2021. Representation learning for short text clustering. In *Web Information Systems Engineering - WISE 2021 - 22nd International Conference on Web Information Systems Engineering, WISE 2021, Melbourne, VIC, Australia, October 26-29, 2021, Proceedings, Part II*, volume 13081 of *Lecture Notes in Computer Science*, pages 321–335. Springer.

Jianhua Yin and Jianyong Wang. 2016. A model-based approach for text clustering with outlier detection. In *32nd IEEE International Conference on Data Engineering, ICDE 2016, Helsinki, Finland, May 16-20, 2016*, pages 625–636. IEEE Computer Society.

Dessalew Yohannes and Yeregal Assabie. 2021. Amharic text clustering using encyclopedic knowledge with neural word embedding. *CoRR*, abs/2105.00809.

Dejiao Zhang, Feng Nan, Xiaokai Wei, Shang-Wen Li, Henghui Zhu, Kathleen R. McKeown, Ramesh Nallapati, Andrew O. Arnold, and Bing Xiang. 2021. Supporting clustering with contrastive learning. *CoRR*, abs/2103.12953.

Dejiao Zhang, Wei Xiao, Henghui Zhu, Xiaofei Ma, and Andrew Arnold. 2022. Virtual augmentation supported contrastive learning of sentence representations. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 864–876, Dublin, Ireland. Association for Computational Linguistics.

Xiang Zhang and Yann LeCun. 2015. Text understanding from scratch. *CoRR*, abs/1502.01710.

Yan Zhang, Ruidan He, Zuozhu Liu, Kwan Hui Lim, and Lidong Bing. 2020. An unsupervised sentence embedding method by mutual information maximization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 1601–1610. Association for Computational Linguistics.

A Appendices

A.1 Datasets

We conduct experiments and assess the performance of our model on eight benchmark datasets for short text clustering. The important statistics of all datasets are presented in Table 2.

- **AgNews**: a subset of the English news titles dataset (Zhang and LeCun, 2015) across 4 different topics, where 2,000 samples from each topic were randomly chosen by Rakib et al. (2020).
- **SearchSnippets**: a dataset consisting of 12,340 web search snippets from 8 different categories (Phan et al., 2008).
- **Biomedical**: 20,000 paper titles, from 20 different Medical Subject Headings (MeSH), randomly selected by Xu et al. (2017) from the PubMed data distributed by BioASQ3.
- **StackOverflow**: challenge data published on Kaggle and randomly chosen by Xu et al. (2017), comprising 20,000 questions from Stack Overflow related to 20 distinct tags.
- **Tweet**: a dataset comprising 2,472 tweets with 89 groups (Yin and Wang, 2016).
- **GoogleNews**: a collection of both titles and text snippets from 11,109 news articles covering 152 events (Yin and Wang, 2016). Only the titles and the text snippet of each news article were extracted out of the GoogleNews-TS to produce GoogleNews-T and GoogleNews-S, respectively.

We spend up to 14 GPU hours on a Tesla V100 32G GPU to complete the training on all datasets for each MIST model's configuration.

Dataset	$N^{Cluster}$	N^{Doc}	N^{Word}
AgNews	4	8,000	23
SearchSnippets	8	12,340	18
Biomedical	20	20,000	13
StackOverflow	20	20,000	8
Tweet	89	2,472	8
Googlenews-TS	152	11,109	28
Googlenews-T	152	11,109	6
Googlenews-S	152	11,109	22

Table 2: Dataset statistics. $N^{Cluster}$: number of clusters; N^{Doc}: number of short text documents; N^{Word} : average number of words in each document

A.2 The Effects of Sequence- and Token-MI Maximization Objectives on NMI

Figure 4 shows the effects of the MI objective on NMI. It follows the same trend as Accuracy as discussed in Section 4.3.1.

A.3 Positive Pairs in Contrastive Learning

It is a common practice in contrastive learning frameworks to only consider augmented texts as inputs, excluding original samples. However, we adopt a different input scheme. We discovered that feeding both original and augmented samples into our representation learning framework (as shown in Figure 1) yields better clustering results than exclusively taking two augmented texts as an input pair. One plausible reason is that when augmented texts are generated, the augmenter replaces some keywords in the original texts with new words. Short texts frequently have few keywords; hence, the absence of crucial words required for text categorization impacts clustering performance.

A.4 The Analysis of the Clustering Objective

As discussed in Section 4.3.2, the clustering performance is substantially affected by the weight of the clustering objective. Table 3 presents the performance of MIST across eight datasets in three situations, i.e., the coefficient of the clustering objective, η, in Eq.1 is assigned to 0, 1, and 2. The optimal results for the majority in terms of ACC and NMI are produced when η is set to 2.

A.5 Exploration of Data Augmentations

According to Zhang et al. (2021), which has studied the impacts of data augmentation in extensive details and the *Contextual Augmenter* has shown that it substantially outperforms other augmenters in their study. They hypothesized that since both

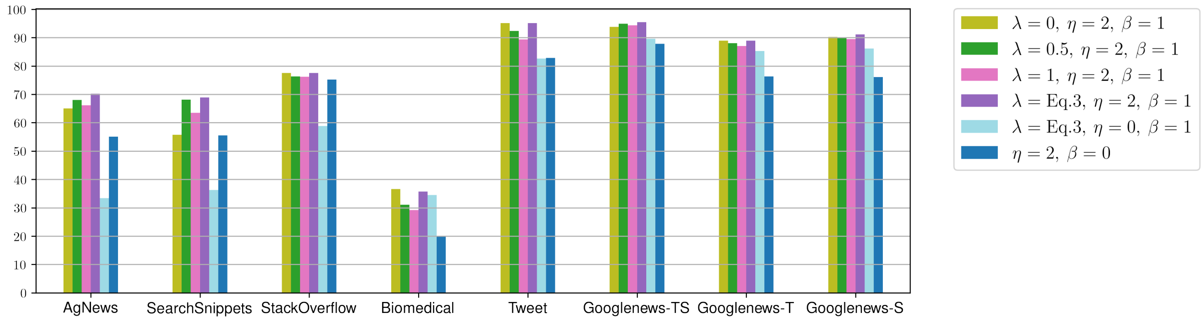


Figure 4: NMI for six different settings including four different weighting ratios between sequence-level and token-level MI maximization objectives. As well as, a setting where a clustering loss is absent ($\eta = 0$), and a setting where an MI loss is absent ($\beta = 0$). Note that when we set β to 0, λ has no effect.

	AgNews		**SearchSnippets**		**StackOverflow**		**Biomedical**	
	ACC	**NMI**	**ACC**	**NMI**	**ACC**	**NMI**	**ACC**	**NMI**
MIST w/ $\eta = 0$	56.96	33.40	50.30	36.30	64.40	58.80	43.26	34.55
MIST w/ $\eta = 1$	81.40	57.39	70.99	56.90	76.41	71.92	47.66	40.34
MIST w/ $\eta = 2$	89.47	70.25	76.72	67.69	78.74	77.59	39.15	34.66

	Tweet		**GoogleNewsTS**		**GoogleNewsT**		**GoogleNewsS**	
	ACC	**NMI**	**ACC**	**NMI**	**ACC**	**NMI**	**ACC**	**NMI**
MIST w/ $\eta = 0$	56.27	82.64	68.89	89.59	62.85	85.28	65.74	86.16
MIST w/ $\eta = 1$	64.46	86.27	74.86	91.89	66.91	87.04	71.98	88.58
MIST w/ $\eta = 2$	91.75	95.12	89.93	95.47	75.97	88.97	81.91	90.79

Table 3: The clustering results of MIST on three different weights of the clustering objective, η.

the *Contextual Augmenter* and their encoder use the pretrained transformers as the backbones, this allows the *Contextual Augmenter* to produce augmentations that are more informative. Our encoder is also a pretrained transformer, and we observed that the experimental results followed the same trend as Zhang et al. (2021). We thus employ this augmenter in our experiments.

In this section, we investigate the impact of the *Contextual Augmenter* configurations in terms of masked language models and word substitution ratios. As shown in Table 4, we found that MIST using augmented texts generated from the BERT model with 20% substitution rate yields the best overall performance. Interestingly, MIST with augmented texts produced by other encoders with a 20% substitution rate also yields outcomes close to those of BERT with the same substitution rate.

A.6 SCCL Reimplementation

To thoroughly compare the performance of our representation learning strategy against SCCL (Zhang et al., 2021), an existing contrastive learning method for short text clustering, we reproduced SCCL in both an end-to-end version (SCCL) and a multiple-stage version (SCCL-Multi), by applying the *k*-means algorithm on top of SCCL representations to make their pipeline identical to our framework. In the reference paper, SCCL considers *Contextual Augmenter* with three configurations by setting the word substitution ratio of each text instance to 10%, 20%, and 30%. However, their study does not identify which setting produces the best outcomes. Therefore, we evaluate both reproduced versions of SCCL using three alternative masked language models: BERT-base, RoBERTa, and DistilBERT, with the aforementioned word substitution ratios for augmented pair generation to cover all scenarios reported in their study.

Table 5 reports the clustering performance of SCCL in both reproduced versions and in all configurations. Note that, the results were collected by choosing the best Accuracy and NMI throughout 3000 iterations. The percentage of word replacement and masked language models employed for augmented text generation have an impact on the clustering performance, since the best setting for these two parameters varies across datasets. Our method with the setup described in Section 4 outperforms SCCL and SCCL-Multi with the best parameter settings in the majority of cases.

	AgNews		SearchSnippets		StackOverflow		Biomedical	
	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
MIST w/ BERT 10%	87.74	66.99	75.98	67.71	77.78	76.42	37.51	33.97
MIST w/ BERT 20%	89.47	70.25	76.72	67.69	78.74	77.59	39.15	34.66
MIST w/ BERT 30%	86.33	66.09	81.46	67.71	73.60	71.55	39.79	34.61
MIST w/ RoBERTa 10%	87.51	66.81	75.64	67.11	77.84	76.50	38.61	35.11
MIST w/ RoBERTa 20%	88.85	69.12	76.21	68.52	77.74	76.41	37.17	31.62
MIST w/ RoBERTa 30%	86.43	66.4	73.77	65.72	77.76	77.03	29.48	27.38
MIST w/ DistilBERT 10%	87.22	66.44	74.96	65.89	77.67	76.30	38.29	34.29
MIST w/ DistilBERT 20%	89.42	70.26	75.74	67.85	77.72	77.05	38.29	32.31
MIST w/ DistilBERT 30%	87.96	67.66	74.23	64.11	77.67	76.34	38.83	34.63

	Tweet		GoogleNews-TS		GoogleNews-T		GoogleNews-S	
	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
MIST w/ BERT 10%	88.76	93.04	86.65	94.76	72.41	87.99	76.56	89.3
MIST w/ BERT 20%	91.75	95.12	89.93	95.47	75.97	88.97	81.91	90.79
MIST w/ BERT 30%	90.07	94.14	89.28	94.98	75.63	88.55	80.74	89.99
MIST w/ RoBERTa 10%	88.18	92.64	85.85	94.48	73.68	88.00	77.89	89.52
MIST w/ RoBERTa 20%	90.97	94.67	90.10	95.35	74.61	88.27	77.62	90.00
MIST w/ RoBERTa 30%	83.40	95.15	88.29	96.20	70.27	88.24	78.43	89.82
MIST w/ DistillBERT 10%	85.48	92.24	85.15	94.42	75.89	88.51	77.55	89.69
MIST w/ DistillBERT 20%	91.24	94.99	90.16	95.43	74.14	88.53	82.54	90.69
MIST w/ DistillBERT 30%	86.56	92.50	85.85	94.46	75.57	88.50	77.18	89.52

Table 4: The clustering performance of MIST when feeding augmented texts generated by Contextual Augmenter as inputs across nine different configurations.

A.7 Limitations

Despite the state-of-the-art performance, there are some limitations, which we highlight in this section. Firstly, the encoder of our model is pretrained using general domain data. Hence, when our model encounters short texts in a specific domain, such as Biomedical, the performance drops drastically. Furthermore, our representation learning procedure performs poorly on short texts with only content words, especially when some terms are shared across multiple clusters, or short texts contain incoherent text sequences. In particular, learning representations for lengthy texts with incoherent phrases, by incorporating token-level MI maximization objective alongside the sequence-level MI maximization, forces a sequence representation to resemble each individual token embedding. The token-level MI maximization objective provides no further improvement in this case. This constraint should be taken into account in future research.

Another limitation of our framework is that, according to the general operation of contrastive learning, augmented samples are crucial for learning representations. However, the best augmentation strategy is still a subject of discussion and exploration. A study in Zhang et al. (2021) and our own experiments with various augmentation settings show that changing an augmenter as well as adjusting the configuration parameters both affect the performance of clustering. Additionally, even if the augmenter and the parameters used to generate augmented texts are exactly the same, there is a possibility that the outcomes from the two trials may vary, adding a variance to the performance results.

	AgNews		SearchSnippets		StackOverflow		Biomedical	
	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
SCCL (in the reference paper)	88.20	68.20	85.20	71.10	75.50	74.50	46.20	41.50
SCCL w/ BERT 10%	87.20	66.94	83.70	70.05	71.40	71.28	46.00	40.06
SCCL-Multi w/ BERT 10%	87.2	66.94	83.40	69.88	77.30	73.76	46.00	40.13
SCCL w/ BERT 20%	87.10	66.91	84.40	69.58	64.20	56.23	46.40	40.39
SCCL-Multi w/ BERT 20%	87.10	66.80	83.60	69.28	60.02	52.22	45.50	40.07
SCCL w/ BERT 30%	87.50	67.46	83.70	68.54	60.70	52.18	42.40	38.14
SCCL-Multi w/ BERT 30%	87.50	67.45	82.60	66.45	60.90	52.29	42.30	37.95
SCCL w/ RoBERTa 10%	87.00	66.57	84.50	70.21	62.10	54.26	28.50	20.35
SCCL-Multi w/ RoBERTa 10%	87.00	66.55	84.10	70.14	61.40	53.05	28.50	20.34
SCCL w/ RoBERTa 20%	85.20	64.20	62.60	41.66	60.70	52.26	39.60	32.66
SCCL-Multi w/ RoBERTa 20%	85.10	64.24	72.00	51.23	60.09	52.31	38.40	38.40
SCCL w/ RoBERTa 30%	84.00	62.24	30.70	10.07	60.70	52.28	39.10	32.77
SCCL-Multi w/ RoBERTa 30%	84.00	62.26	30.70	10.05	60.90	52.44	39.50	32.63
SCCL w/ DistilBERT 10%	87.30	67.16	84.70	70.79	70.20	69.49	46.10	39.87
SCCL-Multi w/ DistilBERT 10%	87.30	67.16	84.50	70.64	72.10	68.20	46.20	39.92
SCCL w/ DistilBERT 20%	86.80	65.87	84.70	70.62	71.40	69.38	46.30	39.94
SCCL-Multi w/ DistilBERT 20%	86.80	65.87	84.20	70.45	72.20	70.84	46.40	40.01
SCCL w/ DistilBERT 30%	87.20	66.77	85.00	71.63	70.80	70.04	46.30	40.49
SCCL-Multi w/ DistilBERT 30%	87.20	66.75	84.60	71.35	76.50	72.57	46.40	40.58

	Tweet		GoogleNews-TS		GoogleNews-T		GoogleNews-S	
	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
SCCL (in the reference paper)	78.20	89.20	89.80	94.90	75.80	88.30	83.10	90.40
SCCL w/ BERT 10%	56.80	81.91	70.10	89.49	62.50	81.53	69.00	86.29
SCCL-Multi w/ BERT 10%	75.30	88.39	86.70	93.95	76.30	88.25	81.00	89.82
SCCL w/ BERT 20%	57.10	82.54	75.60	90.99	63.00	81.72	67.80	85.97
SCCL-Multi w/ BERT 20%	78.20	89.41	88.70	94.70	76.20	87.97	81.10	89.60
SCCL w/ BERT 30%	56.6	82.23	74.2	90.83	61.30	81.20	64.9	89.78
SCCL-Multi w/ BERT 30%	78.80	89.58	89.90	94.91	75.60	87.88	82.10	89.77
SCCL w/ RoBERTa 10%	56.00	79.89	73.60	90.46	55.60	78.08	65.50	85.26
SCCL-Multi w/ RoBERTa 10%	71.10	85.86	86.60	93.94	56.90	78.52	80.50	89.50
SCCL w/ RoBERTa 20%	56.80	79.56	74.90	90.37	55.60	78.08	66.90	85.38
SCCL-Multi w/ RoBERTa 20%	74.20	86.61	88.10	94.27	58.40	79.28	81.30	89.87
SCCL w/ RoBERTa 30%	53.80	78.47	71.80	71.80	55.60	78.42	65.30	83.99
SCCL-Multi w/ RoBERTa 30%	63.60	76.98	85.20	93.53	56.60	78.42	78.00	88.14
SCCL w/ DistilBERT 10%	56.10	80.87	72.70	90.03	61.40	80.94	69.60	85.81
SCCL-Multi w/ DistilBERT 10%	78.80	88.91	87.70	94.25	74.30	87.78	79.70	89.20
SCCL w/ DistilBERT 20%	56.40	80.28	71.70	90.04	61.30	81.19	67.70	86.02
SCCL-Multi w/ DistilBERT 20%	77.10	88.61	86.50	94.03	75.10	87.51	79.50	89.70
SCCL w/ DistilBERT 30%	56.60	81.65	72.10	90.18	62.00	81.09	66.50	85.48
SCCL-Multi w/ DistilBERT 30%	76.00	88.39	88.50	94.18	75.80	87.60	79.10	89.01

Table 5: The clustering performances of the reimplemented SCCL and SCCL-Multi with nine different configurations for Contextual Augmenter. These configurations are obtained by setting the word substitution ratio of each text instance to 10% , 20%, and 30%, as well as using three alternative masked language models: BERT-base, RoBERTa, and DistilBERT.