Multi-Surrogate-Objective Optimization for Neural Topic Models

Tue Le[1*], Hoang Tran Vuong[1*], Tung Nguyen[1†], Linh Ngo Van[1], Sang Dinh[1], Trung Le[2], Thien Huu Nguyen[3]
[1]Hanoi University of Science and Technology (HUST), Vietnam
[2] Monash University, Australia
[3]University of Oregon, USA

Abstract

Neural topic modeling has substantially improved topic quality and document topic distribution compared to traditional probabilistic methods. These models often incorporate multiple loss functions. However, the disparate magnitudes of these losses can make hyperparameter tuning for these loss functions challenging, potentially creating obstacles for simultaneous optimization. While gradient-based Multi-objective Optimization (MOO) algorithms offer a potential solution, they are typically applied to shared parameters in multi-task learning, hindering their broader adoption, particularly in Neural Topic Models (NTMs). Furthermore, our experiments reveal that naïve MOO applications on NTMs can yield suboptimal results, even underperforming compared to implementations without the MOO mechanism. This paper proposes a novel approach to integrate MOO algorithms, independent of hard-parameter sharing architectures, and effectively optimizes multiple NTMs loss functions. Comprehensive evaluations on widely used benchmark datasets demonstrate that our approach significantly enhances baseline topic model performance and outperforms direct MOO applications on NTMs.

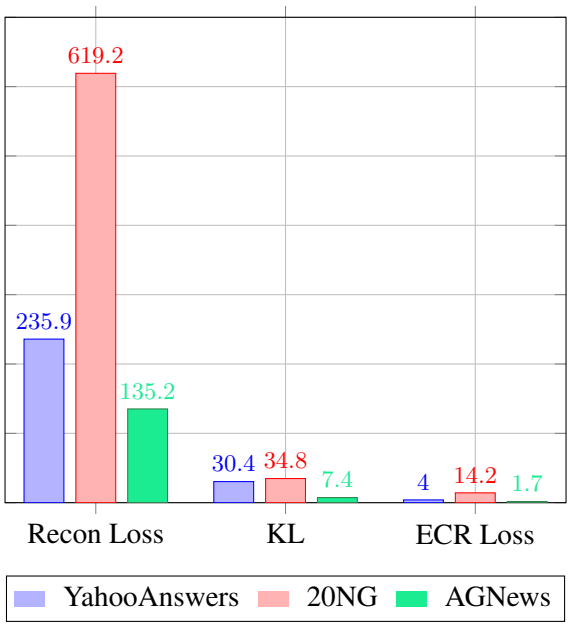


Figure 1: Illustration of the magnitudes of training loss components. We recorded the training losses in the final iteration when training ECRTM on three datasets: YahooAnswers, 20NG, and AGNews. These losses include Reconstruction Loss, Kullback-Leibler Divergence, and Embedding Clustering Regularization Loss, which exhibit significant differences in magnitude.

1 Introduction

Traditional topic models (Hofmann, 1999; Blei et al., 2003; Blei and Lafferty, 2005; Wang et al., 2008) have become widely used due to their ability to uncover hidden topics from unstructured data and their interpretability. Although such models have achieved some success, they often suffer from inefficient and laborious parameter inference (Wu et al., 2024). To address these limitations, neural topic models (NTMs) have emerged as a promising solution, achieving effectiveness in various domains such as content organization (Valero et al., 2022), text mining (Linh et al., 2017; Valero et al., 2022; Ha et al., 2019), health research (Gao and Sazara, 2023) and streaming learning (Nguyen et al., 2019; Van Linh et al., 2022; Tuan et al., 2020; Nguyen et al., 2021).

Based on Variational Autoencoders (VAEs) (Kingma and Welling, 2013a), neural topic models (Dieng et al., 2020; Wu et al., 2023; Pham et al., 2024; Nguyen et al., 2025b; Vuong et al., 2025; Nguyen et al., 2025e) enhance traditional topic modeling techniques (Hofmann, 1999; Griffiths et al., 2003) by utilizing the power of neural network architectures. To significantly improve both the quality of discovered topics and the effectiveness of document representations,

*These authors contributed equally to this work.
†Corresponding author: tungns@soict.hust.edu.vn

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 135–151
November 4-9, 2025

most neural topic models typically optimize multiple loss functions simultaneously (e.g., reconstruction loss (Wu et al., 2023), KL divergence (Pham et al., 2024), contrastive loss (Nguyen and Luu, 2021), Wasserstein distance (Nguyen et al., 2025a)). Therefore, they may encounter challenge in optimizing multiple objective functions at the same time.

The substantial disparities in the magnitudes of these loss components, as evident in Figure 1, intuitively suggest that such an imbalance could lead to certain objectives dominating the training process, consequently degrading overall performance. Therefore, a mechanism to regulate the influence of each loss function during training is extremely necessary. A solution to balance this trade-off involves assigning fixed linear weights to each loss function (Rybkin et al., 2021). However, this method necessitates an exhaustive and computationally expensive manual hyperparameter search. Moreover, the magnitudes of these loss functions can fluctuate unpredictably during training, rendering pre-defined weights potentially ineffective. Furthermore, if the objectives are conflicting, a simple linear combination may not suffice (Liang et al., 2021; Mahapatra and Rajan, 2020; Nguyen et al., 2024, 2025d). Instead, we propose framing the training of NTMs as a multi-objective optimization (MOO) problem. This optimization considers the gradients of the individual objectives to achieve a Pareto stationary solution, attaining an optimal balance among them. However, most gradient-based MOO algorithms are designed for hard-parameter sharing architectures. Modern models, including NTMs, may deviate from this architectural constraint.

In earlier work, (Nguyen et al., 2024) applied MOO to contrastive topic modeling, comparing methods such as GradNorm (Chen et al., 2018), PCGrad (Yu et al., 2020), Random Weighting (Lin et al., 2022), and MGDA (Sener and Koltun, 2018). However, except for MGDA, these methods generally did not yield improvements and, in some cases, even underperformed compared to baselines without MOO. Even with MGDA, the improvements were not particularly significant. This led us to hypothesize that a direct and naïve application of MOO algorithms to NTMs might not be universally effective. To validate this hypothesis, we conducted experiments with a range of state-of-the-art MOO algorithms and NTM architectures. As shown in tables 1 and 2, our findings indicate that this direct approach often produces negligible improvements, and can sometimes even degrade performance compared to the baseline. Several factors could contribute to this phenomenon: first, the loss imbalance issue discussed earlier; and second, the architectural differences between modern Neural Topic Models (NTMs) and the hard-parameter sharing paradigm commonly used in many MOO algorithms.

To address the first issue, we propose constructing surrogate losses, as presented in Equation 3, which are linear combinations of the original loss functions. Instead of directly optimizing the original objectives by MOO algorithm, we apply them to these newly surrogate losses. These surrogate losses are designed to exhibit more balanced magnitudes and are influenced not only by the original loss components but also by the overall loss. Specifically, our approach aligns well with the principles of ensemble learning, where each surrogate loss is associated with a learner, and multiple learners collaborate to enhance overall performance. The MOO mechanism aggregates the gradients of learners to update the model parameters, ensuring that all learners perform well and no surrogate loss increases during the ensemble process. Regarding the second issue, with this approach, each surrogate loss function depends on the model's entire parameters. Therefore, neural topic models can easily satisfy hard-parameter sharing architectures.

We summarize the contributions of this paper as follows:

- Comprehensive experiments with direct MOO approaches indicate their limitations on neural topic models.
- We propose a novel optimization method called **MSOO** (Multi-Surrogate-Objective Optimization), which introduces new surrogate objectives and applies Multi-objective Optimization algorithms to these objectives.
- Empirical evidence shows that MSOO not only improves baseline neural topic models performance but also outperforms prior direct MOO approaches.

2 Background

Denote $\mathbf{X} = \{\mathbf{x}_d\}_{d=1}^{D}$ represent the Bag-of-Words (BoW) vectors for D documents, based on a vocabulary containing V words. The objective of neural topic modeling targets to discover K latent topics. We have $\beta = (\beta_1, \ldots, \beta_K) \in \mathbb{R}^{V \times K}$ is

the topic-word distribution matrix of all K topics. Given word embedding dimension L, we define $\mathbf{W} = (\mathbf{w}_1, \ldots, \mathbf{w}_V) \in \mathbb{R}^{V \times L}$ as the word embedding matrix and $\mathbf{T} = (\mathbf{t}_1, \ldots, \mathbf{t}_K) \in \mathbb{R}^{K \times L}$ as the topic embedding matrix. Recent state-of-the-art neural topic models (Wu et al., 2023; Pham et al., 2024; Vu et al., 2025; Nguyen et al., 2025c) no longer decompose β into two components: word embeddings W and topic embeddings T; but instead represent β as:

$$\beta_{ij} = \frac{\exp\left(-\|\mathbf{w}_i - \mathbf{t}_j\|^2/\tau\right)}{\sum_{j'=1}^{K} \exp\left(-\|\mathbf{w}_i - \mathbf{t}_{j'}\|^2/\tau\right)},$$

where τ is a temperature hyperparameter. For each document d, another objective of neural topic models is to estimate the topic proportions $\theta_d \in \mathbb{R}^K$, indicating the distribution of topics present in the document. Specifically, the topic proportion θ is dependent on a latent variable z, which follows a logistic-normal distribution defined as $p(z) = \mathcal{N}(z|\mu_0, \sigma_0)$. Then the BoW representation of a document x_d is encoded through neural networks. The parameters of the Gaussian distribution are computed with mean $\mu = h_\mu(x_d, \gamma)$ and the diagonal covariance matrix is $\Sigma = \text{diag}(h_\Sigma(x_d, \gamma))$, where γ is the parameter of these inference networks.

Latent variable z is subsequently sampled from the posterior distribution $q(z|x_d) = \mathcal{N}(z|\mu, \Sigma)$ employing the reparameterization trick proposed by (Kingma and Welling, 2013b). The topic proportions are then derived by using the softmax function, resulting in $\theta = \text{softmax}(z)$. Next, BoW representation is reconstructed through topic-word distribution matrix β and topic proportion θ from a multinomial distribution: $\hat{\mathbf{x}}_{\text{BoW}} \sim \text{Multi}(\text{softmax}(\beta\theta))$. Finally, the objective function for the topic model comprises two components: a reconstruction term and a regularization term, defined as follows:

$$\mathcal{L}_{\text{TM}} = \frac{1}{D}\sum_{i=1}^{D}\Big[-(\mathbf{x}_{i\text{BoW}})^\top \log(\text{softmax}(\beta\theta_i)) + \text{KL}(q(z|\mathbf{x}_i)\|p(z))\Big] \quad (1)$$

Recent modern neural topic models frequently integrate multiple objective functions into their overall objective to enhance both the performance of topic quality and document representations. For instance, ECRTM (Wu et al., 2023) adds the Embedding Clustering Regularization objective $\mathcal{L}_{\text{ECR}}$ to effectively mitigate topic collapsing. NeuroMax (Pham et al., 2024) adds three objectives: Embedding Clustering Regularization $\mathcal{L}_{\text{ECR}}$, Group Topic Regularization $\mathcal{L}_{\text{GR}}$, and Information Noise-Contrastive Estimation $\mathcal{L}_{\text{InfoNCE}}$.

To effectively manage these multiple objectives, gradient-based Multi-objective Optimization (MOO) methods are employed. Several notable frameworks, including MGDA (Sener and Koltun, 2018), PCGrad (Yu et al., 2020), IMTL (Liu et al., 2021b), FairGrad (Ban and Ji, 2024), and ExcessMTL (He et al., 2024), propose gradient-based MTL methods aimed at finding solutions on the Pareto front. Let K represent the total number of tasks. We define δ^{share} as the shared parameters while $\delta^1, \delta^2, \ldots, \delta^K$ denote task-specific parameters. These approaches commonly seek the update direction as a linear combination of individual task gradients:

$$\Delta\delta^{\text{share}} = \sum_{i=1}^{K} w_i \nabla_{\delta^{\text{share}}} \mathcal{L}_i(\delta^{\text{share}}, \delta^i) \quad (2)$$

where $\mathcal{L}_i(\delta^{\text{share}}, \delta^i)$ denotes the loss for i-th task and w represents a dynamic weighting vector that adapts based on the model's current state at each optimization step. The primary difference among these methods lies in the strategy used to select w. More details regarding MOO can be found in Appendix A.

3 Methodology

3.1 Redefining Objectives for Multi-objective Optimization

Multi-objective Optimization methods are widely recognized as effective techniques for optimization of multiple loss functions simultaneously. As discussed earlier, with each loss function $\mathcal{L}_i$ contributing its gradient $\mathbf{g}_i = \nabla\mathcal{L}_i$, direct MOO approaches directly operate on these individual gradients. However, these gradients are inherently objective-specific and lack awareness of the general objective. Furthermore, in NTMs, the varying magnitudes of the loss components can lead to certain objectives exhibiting a significantly higher loss decreasing rate compared to others. This imbalance may hinder the optimization of less dominant objectives, making it difficult to learn the dynamic

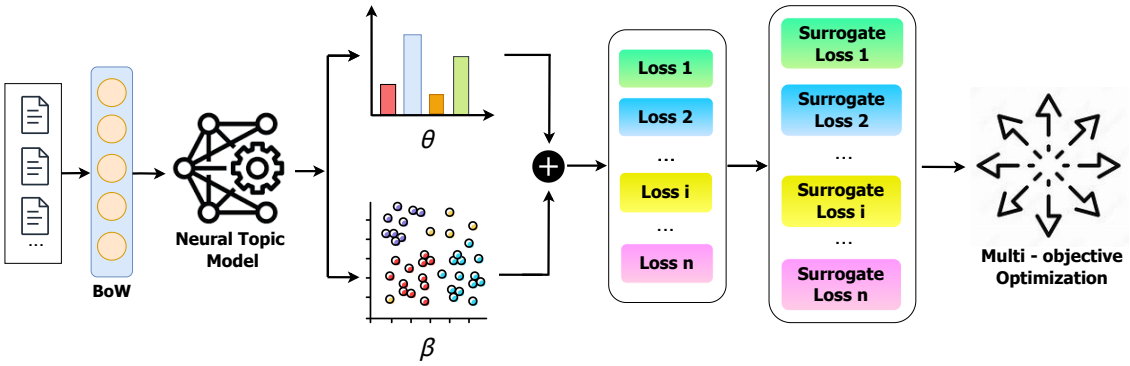


Figure 2: The overall workflow of MSOO when applied to topic models.

weighting vector of the objectives. To address these limitations, we propose new surrogate objectives that not only focus on the corresponding objective $\mathcal{L}_i$ but also incorporate overall information from other objectives, formulated as follows:

$$\mathcal{L}_i^{\text{MSOO}} = \mathcal{L}_i + \lambda_i \sum_{j=1}^{n} \mathcal{L}_j, \tag{3}$$

where $\lambda_i > 0$ governs the influence of the aggregated objectives $\sum_{j=1}^{n} \mathcal{L}_j$ with corresponding objective. It acts as a balancing mechanism, ensuring that individual objectives contribute meaningfully while aligning with global optimization goals. The specific strategy for selecting the weighting coefficients $(\lambda_1, \lambda_2, ..., \lambda_n)$ is detailed in Section 3.2. The complete workflow of MSOO, as applied to standard topic models, is illustrated in Figure 2. The algorithm is described in detail in Algorithm 1.

3.2 Multi-Surrogate-Objective Optimization for Topic Model

The proposed surrogate losses are particularly impactful for topic models, where balancing multiple objectives is essential for achieving high-quality topic representations. The choice of λ can affect the performance of the model. In this section, we provide a detailed discussion on the selection strategies for λ and the integration of our approach into topic modeling. Specifically, we introduce two methodologies for selecting λ. The first adopts a straightforward approach where λ is treated as a constant, while the second explores an adaptive strategy to dynamically adjust λ during the training process. To provide a clearer explanation, we outline the specifics of how our approach is applied to ECRTM (Wu et al., 2023) which has three different losses corresponding to Reconstruction Loss (Recon Loss), Kullback-Leibler Divergence (KL), and Embedding Clustering Regularization Loss (ECR Loss), with the MOO algorithm is FairGrad (Ban and Ji, 2024).

3.2.1 Static Multi-Surrogate-Objective Optimization

With this strategy, we simply consider $\lambda_i = \lambda, \ \forall \ i \in \{1, 2, ..., n\}$; where λ is a hyperparameter. These surrogate objectives can be formulated as:

$$\begin{cases} \mathcal{L}_{\text{Recon}}^{\text{MSOO}} = \mathcal{L}_{\text{Recon}} + \lambda \mathcal{L}_{\text{Total}} \\ \mathcal{L}_{\text{KL}}^{\text{MSOO}} = \mathcal{L}_{\text{KL}} + \lambda \mathcal{L}_{\text{Total}} \\ \mathcal{L}_{\text{ECR}}^{\text{MSOO}} = \mathcal{L}_{\text{ECR}} + \lambda \mathcal{L}_{\text{Total}} \end{cases} \tag{4}$$

where $\mathcal{L}_{\text{Total}} = \mathcal{L}_{\text{Recon}} + \mathcal{L}_{\text{KL}} + \mathcal{L}_{\text{ECR}}$. Now, instead of directly optimizing the original losses, we feed these surrogate objectives into a Multi-objective Optimization algorithm. Let δ represent the model parameters. For each surrogate objective $\mathcal{L}_i^{\text{MSOO}}$, we can calculate its gradient with respect to the model parameters, which we call $g_i^{\text{MSOO}} = \nabla_\delta \mathcal{L}_i^{\text{MSOO}}$. Think of this gradient as a direction in the parameter space that, if followed, we would reduce the corresponding surrogate loss. Our goal is to find a single update direction, d, that improves all surrogate objectives simultaneously. This direction should lie within a small region around the current parameters, which we define as a ball B_ϵ of radius ϵ. This constraint ensures that we do not make drastic changes to the parameters in a single step, promoting stable optimization. To find this direction d, we adopt a utility function called the "α-fair" utility function, inspired by fair

Algorithm 1 Static Multi-Surrogate-Objective Optimization algorithm

Input: Model parameters δ, learning rate η, total number of training epochs N, hyperparameter λ
Output: Updated model parameters δ

for $t = 1, 2, \ldots, N$ **do**
  Compute original objectives $\{\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_n\}$
  Compute surrogate objectives $\mathcal{L}_{\text{i}}^{\text{MSOO}}$ with $\lambda_i = \lambda$ using Equation 3; for $i = 1, 2, \ldots, n$
  Compute gradients $g_i^{\text{MSOO}} = \nabla \mathcal{L}_{\text{i}}^{\text{MSOO}}$; for $i = 1, 2, \ldots, n$
  Calculate $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ by:
    $\alpha = \text{MOO\_algorithm}(g_1^{\text{MSOO}}, \ldots, g_n^{\text{MSOO}})$
  Calculate $d_t = \sum_{i=1}^{n} \alpha_i g_i^{\text{MSOO}}$
  Update the parameters: $\delta_{t+1} = \delta_t - \eta d_t$
end for

resource allocation in communication networks, as in (Ban and Ji, 2024). This utility function helps us balance the improvement across different objectives fairly. Formally, we want to find d that maximizes the following:

$$\max_{d \in B_\epsilon} \sum_{i=1}^{n} \frac{((g_i^{\text{MSOO}})^\top d)^{1-\alpha}}{1-\alpha} \quad \text{s.t. } (g_i^{\text{MSOO}})^\top d \geq 0 \tag{5}$$

Here, $(g_i^{\text{MSOO}})^\top d$ represents the improvement along the direction d for the i-th surrogate objective, and $\alpha \in [0, 1) \cup (1, +\infty)$ controls the different ideas of fairness. The constraint $(g_i^{\text{MSOO}})^\top d \geq 0$ ensures that the direction d improves (or at least does not worsen) each surrogate objective. Following (Ban and Ji, 2024)'s methodology, we can indicate that the best direction d^* lies on the edge of our small region B_ϵ. This best direction also aligns with the gradient of the overall α-fair utility. Mathematically, this means:

$$\sum_{i=1}^{n} ((g_i^{\text{MSOO}})^\top d)^{-\alpha} g_i^{\text{MSOO}} = cd \tag{6}$$

for some positive constant c. For simplicity, we set $c = 1$ (as in (Ban and Ji, 2024)). Now, we can express this optimal direction d as a weighted combination of the individual surrogate gradients: $d = \sum_{i=1}^{n} w_i g_i^{\text{MSOO}}$. The weights w_i determine how much each surrogate objective contributes to the final update direction. Specifically, by substituting $d = \sum_{i=1}^{n} w_i g_i^{\text{MSOO}}$ into Equation 6 and setting $c = 1$, we obtain:

$$G^\top G w = w^{-1/\alpha} \tag{7}$$

Algorithm 2 Adaptive Multi-Surrogate-Objective Optimization algorithm

Input: Model parameters δ, learning rate η, total number of training epochs N
Output: Updated model parameters δ

for $t = 1, 2, \ldots, N$ **do**
  Compute original objectives $\{\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_n\}$
  if $t = 1, 2$ **then**
    $v_i(t) = 1$; for $i = 1, 2, \ldots, n$
  else
    Compute $v_i(t)$ using Equation 9; for $i = 1, 2, \ldots, n$
  end if
  Compute $\lambda_i(t)$ using Equation 9; for $i = 1, 2, \ldots, n$
  Compute surrogate objectives $\mathcal{L}_{\text{i}}^{\text{MSOO}}$ with $\lambda_i = \lambda_i(t)$ using Equation 3; for $i = 1, 2, \ldots, n$
  Compute gradients $g_i^{\text{MSOO}} = \nabla \mathcal{L}_{\text{i}}^{\text{MSOO}}$; for $i = 1, 2, \ldots, n$
  Calculate $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ by:
    $\alpha = \text{MOO\_algorithm}(g_1^{\text{MSOO}}, \ldots, g_n^{\text{MSOO}})$
  Calculate $d_t = \sum_{i=1}^{n} \alpha_i g_i^{\text{MSOO}}$
  Update the parameters: $\delta_{t+1} = \delta_t - \eta d_t$
end for

where $G = \left[g_1^{\text{MSOO}}, \cdots, g_n^{\text{MSOO}}\right]$ is a matrix whose columns g_i^{MSOO} represent the gradients of the i-th surrogate objective, and $w := (w_1, \cdots, w_n)^\top \in \mathbb{R}_+^n$ denotes the set of weights. We address Equation 7 directly as a constrained nonlinear least squares problem. Specifically, we aim to find w that minimize the following objective function:

$$\begin{aligned} \min_{w} \quad & \sum_{i} f(w)_i^2 \\ \text{s.t.} \quad & f(w) = G^\top G w - w^{-1/\alpha},\ w \in \mathbb{R}_+^n \end{aligned} \tag{8}$$

By updating our model parameters δ using this direction d, we take a step towards optimizing all objectives simultaneously, guided by our surrogate losses and the principle of α-fairness. This method helps us avoid situations where some objectives dominate the optimization process, leading to a more balanced and overall better solution.

3.2.2 Adaptive Multi-Surrogate-Objective Optimization

Treating λ as a hyperparameter and selecting its value appropriately often requires both experience and extensive experimentation. In this section, we introduce a variant of MSOO that adaptively selects λ instead of fixing it. We call this variant MSOO-Adaptive (MSOO-A). Intuitively, if in a given training iteration, the i-th loss function, $\mathcal{L}_i$, demonstrates a more substantial decrease compared to other loss components, it implies that the relative

priority of $\mathcal{L}_i$ should be reduced in the subsequent iteration. We propose a simple yet effective adaptive weighting method inspired by Dynamic Weight Averaging (DWA) (Liu et al., 2019). For each original loss function $\mathcal{L}_i$ and surrogate loss function $\mathcal{L}_i^{\text{MSOO}}$, we choose λ_i as an adaptive weighting over time by continuously recalibrating λ_i based on the rate of change of loss from one iteration to the next. Specifically, our formula is as follows:

$$\lambda_i(t) := \frac{\exp(v_i(t-1)/T)}{\sum_{j=1}^{n} \exp(v_j(t-1)/T)}, \quad \text{where} \quad v_i(t-1) = \frac{\mathcal{L}_i(t-2)}{\mathcal{L}_i(t-1)}. \tag{9}$$

where t is an iteration index, T represents a temperature which controls the softness of λ_i and $\mathcal{L}_i(t)$ is the value of i-th loss function $\mathcal{L}_i$ at iteration t. For $t = 1, 2$, we initialize $v_i(t) = 1$. For DWA, at each iteration, the weight coefficient $\lambda_i(t)$ for the loss term $\mathcal{L}_i(t)$ is determined by the ratio of the losses for the corresponding objective across the previous two iterations, $t-1$ and $t-2$. Specifically, if the weight $v_i(t-1)$ is higher relative to other weights $v_j(t-1)$ for $j \neq i$, it indicates that at iteration $t-1$ the model prioritizes $\mathcal{L}_i$ more heavily compared to other components $\mathcal{L}_j$. In contrast, our method does not seek the coefficient $\lambda_i(t)$ for $\mathcal{L}_i(t)$, but rather adjusts the weight of the total loss term $\mathcal{L}_{\text{Total}}$ within the i-th surrogate loss. More concretely, if at iteration $t-1$ the model prioritizes $\mathcal{L}_i(t)$, then in the surrogate loss $\mathcal{L}_i^{\text{MSOO}}$, the weight of the $\mathcal{L}_{\text{Total}}$ component should be increased. Consequently, in terms of underlying semantics, the determination of $v_i(t-1)$ in our method calculates the relative ascending rate, i.e., $v_i(t-1) = \mathcal{L}_i(t-2)/\mathcal{L}_i(t-1)$. Finally, these surrogate objectives at iteration t can be formulated as:

$$\begin{cases} \mathcal{L}_{\text{Recon}}^{\text{MSOO}} = \mathcal{L}_{\text{Recon}} + \lambda_{\text{Recon}}(t)\mathcal{L}_{\text{Total}} \\ \mathcal{L}_{\text{KL}}^{\text{MSOO}} = \mathcal{L}_{\text{KL}} + \lambda_{\text{KL}}(t)\mathcal{L}_{\text{Total}} \\ \mathcal{L}_{\text{ECR}}^{\text{MSOO}} = \mathcal{L}_{\text{ECR}} + \lambda_{\text{ECR}}(t)\mathcal{L}_{\text{Total}} \end{cases} \tag{10}$$

These surrogate losses are then similar to Static MSOO, they are used as the input for the MOO algorithms. The detailed training algorithms for MSOO-Static and MSOO-Adaptive are provided in Algorithms 1 and 2 respectively.

4 Experiments

4.1 Settings

Datasets. We employ some well-known datasets, including three standard datasets: **20 News Groups (20NG)** (Lang, 1995), a popular benchmark for topic modeling with 20 labels, **YahooAnswers** (Zhang et al., 2015), which contains question titles, contents, and the best answers from the Yahoo! Answers platform, and **AGNews** (Zhang et al., 2015) includes news articles and descriptions from over 2,000 sources. The preprocessing steps and statistics of all datasets are detailed in Appendix B.3.

Evaluation Metrics. We follow the evaluation framework from (Wu et al., 2023) to assess topic quality and document-topic distributions using topic coherence (Cv), Topic Diversity (TD), NMI, and Purity. Detailed descriptions of these evaluation metrics are provided in the Appendix B.1.

Baseline models. We assess the performance of our approach by applying it to two recent state-of-the-art neural topic modeling frameworks. In particular, we consider ECRTM (Wu et al., 2023), which generates diverse and coherent topics while ensuring high-quality topic distributions for documents by forcing each topic embedding to be the center of a separately aggregated word embedding cluster; and NeuroMax (Pham et al., 2024), which regularizes document-topic distributions by leveraging pre-trained language model embeddings to maximize mutual information, and employs optimal transport to learn the relationships between topics.

Multi-objective Optimization Algorithm Our experiments are conducted on two methods for Multi-objective Optimization. These models include MGDA (Sener and Koltun, 2018), which computes a Pareto-stationary descent direction by finding the minimum-norm convex combination of per-task gradients; and FairGrad (Ban and Ji, 2024), a recent advanced method which model as a utility maximization problem, where each task is associated with α-fair utility function and different α yields different ideas of fairness.

4.2 Main Results

To demonstrate the performance of MSOO methods and the ineffectiveness of traditional MOO approaches, we conducted experiments on standard datasets using two advanced MOO algo-

Models	AGNews				YahooAnswers				20NG			
	TD15	NMI	Purity	Cv15	TD15	NMI	Purity	Cv15	TD15	NMI	Purity	Cv15
ECRTM	0.961	0.367	0.802	0.466	0.985	0.295	0.550	0.405	0.964	0.524	0.560	0.431
+ FairGrad (MOO)	0.844	0.375	0.772	0.465	0.896	0.301	0.543	0.404	0.945	0.506	0.549	0.422
+ FairGrad (MSOO-S)	0.976	0.369	0.815	0.471	0.985	0.317	0.577	0.416	0.901	0.548	0.584	0.447
+ FairGrad (MSOO-A)	0.984	0.399	0.834	0.469	0.955	0.328	0.567	0.409	0.951	0.527	0.569	0.445
NeuroMax	0.952	0.410	0.804	0.385	0.979	0.331	0.588	0.404	0.912	0.570	0.623	0.435
+ FairGrad (MOO)	0.917	0.350	0.701	0.468	0.997	0.295	0.545	0.407	0.815	0.529	0.610	0.436
+ FairGrad (MSOO-S)	0.992	0.415	0.827	0.430	0.979	0.332	0.590	0.414	0.857	0.578	0.645	0.439
+ FairGrad (MSOO-A)	0.939	0.416	0.827	0.432	0.984	0.335	0.591	0.407	0.916	0.577	0.629	0.437

Table 1: Evaluation of ECRTM and NeuroMax performances on AGNews, YahooAnswers, and 20NG, measured using TD15, NMI, Purity, and Cv15 with FairGrad under MOO, MSOO-Static, and MSOO-Adaptive.

Models	AGNews				YahooAnswers				20NG			
	TD15	NMI	Purity	Cv15	TD15	NMI	Purity	Cv15	TD15	NMI	Purity	Cv15
ECRTM	0.961	0.367	0.802	0.466	0.985	0.295	0.550	0.405	0.964	0.524	0.560	0.431
+ MGDA (MOO)	0.895	0.362	0.762	0.456	0.906	0.303	0.551	0.384	0.909	0.511	0.527	0.418
+ MGDA (MSOO-S)	0.992	0.387	0.820	0.476	0.977	0.319	0.550	0.384	0.926	0.528	0.563	0.432
+ MGDA (MSOO-A)	0.989	0.391	0.817	0.463	0.891	0.310	0.555	0.393	0.891	0.544	0.602	0.440
NeuroMax	0.952	0.410	0.804	0.385	0.979	0.331	0.588	0.404	0.912	0.570	0.623	0.435
+ MGDA (MOO)	0.900	0.364	0.749	0.467	0.995	0.310	0.554	0.406	0.717	0.559	0.597	0.434
+ MGDA (MSOO-S)	0.965	0.413	0.816	0.436	0.981	0.338	0.586	0.404	0.841	0.585	0.645	0.449
+ MGDA (MSOO-A)	0.939	0.423	0.833	0.432	0.995	0.334	0.589	0.404	0.836	0.582	0.642	0.442

Table 2: Evaluation of ECRTM and NeuroMax performances on AGNews, YahooAnswers, and 20NG, measured using TD15, NMI, Purity, and Cv15 with MGDA under MOO, MSOO-Static, and MSOO-Adaptive.

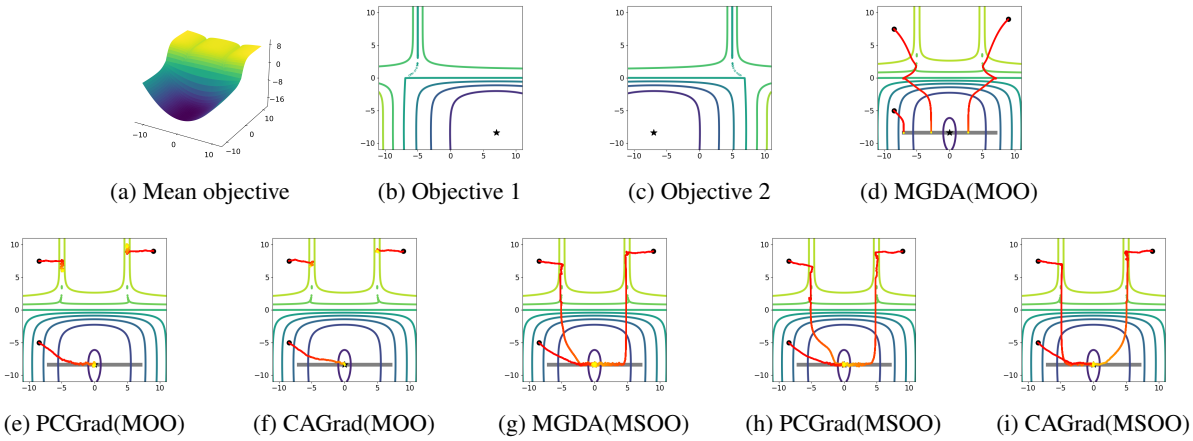
rithms: MGDA (Sener and Koltun, 2018) and FairGrad (Ban and Ji, 2024). We compared the baseline models with their corresponding MOO, MSOO-Static (MSOO-S), and MSOO-Adaptive (MSOO-A) variants. Tables 1 and 2 present the mean results for ECRTM and NeuroMax when using FairGrad and MGDA, respectively, highlighting the significant improvements brought by MSOO methods and the limited effectiveness of directly applying traditional MOO approaches to these models. The full results, including standard deviations, are reported in detail in Appendix C.5.

In general, both the static and adaptive variants of MSOO improved the quality of the topic models compared to the baseline and outperformed the direct MOO approach. Specifically, our MSOO methods considerably enhanced the quality of document-topic distributions, as demonstrated by the superior Purity and NMI scores, particularly in advanced models such as ECRTM and NeuroMax.

4.3 A Toy Example

To validate the effectiveness of our proposed approach beside its application in topic modeling, we assess its ability to achieve superior or comparable performance under stochastic conditions, we perform an empirical study on the two-objective toy example introduced in CAGrad (Liu et al., 2021a).

We consider three different initializations: $\mathbf{x}_0 \in \{(-8.5, 7.5), (-8.5, 5), (9, 9)\}$, which are employed for the various methods under consideration. The corresponding optimization trajectories are visualized in Figure 3. In Figures 3d–3i, the starting point of each trajectory is denoted by the ● symbol, while the trajectory color gradually transitions from red to yellow. The Pareto front is illustrated by the gray line, with the global optimum marked by the $\star$ symbol at its center. We implement MGDA (Sener and Koltun, 2018), PCGrad (Yu et al., 2020), and CAGrad (Liu et al., 2021a). Each algorithm is implemented both in its original form (MOO) and in conjunction with MSOO-S ($\lambda = 0.6$). In order to replicate the stochastic setting, we add zero-mean Gaussian noise to the gradient of each objective for all methods except MGDA. For each experimental run, we employ the Adam optimizer with a learning rate of 0.002 over 100,000 iterations. We can observe that, except for MGDA (Figure 3d),



(a) Mean objective (b) Objective 1 (c) Objective 2 (d) MGDA(MOO)

(e) PCGrad(MOO) (f) CAGrad(MOO) (g) MGDA(MSOO) (h) PCGrad(MSOO) (i) CAGrad(MSOO)

Figure 3: A two-objective toy example

PCGrad (Figure 3e) and CAGrad (Figure 3f) fail to converge to the Pareto front in some initializations. Although MGDA consistently reaches the Pareto front across all three initializations, it does not reliably converge to the global optimum ($\star$). In contrast, when using the MSOO framework, all three algorithms—MGDA, PCGrad, and CAGrad (Figure 3g, 3h and 3i)—successfully converge to the global optimum.

4.4 Effect of λ Hyperparameter

In Equation 3, the hyperparameter λ in our MSOO framework plays a crucial role in balancing the influence between the original loss components and the total loss within the surrogate objectives. As λ approaches zero, the surrogate losses converge to their original counterparts, diminishing the impact of $\mathcal{L}_{\text{Total}}$. Conversely, larger values of λ cause the surrogate losses to more closely resemble $\mathcal{L}_{\text{Total}}$. To investigate the effect of λ, we conducted experiments using ECRTM on the AGNews dataset, evaluating performance across a range of λ values: $[0.001, 0.01, 0.05, 0.1, 0.2, 0.4, 0.8, 1.5, 3, 6]$ in the MSOO-Static strategy and using FairGrad as MOO algorithm. Table 3 presents the performance metrics TD15, NMI, Purity, and Cv15.

The results demonstrate that both extremely small and large values of λ lead to suboptimal performance. When λ is very small (e.g., 0.001 or 0.01), the model's performance is significantly degraded, as evidenced by the low values across all metrics. On the other hand, when λ is excessively large, the surrogate losses become overly dominated by $\mathcal{L}_{\text{Total}}$, and while some metrics may appear high, they do not reflect a genuine improvement in topic quality.

λ	TD15	NMI	Purity	Cv15
0.001	0.463	0.021	0.269	0.382
0.01	0.525	0.020	0.264	0.340
0.05	0.100	0.381	0.692	0.482
0.1	0.141	0.399	0.804	0.473
0.2	0.232	0.370	0.794	0.546
0.4	0.980	0.369	0.844	0.470
0.8	0.956	0.369	0.843	0.465
1.5	0.887	0.357	0.826	0.478
3.0	0.829	0.360	0.829	0.483
6.0	0.833	0.351	0.815	0.471

Table 3: Effect of λ on the performance of ECRTM using AGNews dataset. TD15, NMI, Purity, and Cv15 are the evaluation metrics.

Optimal performance is often achieved within a moderate range of λ values (e.g. between 0.4 and 0.8, as evidenced by the results in Table 3). Within this range, the model benefits from the balancing effect of the surrogate loss formulations without being overly constrained by $\mathcal{L}_{\text{Total}}$. This highlight the importance of selecting λ values that are neither too high nor too low. In our experiments, the $\lambda_i(t)$ values determined by MSOO-Adaptive consistently fall within this moderate range.

5 Conclusion

This paper introduces a novel approach to address the challenge of optimizing multiple loss functions in NTMs. Directly applying MOO methods is often ineffective in this context. Our proposed method, MSOO, overcomes this limitation by employing surrogate loss functions with more balanced magnitudes, which consider not only individual ob-

jectives but also the total loss. Comprehensive experiments demonstrate that MSOO significantly improves the performance of baseline topic models, outperforming direct applications of MOO. We explore both static and adaptive strategies for weighting the surrogate losses, with the adaptive approach showing particularly strong potential.

Limitations

While MSOO demonstrates significant improvements, some limitations warrant consideration. First, MSOO may introduce a higher computational cost, which increases training time compared to baseline models. This could be a concern for resource-constrained large-scale or real-time application scenarios, and further research is necessary to improve its computational efficiency. Second, the current surrogate loss formulation, which relies on a linear combination of objectives, could potentially be enhanced with more sophisticated in the future.

Ethical Considerations

Our research follows the ACL Code of Ethics and all relevant licensing terms. We believe our work in topic modeling contributes positively to the field and presents no significant societal risks with responsible use.

Acknowledgements

Trung Le was partly supported by the Air Force Office of Scientific Research under award number FA2386-23-1-4044. Thien Huu Nguyen has been supported by the NSF grant # 2239570. He is also supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via the HIATUS Program contract 2022-22072200003. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes, notwithstanding any copyright annotation therein.

References

Hao Ban and Kaiyi Ji. 2024. Fair resource allocation in multi-task learning. *Preprint*, arXiv:2402.15638.

David M. Blei and John D. Lafferty. 2005. Correlated topic models. In *Neural Information Processing Systems*.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Gerlof Bouma. 2009. Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL*, 30:31–40.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. 2018. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. *Preprint*, arXiv:1711.02257.

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

Xin Gao and Cem Sazara. 2023. Discovering mental health research topics with topic modeling. *Preprint*, arXiv:2308.13569.

Thomas Griffiths, Michael Jordan, Joshua Tenenbaum, and David Blei. 2003. Hierarchical topic models and the nested chinese restaurant process. In *Advances in Neural Information Processing Systems*, 16.

Cuong Ha, Van-Dang Tran, Linh Ngo Van, and Khoat Than. 2019. Eliminating overfitting of probabilistic topic models on short and noisy text: The role of dropout. *International Journal of Approximate Reasoning*, 112:85–104.

Yifei He, Shiji Zhou, Guojun Zhang, Hyokun Yun, Yi Xu, Belinda Zeng, Trishul Chilimbi, and Han Zhao. 2024. Robust multi-task learning with excess risks. *Preprint*, arXiv:2402.02009.

Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015*.

Diederik P. Kingma and Max Welling. 2013a. Auto-encoding variational bayes. *CoRR*, abs/1312.6114.

Diederik P Kingma and Max Welling. 2013b. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014*.

Ken Lang. 1995. Newsweeder: Learning to filter netnews. In *Machine Learning Proceedings 1995*, pages 331–339, San Francisco (CA). Morgan Kaufmann.

Jian Liang, Kaixiong Gong, Shuang Li, Chi Harold Liu, Han Li, Di Liu, Guoren Wang, et al. 2021. Pareto domain adaptation. *Advances in Neural Information Processing Systems*, 34:12917–12929.

Baijiong Lin, Feiyang Ye, Yu Zhang, and Ivor W. Tsang. 2022. Reasonable effectiveness of random weighting: A litmus test for multi-task learning. *Preprint*, arXiv:2111.10603.

Ngo Van Linh, Nguyen Thi Kim Anh, Khoat Than, and Chien Nguyen Dang. 2017. An effective and interpretable method for document classification. *Knowledge and Information Systems*, 50:763–793.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. 2021a. Conflict-averse gradient descent for multi-task learning. *Advances in Neural Information Processing Systems*, 34:18878–18890.

Liyang Liu, Yi Li, Zhanghui Kuang, J Xue, Yimin Chen, Wenming Yang, Qingmin Liao, and Wayne Zhang. 2021b. Towards impartial multi-task learning. iclr.

Shikun Liu, Edward Johns, and Andrew J. Davison. 2019. End-to-end multi-task learning with attention. *Preprint*, arXiv:1803.10704.

Debabrata Mahapatra and Vaibhav Rajan. 2020. Multi-task learning with user preferences: Gradient descent with controlled ascent in pareto optimization. In *International Conference on Machine Learning*, pages 6597–6607. PMLR.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

Duc Anh Nguyen, Kim Anh Nguyen, Canh Hao Nguyen, Khoat Than, et al. 2021. Boosting prior knowledge in streaming variational bayes. *Neurocomputing*, 424:143–159.

Quang Duc Nguyen, Tung Nguyen, Duc Anh Nguyen, Linh Ngo Van, Sang Dinh, and Thien Huu Nguyen. 2025a. Glocom: A short text neural topic model via global clustering context. 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 1109–1124.

Thong Nguyen and Anh Tuan Luu. 2021. Contrastive learning for neural topic model. *Advances in neural information processing systems*, 34:11974–11986.

Thong Thanh Nguyen, Xiaobao Wu, Xinshuai Dong, Cong-Duy T Nguyen, See-Kiong Ng, and Anh Tuan Luu. 2024. Topic modeling as multi-objective optimization with setwise contrastive learning. In *The Twelfth International Conference on Learning Representations*.

Tung Nguyen, Tue Le, Hoang Tran Vuong, Quang Duc Nguyen, Duc Anh Nguyen, Linh Ngo Van, Sang Dinh, and Thien Huu Nguyen. 2025b. Sharpness-aware minimization for topic models with high-quality document representations. 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 4507–4524, Albuquerque, New Mexico. Association for Computational Linguistics.

Tung Nguyen, Trung Mai, Nam Nguyen, Linh Ngo Van, and Khoat Than. 2022. Balancing stability and plasticity when learning topic models from short and noisy text streams. *Neurocomputing*, 505:30–43.

Tung Nguyen, Duy-Tung Pham, Quang Duc Nguyen, Linh Ngo Van, Anh Nguyen Duc, and Sang Dinh Viet. 2025c. Topicot: Neural topic model aligning with pre-trained clustering and optimal transport. *Neurocomputing*, 654:131268.

Tung Nguyen, Tung Pham, Linh Ngo Van, Ha-Bang Ban, and Khoat Than. 2025d. Out-of-vocabulary handling and topic quality control strategies in streaming topic models. *Neurocomputing*, 614:128757.

Tung Nguyen, Linh Ngo Van, Anh Nguyen Duc, and Sang Dinh Viet. 2025e. A framework for neural topic modeling with mutual information and group regularization. *Neurocomputing*, 645:130420.

Van-Son Nguyen, Duc-Tung Nguyen, Linh Ngo Van, and Khoat Than. 2019. Infinite dropout for training bayesian models from data streams. In *2019 IEEE international conference on big data (Big Data)*, pages 125–134. IEEE.

Duy-Tung Pham, Thien Trang Nguyen Vu, Tung Nguyen, Linh Ngo Van, Duc Anh Nguyen, and Thien Huu Nguyen. 2024. Neuromax: Enhancing neural topic modeling via maximizing mutual information and group topic regularization. In *Findings of the Association for Computational Linguistics: EMNLP 2024*.

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*, page 399–408. Association for Computing Machinery.

Oleh Rybkin, Kostas Daniilidis, and Sergey Levine. 2021. Simple and effective vae training with calibrated decoders. In *International conference on machine learning*, pages 9179–9189. PMLR.

Ozan Sener and Vladlen Koltun. 2018. Multi-task learning as multi-objective optimization. *Advances in neural information processing systems*, 31.

Anh Phan Tuan, Bach Tran, Thien Huu Nguyen, Linh Ngo Van, and Khoat Than. 2020. Bag of biterms modeling for short texts. *Knowledge and Information Systems*, 62(10):4055–4090.

Francisco B. Valero, Marion Baranes, and Elena V. Epure. 2022. Topic modeling on podcast short-text metadata. *ArXiv*, abs/2201.04419.

Ngo Van Linh, Tran Xuan Bach, and Khoat Than. 2022. A graph convolutional topic model for short and noisy text streams. *Neurocomputing*, 468:345–359.

Tu Vu, Manh Do, Tung Nguyen, Linh Ngo Van, Sang Dinh, and Thien Huu Nguyen. 2025. Topic modeling for short texts via optimal transport-based clustering. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 7666–7680.

Hoang Tran Vuong, Tue Le, Tu Vu, Tung Nguyen, Linh Ngo Van, Sang Dinh, and Thien Huu Nguyen. 2025. HiCOT: Improving neural topic models via optimal transport and contrastive learning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 13894–13920, Vienna, Austria. Association for Computational Linguistics.

Chong Wang, David M. Blei, and David E. Heckerman. 2008. Continuous time dynamic topic models. In *Conference on Uncertainty in Artificial Intelligence*.

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.

Xiaobao Wu, Thong Nguyen, and Anh Tuan Luu. 2024. A survey on neural topic models: Methods, applications, and challenges. *Artificial Intelligence Review*, 57(2):1–30.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. 2020. Gradient surgery for multi-task learning. *Preprint*, arXiv:2001.06782.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, 28. Curran Associates, Inc.

A Gradient-based Multi-objective Optimization for Multi-task Learning

Multi-task learning (MTL) can be formulated as a Multi-objective Optimization problem. With K be the total number of tasks, let δ^{share} are shared parameters while others $\delta^1, \delta^2, \ldots, \delta^K$ denote task-specific parameters within the feasible set Θ, $\mathcal{L}_i$ represent the training loss associated with task i. The objective is to minimize all K losses simultaneously:

$$\min_{\delta^{\text{share}}} [\mathcal{L}_1(\delta^{\text{share}}, \delta^1), \mathcal{L}_2(\delta^{\text{share}}, \delta^2), \ldots, \mathcal{L}_K(\delta^{\text{share}}, \delta^K)] \tag{11}$$

Given two feasible solutions δ_1 and δ_2 to problem (11).

Definition 2.1 (Pareto dominance). We state that δ_1 dominates δ_2 if and only if $\mathcal{L}_i(\delta_1) \leq \mathcal{L}_i(\delta_2)$ for all $i \in \{1, \ldots, K\}$ and there exists at least one $j \in \{1, \ldots, K\}$ such that $\mathcal{L}_j(\delta_1) < \mathcal{L}_j(\delta_2)$ with the notation $\delta_1 \prec \delta_2$.

Definition 2.2 (Pareto optimal). A feasible solution is considered Pareto-optimal if it is not dominated by any other solutions. The set of all Pareto-optimal solutions is known as the Pareto front. Achieving Pareto optimality is the purpose of Multi-objective Optimization.

There can be multiple Pareto optimal solutions, which collectively form the Pareto set. A weaker condition known as Pareto stationarity and all Pareto optimal points are Pareto stationary; however, the converse does not hold true.

Definition 2.3 (Pareto stationary). A point $\delta \in \mathbb{R}^m$ is considered Pareto stationary if $\min_{w \in \mathcal{W}} \|G(\delta)w\| = 0$, where $G(\delta) = \left[g_1(\delta), \cdots, g_K(\delta)\right] \in \mathbb{R}^{m \times K}$ is a matrix whose columns $g_i(\delta)$ represent the gradients of the i-th objective, and $\mathcal{W}$ denotes the probability simplex over $[K]$. Pareto stationarity serves as a necessary condition for Pareto optimality.

B Experiment Details

B.1 Evaluation Metrics.

We follow the evaluation framework proposed in (Wu et al., 2023) to assess both topic quality and document-topic distributions. Topic quality is evaluated using coherence and diversity metrics. With coherence, we apply $\mathrm{Cv15}$, where 15 denotes the top words in each discovered topic — this metric has been shown to strongly align with human judgment (Röder et al., 2015). The coherence calculations are based on the Wikipedia corpus[1] as an external reference corpus. To evaluate topic diversity, we use Topic Diversity metric (Dieng et al., 2020) which computes the proportion of unique words in the discovered topics. We select the top 15 words of discovered topics for the above topic quality evaluation. For evaluating document-topic distribution, we employ Normalized Mutual Information (NMI) and Purity (Manning et al., 2008), where the dominant topic of each document is used to assign it to a cluster. While $\mathrm{Cv15}$, Purity, and NMI assess the generalization performance on external and test data, $\mathrm{TD15}$ ensures that topics maintain sufficient diversity without overlap. In our experiments, we set the number of topics to 50.

B.2 Implementation Details.

All experiments were conducted on a machine equipped with an NVIDIA GeForce RTX 3090 GPU (24GB RAM), using PyTorch 2.1.0 with CUDA 12.1 in a Python 3.12 environment. Models were trained for 500 epochs with a batch size of 200, employing the Adam optimizer (Kingma and Ba, 2015) with a learning rate of 0.002. For MSOO-Static, the λ hyperparameter was selected from the set $\{0.3, 0.4, 0.5, 0.6, 0.8, 1.0, 1.5\}$.

B.3 Dataset Statistics

Our experiments utilized some well-known datasets, including three standard datasets: **20 News Groups (20NG)** (Lang, 1995), **AGNews** (Zhang et al., 2015), and **YahooAnswers** (Zhang et al., 2015). We applied the pre-processing steps described in (Wu et al., 2023) to generate bag-of-words representations. These

[1] https://github.com/dice-group/Palmetto/

Dataset	# of texts	average text length	# of labels	vocab size
20NG	18,846	110.5	20	5,000
YahooAnswers	12,500	35.4	10	5,000
AGNews	12,500	20.1	4	5,000

Table 4: Dataset statistics after preprocessing.

pre-processing procedures were carried out using the TopMost tool[2]. The detailed statistics of all datasets after processing are presented in Table 4.

C Additional Experiments

C.1 NPMI Coherence Evaluation

To further evaluate the coherence of discovered topics, we report Normalized Pointwise Mutual Information (NPMI) (Bouma, 2009) scores for two topic models: ECRTM and NeuroMax across two benchmark datasets (20NG and AGNews). The models are evaluated under the MSOO framework using MGDA as the optimization method with $K = 50$ topics. Table 5 summarizes the comparison between the original baseline models and their corresponding MSOO-Static (MSOO-S) variants. The results demonstrate that MSOO-S consistently improves NPMI coherence across most configurations, highlighting its robustness in topic quality enhancement.

Model	Dataset	Baseline	MGDA (MSOO-S)
NeuroMax	20NG	0.061	**0.118**
NeuroMax	AGNews	0.007	**0.010**
ECRTM	20NG	-0.061	**-0.023**
ECRTM	AGNews	-0.092	**-0.043**

Table 5: NPMI coherence scores for baseline models and their MSOO-Static (MSOO-S) variants using MGDA ($K = 50$). Results indicate that MSOO-S consistently improves topic coherence across both datasets and models.

C.2 Impact of Topic Count K

To assess the robustness of our method under varying topic dimensionality, we conduct additional experiments on the YahooAnswers dataset using the ECRTM model as baseline. Specifically, we vary the number of topics $K \in \{50, 100, 150, 200\}$ and report performance metrics across three training regimes: baseline ECRTM, MSOO-S, and MSOO-A. The results are summarized in Table 6. Both MSOO-S and MSOO-A generally improved performance compared to the baseline, particularly in NMI and Cv15, across different topic settings.

C.3 Perplexity Evaluation

To further assess model quality from a generative perspective, we conducted additional experiments measuring perplexity (PPL). For consistency with prior work, we adopted the PPL computation method from Nguyen et al. (Nguyen et al., 2022), where higher perplexity values indicate a better model fit. We evaluated ECRTM and its MSOO-enhanced variants using three representative MOO algorithms—MGDA, PCGrad, and FairGrad—across three benchmark datasets with $K = 50$ topics.

The results are reported in Table 7. These findings confirm that MSOO consistently achieves improved perplexity scores across all tested datasets, further supporting its effectiveness from a generative modeling perspective.

[2] https://github.com/bobxwu/topmost

K	Model	TD15	NMI	Purity	Cv15
50	ECRTM (Baseline)	**0.985**	0.295	0.550	**0.405**
	ECRTM + MGDA (MSOO-S)	0.977	**0.319**	0.550	0.384
	ECRTM + MGDA (MSOO-A)	0.891	0.310	**0.555**	0.393
100	ECRTM (Baseline)	0.903	0.311	0.563	0.389
	ECRTM + MGDA (MSOO-S)	**0.921**	**0.318**	0.566	**0.391**
	ECRTM + MGDA (MSOO-A)	0.905	0.313	**0.568**	0.389
150	ECRTM (Baseline)	**0.888**	0.310	0.573	0.376
	ECRTM + MGDA (MSOO-S)	**0.888**	0.317	0.575	**0.386**
	ECRTM + MGDA (MSOO-A)	0.851	**0.318**	**0.577**	0.376
200	ECRTM (Baseline)	0.832	0.311	0.573	0.377
	ECRTM + MGDA (MSOO-S)	0.843	**0.319**	0.572	**0.383**
	ECRTM + MGDA (MSOO-A)	**0.898**	0.310	**0.578**	0.379

Table 6: Performance of ECRTM on YahooAnswers with MGDA (MOO) across varying topic counts. Results are shown for the baseline, MSOO-Static, and MSOO-Adaptive settings.

Model	YahooAnswers	AGNews	20NG
ECRTM	-3.592	-3.383	-3.644
ECRTM + MGDA (MSOO-S)	-3.540	-3.347	-3.634
ECRTM + MGDA (MSOO-A)	-3.513	-3.324	-3.639
ECRTM + FairGrad (MSOO-S)	**-3.479**	**-3.288**	-3.620
ECRTM + FairGrad (MSOO-A)	-3.492	-3.296	**-3.602**

Table 7: Perplexity for ECRTM and MSOO variants across three datasets. Higher values indicate better model fit. Bold highlights the best-performing model per dataset.

C.4 Loss Evaluation

We further evaluated the final values of each original loss component—Reconstruction Loss, KL Divergence, and ECR Loss—for the baseline ECRTM and its MSOO variants under the MGDA optimizer. To improve readability, we report results per dataset in Table 8, showing that both MSOO-S and MSOO-A provide a more balanced loss distribution across objectives.

As observed across all datasets, the direct application of MGDA tends to prioritize minimizing Reconstruction Loss, but often fails to jointly optimize KL Divergence and ECR Loss effectively. In contrast, both MSOO-S and MSOO-A promote a more holistic trade-off, significantly reducing KL and ECR Loss without sacrificing reconstruction performance. These results empirically validate the ability of MSOO to mitigate loss imbalance and enhance optimization effectiveness.

C.5 Mean and Standard Deviation Results

In Tables 9 and 10, we report the mean and standard deviation of the performance metrics corresponding to Tables 1 and 2, averaged over five independent runs.

Dataset	Model	Recon	KL	ECR
YahooAnswers	ECRTM	235.908	30.411	7.396
	ECRTM + MGDA (MOO)	225.367	31.355	5.399
	ECRTM + MGDA (MSOO-S)	232.944	30.087	2.443
	ECRTM + MGDA (MSOO-A)	222.754	29.373	2.448
AGNews	ECRTM	135.243	7.389	1.722
	ECRTM + MGDA (MOO)	128.559	7.167	1.844
	ECRTM + MGDA (MSOO-S)	129.057	7.065	1.612
	ECRTM + MGDA (MSOO-A)	131.774	6.998	1.529
20NG	ECRTM	619.238	34.807	14.186
	ECRTM + MGDA (MOO)	608.642	32.067	15.295
	ECRTM + MGDA (MSOO-S)	595.826	31.902	8.181
	ECRTM + MGDA (MSOO-A)	611.020	29.957	8.813

Table 8: Final loss values of ECRTM and MGDA (MOO), MGDA (MSOO-S) and MGDA (MSOO-A) across individual objectives: Reconstruction Loss (Recon), KL Divergence (KL), and ECR Loss (ECR). Lower values are preferred.

Models	AGNews				YahooAnswers				20NG			
	TD15	NMI	Purity	Cv15	TD15	NMI	Purity	Cv15	TD15	NMI	Purity	Cv15
ECRTM	0.961	0.367	0.802	0.466	0.985	0.295	0.550	0.405	0.964	0.524	0.560	0.431
+ FairGrad (MOO)	0.844 ± 0.015	0.375 ± 0.011	0.772 ± 0.017	0.465 ± 0.018	0.986 ± 0.005	0.301 ± 0.006	0.543 ± 0.017	0.404 ± 0.011	0.945 ± 0.019	0.506 ± 0.014	0.549 ± 0.018	0.422 ± 0.010
+ FairGrad (MSOO-S)	0.976 ± 0.008	0.369 ± 0.012	0.815 ± 0.019	0.471 ± 0.016	0.985 ± 0.004	0.317 ± 0.007	0.577 ± 0.013	0.416 ± 0.008	0.901 ± 0.021	0.548 ± 0.013	0.584 ± 0.017	0.447 ± 0.012
+ FairGrad (MSOO-A)	0.984 ± 0.005	0.399 ± 0.011	0.834 ± 0.015	0.469 ± 0.015	0.955 ± 0.018	0.328 ± 0.009	0.567 ± 0.016	0.409 ± 0.010	0.951 ± 0.017	0.527 ± 0.011	0.569 ± 0.014	0.445 ± 0.014
NeuroMax	0.952	0.410	0.804	0.385	0.979	0.331	0.588	0.404	0.912	0.570	0.623	0.435
+ FairGrad (MOO)	0.917 ± 0.018	0.350 ± 0.014	0.701 ± 0.022	0.468 ± 0.017	0.997 ± 0.001	0.295 ± 0.007	0.545 ± 0.015	0.407 ± 0.010	0.815 ± 0.019	0.529 ± 0.011	0.610 ± 0.017	0.436 ± 0.014
+ FairGrad (MSOO-S)	0.992 ± 0.002	0.415 ± 0.013	0.827 ± 0.016	0.430 ± 0.014	0.979 ± 0.008	0.332 ± 0.006	0.590 ± 0.012	0.414 ± 0.008	0.857 ± 0.018	0.578 ± 0.014	0.645 ± 0.015	0.439 ± 0.010
+ FairGrad (MSOO-A)	0.939 ± 0.016	0.416 ± 0.011	0.827 ± 0.018	0.432 ± 0.017	0.984 ± 0.005	0.335 ± 0.008	0.591 ± 0.014	0.407 ± 0.009	0.916 ± 0.017	0.577 ± 0.012	0.629 ± 0.018	0.437 ± 0.011

Table 9: Evaluation of ECRTM and NeuroMax on AGNews, YahooAnswers, and 20NG, measured using TD15, NMI, Purity, and Cv15 with FairGrad under MOO, MSOO-Static, and MSOO-Adaptive. Each cell reports mean ± standard deviation.

Models	AGNews				YahooAnswers				20NG			
	TD15	NMI	Purity	Cv15	TD15	NMI	Purity	Cv15	TD15	NMI	Purity	Cv15
ECRTM	0.961	0.367	0.802	0.466	0.985	0.295	0.550	0.405	0.964	0.524	0.560	0.431
+ MGDA (MOO)	0.895 ± 0.014	0.362 ± 0.020	0.762 ± 0.017	0.456 ± 0.024	0.906 ± 0.023	0.303 ± 0.016	0.551 ± 0.015	0.384 ± 0.017	0.909 ± 0.018	0.511 ± 0.020	0.527 ± 0.016	0.418 ± 0.020
+ MGDA (MSOO-S)	0.992 ± 0.005	0.387 ± 0.018	0.820 ± 0.012	0.476 ± 0.020	0.977 ± 0.012	0.319 ± 0.018	0.550 ± 0.019	0.384 ± 0.014	0.926 ± 0.027	0.528 ± 0.015	0.563 ± 0.012	0.432 ± 0.018
+ MGDA (MSOO-A)	0.989 ± 0.005	0.391 ± 0.016	0.817 ± 0.014	0.463 ± 0.016	0.891 ± 0.022	0.310 ± 0.014	0.555 ± 0.016	0.393 ± 0.010	0.891 ± 0.025	0.544 ± 0.014	0.602 ± 0.018	0.440 ± 0.018
NeuroMax	0.952	0.410	0.804	0.385	0.979	0.331	0.588	0.404	0.912	0.570	0.623	0.435
+ MGDA (MOO)	0.900 ± 0.014	0.364 ± 0.019	0.749 ± 0.015	0.467 ± 0.022	0.995 ± 0.002	0.310 ± 0.016	0.554 ± 0.013	0.406 ± 0.015	0.717 ± 0.037	0.559 ± 0.018	0.597 ± 0.015	0.434 ± 0.020
+ MGDA (MSOO-S)	0.965 ± 0.018	0.413 ± 0.020	0.816 ± 0.016	0.436 ± 0.016	0.981 ± 0.010	0.338 ± 0.014	0.586 ± 0.018	0.404 ± 0.018	0.841 ± 0.016	0.585 ± 0.020	0.645 ± 0.016	0.449 ± 0.018
+ MGDA (MSOO-A)	0.939 ± 0.014	0.423 ± 0.018	0.833 ± 0.017	0.432 ± 0.018	0.995 ± 0.003	0.334 ± 0.014	0.589 ± 0.019	0.404 ± 0.015	0.836 ± 0.011	0.582 ± 0.017	0.642 ± 0.018	0.442 ± 0.013

Table 10: Evaluation of ECRTM and NeuroMax on AGNews, YahooAnswers, and 20NG, measured using TD15, NMI, Purity, and Cv15 under MGDA with MOO, MSOO-Static, and MSOO-Adaptive. Each cell reports mean ± standard deviation.