EDITORS Know Your Style! Editing LoRA Subspaces for Stylistic Attribution and Imitation

Zixuan Wang[*1,2] **Gregory Kang Ruey Lau**[*1,2] **Xiao Tian**[1,4] **Jue Fan**[1,4]
Caroline Chaux[3] **Bryan Kian Hsiang Low**[1]
[1]School of Computing, National University of Singapore, Singapore
[2]CNRS@CREATE, 1 Create Way, #08-01 Create Tower, Singapore 138602
[3]Aix Marseille Univ, CNRS, I2M, Marseille, France
[4]Agency for Science, Technology and Research (A*STAR), Singapore
{e0492448, gregorylau, xiao.tian, jue.fan}@u.nus.edu
caroline.chaux@cnrs.fr, dcslkh@nus.edu.sg

Abstract

Disentangling writing style from semantic content is a fundamental challenge in literary text modeling. Style-content entanglement causes models to rely on semantic content for *authorship attribution* (AA) tasks and memorize author-specific content for *style imitation* (SI) tasks. We propose EDITORS (**Edit**ing L**o**R**A** **S**ubspaces), a *diagnose-then-deflate* framework that adapts pretrained large language models and prioritizes stylistic features, making it effective for both AA and SI tasks. EDITORS first trains a diagnostic LoRA adapter using style-neutral statements derived from the training corpus to identify a content-oriented subspace. It then trains the final style adapter under activation-space deflation that projects input activations away from the identified content-oriented directions. We empirically demonstrate that EDITORS is effective. On three AA benchmarks spanning literature, social media, and news, EDITORS achieves state-of-the-art accuracy, outperforming the strongest existing baseline by up to 29%. On SI, EDITORS generates stylistically faithful and diverse passages, outperforming baselines in achieving style similarity while adhering to content specifications. Our code is available at `https://github.com/zixuan-n/editors`.

1 Introduction

The disentanglement of style and content has been extensively studied in images [24, 31, 43, 46, 51], yet its generalization to literary text remains a complex challenge: style and content are deeply entangled in natural languages. The disentanglement is crucial in the era of large language models (LLMs) due to two important real-world applications: *authorship attribution* (AA), which identifies the most likely author of a given text, and *style imitation* (SI), which generates text in the style of a target author and preserves user-provided content. However, in AA, current methods either (a) conflate style with content [12, 27, 26, 54] due to their over-reliance on content and fail when an author writes across topics or when attribution must generalize across domains, or (b) aggressively remove useful content information [7, 22, 25, 45, 57]. Likewise, in SI, existing methods (a) risk memorizing training content and provide limited support for content customization [36, 39, 50], or (b) forcefully avoid content with expensive data pairs and only perform narrow paragraph-to-paragraph rewriting [18, 38, 52, 59]. These approaches reveal a persistent trade-off: retaining content can improve task performance but risks shortcut learning and memorization, while ignoring content can discard information that remains useful for attribution and generation.

[*]These authors contributed equally.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

To tackle the challenge of balancing style and content in both AA and SI, this paper presents **EDITORS** (**Edit**ing L**o**R**A** **S**ubspaces), a framework that uses LoRA adapters to diagnose and suppress content-oriented directions during supervised fine-tuning (SFT)[2] to **encourage stylistic features while retaining useful content information**. Our framework is motivated by the intuition that style and content are partially, but not fully, separated in the model's activation space. By diagnosing the content-oriented subspace and projecting away from it during style learning, we can train a style adapter which is encouraged to rely on stylistic features (Sec. 3.1). Based on this, EDITORS adopts a *diagnose-then-deflate* strategy: We first **diagnose important content-oriented directions** using a set of *style-neutral statements* (Sec. 3.2). We then **prioritize learning stylistic features** by suppressing these selected content-oriented directions through **activation-space *deflation*** (i.e., projecting training data away from them) (Sec. 3.3). Our framework EDITORS can be applied to both AA and SI tasks with slight adjustments (Sec. 3.4). Our specific contributions include:

- We propose EDITORS, a diagnose-then-deflate framework that prioritizes style learning by suppressing selected content directions in activation space (Sec. 3);
- On AA, EDITORS consistently improves over baselines across various domains (Sec. 4.1);
- On SI, EDITORS achieves **higher style similarity, lower memorization of training content and better preservation of user-provided content**, demonstrating that the learned author-specific stylistic features transfer beyond direct copying (Sec. 4.2).

2 Problem Formulation

Let $\mathcal{A}$ be a finite set of authors, $\mathcal{C}$ be the space of semantic content specifications, and $\mathcal{T}$ be the space of text passages. We consider the setting where each author $a \in \mathcal{A}$ is associated with two distributions: a **content** distribution C_a over $\mathcal{C}$ that captures what author a tends to write about, and an expression distribution $S_a(\cdot \mid c)$ over text passages $t \in \mathcal{T}$ that captures how the author expresses a given content specification $c \in \mathcal{C}$ as text, which we associate with author **style**. The text passage distribution over $\mathcal{T}$ that we can observe for author a is then the marginal $T_a(t) = \int_{\mathcal{C}} S_a(t \mid c)\, dC_a(c)$. Text samples $t \in \mathcal{T}$ from T_a entangle two author-specific signals: the author's typical content C_a and style of realizing content $S_a(\cdot \mid c)$. In this work, given a training dataset $\mathcal{D} = \{(t_i, a_i)\}_{i=1}^N$ covering all authors $a \in \mathcal{A}$, our goal is to develop a framework to address such entanglement and prioritize stylistic features when fine-tuning LLMs, which is essential to tasks including AA and SI.

Authorship attribution (AA). In the closed-set AA task, given a passage t and a candidate author set $\mathcal{A}$, our goal is to predict the most likely author $\hat{a} = \arg\max_{a \in \mathcal{A}} P(a \mid t)$. We aim to do so by training an attribution model based on training data $\mathcal{D} = \{(t_i, a_i)\}_{i=1}^N$, so that it can perform well on held-out passages $\mathcal{D}_{\text{test}}$ too. Ideally, the model should identify which author-specific marginal distribution T_a most likely produced the passage, while avoiding brittle reliance on content shortcuts from C_a. This distinction matters because different authors may write about similar content, and the same author may write about different content across topics, genres, or domains unseen in training.

Style imitation (SI). In the general SI task, the model is given a target author $a \in \mathcal{A}$ and the goal is to generate a passage $\hat{t} \sim P(\cdot|a)$ that approximates author a's marginal distribution T_a. A useful variant is *content-controlled SI*, where the model is additionally given a content specification $\ddot{c} \in \mathcal{C}$. The goal is to generate a passage $\hat{t} \sim P(\cdot \mid a, \ddot{c})$ that approximates author a's expression distribution $S_a(\cdot \mid \ddot{c})$, or essentially how the author would likely express the requested content $\ddot{c}$. The ideal generative model should (1) be stylistically faithful to the target author a, (2) accurately convey the specified content $\ddot{c}$, and (3) avoid just copying from its training data $\mathcal{D}$.

Note that both tasks require learning from samples $t \sim T_a$, where content C_a and style S_a are observed entangled in text passage form. For AA, content information from C_a can be useful but misleading: some content patterns may help identify an author easily during training, but over-reliance on C_a can lead to poor generalization under topic or domain shift (e.g., when an author writes about a different topic). For SI, the issue may be more problematic: preserving author-specific content patterns from C_a can cause the model to reproduce training passages or ignore the user-provided content specification $\ddot{c}$ in content-controlled SI. **Hence, our method must rely less on content information from C_a while retaining the author style information from S_a.** As C_a and S_a are

[2]EDITORS can be applied to any SFT method besides LoRA (App. C.1), but we focus on LoRA in this paper because it is efficient and simplifies our computation. See App. D for more explanations and demonstration on non-LoRA methods.

latent, our method does not estimate them directly but instead adopts a geometric intervention that reduces content-oriented components to achieve our objective (Sec. 3.1).

2.1 Related Works

Authorship attribution (AA). Existing AA methods typically either: **(a)** rely primarily on content information, and **(b)** attempt to reduce content information. Works in **(a)** (e.g., BertAA [12], STAR [27], PART [26] and TRACE [54]) treat each text as a single representation, often cause content signals to dominate and may exploit semantic content shortcuts. Recent LLM-based AA methods use in-context learning [23] where reference texts for all candidate authors are provided in prompts, but face challenges in scaling due to context-window limits and high inference cost, and do not generalize well across domains. In contrast, works in **(b)** explicitly emphasize style and attempt to reduce content dependence. *Stylometry*[16] directly extracts numeric stylistic features to train classification models, but its performance declines quickly with number of authors. Other works include those training content-independent style embeddings by contrastive learning [57, 45] and separate style and content encoders by minimizing their mutual information [57, 45]. However, these methods tend to remove semantic information and yield low performance in real AA tasks. Recent works such as Waterfall [34] proposed an active AA approach where text are paraphrased with LLMs to embed author-specific watermarks. Such watermarks are robust and can be used to track LLM data provenance [20, 41], but require active data processing that may impact the author's original style. EDITOR presents a complementary approach for attribution without edits to the original text.

Style imitation (SI). Similarly, existing SI methods typically either: **(a)** do not attempt to separate content and style, and **(b)** focus on learning style by controlling or removing content. For **(a)**, [50, 39, 36] directly fine-tune a language model on a target author corpus but risk memorizing training passages. Prompt-based methods [10, 56] are tuning-free and flexible, but their stylistic control is limited by context-window size and prompt sensitivity [4, 33], tending to capture surface markers rather than implicit writing habits. For **(b)**, early methods [28] use true parallel source-target pairs which are rare and expensive to construct for real authors. Recent methods such as ASTRAPOP [38] relax this constraint by constructing pseudo-parallel data by neutral paraphrasing or back-translation, then training models against style, content, and fluency rewards. However, most RL-based methods operate on coarse attributes such as sentiment, genre or formality [42, 19, 37] rather than individual author style. Methods such as StyleLM [52], StoryTrans [59] and ParChoice [18] rewrite arbitrary input text in a target style through denoising autoencoders, combinatorial paraphrasing etc. However, they are typically evaluated on a few authors with highly distinctive voices and only operate on paragraph-level rewriting rather than long-passage generation from compact content specification. While not applied to SI, past works have studied parameter-space subspace projection for model editing and LoRA fusion of separate tasks [55, 13, 6]. Unlike these works, EDITORS applies activation-space deflation that allows it to disentangle style and content within text (see App. C.2 for details).

Unlike existing works, our method EDITORS learn style while optionally retaining content information with our diagnose-then-deflate approach, significantly improving AA and SI performance over baselines. It does not require specially curated data, scales to larger author sets, and can generate long stylized passages from compact content specifications.

3 The EDITORS Framework

Our training framework EDITORS adapts a pretrained LLM to perform AA and SI and prioritizes stylistic features. Fig. 1 gives an overview of the EDITORS framework and Alg. 1 gives the pseudocode. EDITORS follows a *diagnose-then-deflate* strategy, where we first train a diagnostic LoRA adapter on style-neutral statements and use it to identify activation directions that are predictive for semantic content. We then deflate the input activations of training data away from these directions and train a new LoRA adapter (i.e., *style adapter*). During inference, the style adapter introduces author-specific style features, while the frozen base model provides necessary content support.

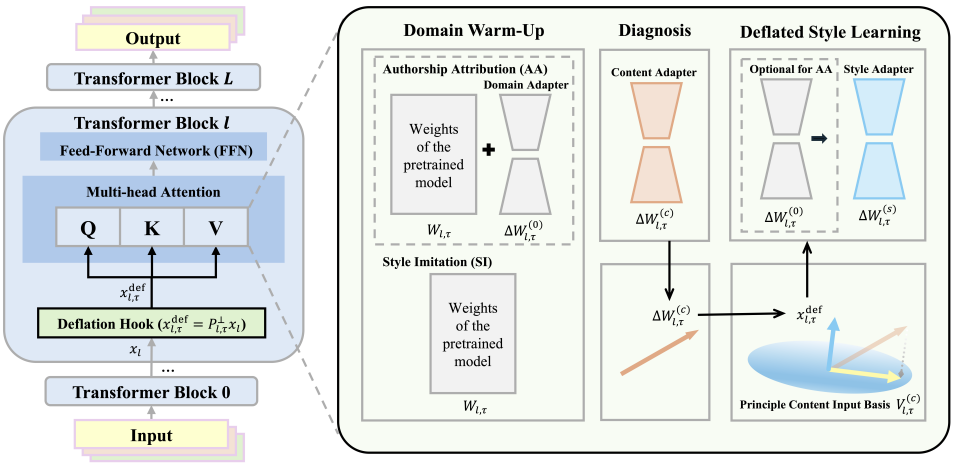

Figure 1: Overview of the EDITORS pipeline.

3.1 Geometric Interventions in LLM Representations

As described in Sec. 2, each author a induces a marginal passage distribution T_a by combining a content distribution C_a with an expression distribution $S_a(\cdot \mid c)$. Ideally, for robust results, AA should avoid relying only on differences in C_a, and (content-controlled) SI should approximate $S_a(\cdot \mid \ddot{c})$ for user-specified content $\ddot{c}$. However, in practice C_a, S_a and the latent content c are not directly observed, and cannot be clearly identified/disentangled from observed text $t \sim T_a$.

Instead of estimating the distributions directly, EDITORS makes use of pretrained LLMs' representation space, with its rich text semantic and stylistic features, as an operational domain to intervene on style-content entanglement. Specifically, let $x_\ell \in \mathbb{R}^{d_{\text{in}}}$ denote the input activation produced by the frozen pretrained model at layer ℓ (e.g., for passage t). Samples from the marginal distribution T_a induce a corresponding distribution over activations x_ℓ. As discussed in Sec. 2, variation in text and consequently the activations can arise from both the latent content $c \sim C_a$ and the author's expression pattern $S_a(\cdot \mid c)$. A standard adapter trained on $\mathcal{D} = \{(t_i, a_i)\}_{i=1}^N$ may therefore learn directions in activation space that are predictive of the author only due to content patterns from C_a, making it less robust to any test-time distribution shift of C_a.

Our key intuition is that the dominant content-oriented variation in x_ℓ can be captured by a low-dimensional linear subspace. Deflating the input activations x_ℓ away from this subspace will then give a tractable way to focus training of a style adapter on modeling an author's stylistic features. In Sec. 3.2, we explain how we can practically identify the diagnostic content-oriented subspace; in Sec. 3.3, we describe how we can use the identified subspace to prioritize style learning.

3.2 Diagnosing Content-Oriented Directions

Intuitively, the content-oriented directions will dominate in an adapter trained using content-heavy and style-neutral passages. Such passages can be effectively obtained in practice by prompting an LLM to rewrite each training passage t_i into pieces of style-neutral statements[3]. Using this technique (more details in App. G.1), we first construct an auxiliary style-neutral training corpus $\mathcal{D}^{(c)}$, which contains pieces of style-neutral statements that preserve entities, events, claims, and other semantic information while suppressing author-specific expression patterns. These statements $\mathcal{D}^{(c)}$ serve as an observable proxy for latent content $c \sim C_a$, and we make a local geometric assumption that the directions most useful for a content-oriented auxiliary task ($\mathcal{D}^{(c)}$) provide an approximate basis for content variation in the original task ($\mathcal{D}$). We empirically verify this assumption in App. I.5.

Thus, to identify the content-oriented directions, we start by training a *content adapter* $\Delta W^{(c)}$ from scratch using the standard next-token prediction objective on $\mathcal{D}^{(c)}$. For a selected LoRA module (ℓ, τ) at layer ℓ, the content-adapted computation is $h^{(c)}_{\ell,\tau} = \left(W_{\ell,\tau} + \Delta W^{(c)}_{\ell,\tau}\right) x_\ell$, where $W_{\ell,\tau}$ is

[3]In contrast, it is infeasible to prompt an external LLM to generate content-controlled passages with different author styles (required by existing works) due to poor SI performance.

the frozen base model. **The content adapter is not used in the final model, but is used only as a diagnostic object.** As it is trained for a content-oriented task, the input directions to which it is most oriented provide an estimate of content-oriented directions in the pretrained representation space.

To extract the content-oriented directions, we then apply singular value decomposition (SVD) [1] to the content adapter weights $\Delta W^{(c)}_{\ell,\tau} = U_{\ell,\tau}\Sigma_{\ell,\tau}V^{\top}_{\ell,\tau}$ for every LoRA module (ℓ, τ). Since $\Delta W^{(c)}_{\ell,\tau} \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$ acts on the input activation x_ℓ, its right singular vectors $V_{\ell,\tau}$ describe directions in the input activation space. We select the top-k of them as the *principal content input basis* :

$$V^{(c)}_{\ell,\tau} \;=\; V_{\ell,\tau}[:, 1:k] \in \mathbb{R}^{d_{\text{in}} \times k} \,. \tag{1}$$

The dimension k controls the strength of the intervention: larger k removes more diagnosed content-oriented directions, while smaller k retains more content-oriented directions. Note that this parameter is distinct from the LoRA rank r, which controls adapter capacity.

3.3 Deflated Style Learning

Given the content-oriented basis $V^{(c)}_{\ell,\tau}$, we define the deflation operator[4]

$$P^{\perp}_{\ell,\tau} = I - V^{(c)}_{\ell,\tau}\left(V^{(c)}_{\ell,\tau}\right)^{\top} \in \mathbb{R}^{d_{\text{in}} \times d_{\text{in}}} \,,$$

which projects input activation x_ℓ to the orthogonal complement of $V^{(c)}_{\ell,\tau}$. The deflated input is $x^{\text{def}}_{\ell,\tau} = P^{\perp}_{\ell,\tau}\, x_\ell \in \mathbb{R}^{d_{\text{in}}}$. By construction, $(V^{(c)}_{\ell,\tau})^{\top} x^{\text{def}}_{\ell,\tau} = 0$ and hence the component of the activation lying in the diagnosed content-oriented subspace will be absent from the module input. Note that EDITORS only removes the top-k directions identified by the content adapter $\Delta W^{(c)}_{\ell,\tau}$ at selected modules, instead of all semantic information.

We then train the final *style adapter* $\Delta W^{(s)}_{\ell,\tau}$ on the original author-labeled corpus using the deflated input activation $x^{\text{def}}_{\ell,\tau}$ at each layer ℓ (also used during inference):

$$h^{(s)}_{\ell,\tau} = \left(W_{\ell,\tau} + \Delta W^{(s)}_{\ell,\tau}\right)\, x^{\text{def}}_{\ell,\tau} = \left(W_{\ell,\tau} + \Delta W^{(s)}_{\ell,\tau}\right) P^{\perp}_{\ell,\tau} x_\ell \,. \tag{2}$$

Eq. (2) applies deflation to the full module input rather than only to the LoRA branch: neither the frozen base weights nor the trainable style update receives the component from the selected content input basis. Note however that the intervention is local and partial: it is applied only to selected attention modules and only removes a k-dimensional subspace, leaving the remaining activation space and other layers available for linguistic processing.

Activation- vs. parameter-space intervention. EDITORS intervenes in the input activation space rather than directly constraining the style LoRA weights. Note that the diagnosed basis $V^{(c)}_{\ell,\tau}$ is in the input activation space where content-oriented variation was analyzed, and not the parameter space of the style adapter unlike past works[55, 13, 6]. In contrast, a constraint such as $\Delta W^{(s)}_{\ell,\tau} \perp \Delta W^{(c)}_{\ell,\tau}$ would only impose a parameter-space relation between two matrices and not ensure that the style adapter avoids content-correlated directions in the actual input distribution. Deflating activations constrains the signal available during training, forcing the style adapter to learn from the residual representation from the start rather than have the style adapter first learn an entangled mixture of content and style before trying to reduce content influence post hoc. We empirically justify this design choice in App. I.7.

3.4 Task-Specific Training

EDITORS can be applied to both AA and SI tasks. The main *diagnose-then-deflate* procedure above is shared by AA and SI: In both settings, the same deflation operators $P^{\perp}_{\ell,\tau}$ are used during training. The deployed EDITORS model consists of the frozen base model W_0, the deflation operators $P^{\perp}$, and only the final style adapter $\Delta W^{(s)}$. The difference lies only in what supervision the model receives and how the final style adapter is initialized. We will describe the details below.

[4]More background on deflation can be found in App. C.2.

Table 1: Lifecycles of objects in EDITORS. The content adapter $\Delta W^{(c)}$ is discarded after extracting the operator $P^{\perp}$. The deployed model uses the frozen base W, $P^{\perp}$, and the style adapter $\Delta W^{(s)}$.

Object	Trained on	Purpose	Used at inference?	Initialization
W	Pretraining	Frozen base model	Yes	–
$\Delta W^{(0)}$	$\mathcal{D}$ (optional)	Domain/task warm-up	No	$\mathbf{0}$
$\Delta W^{(c)}$	Style-neutral $\mathcal{D}^{(c)}$	Diagnose content directions	No	$\mathbf{0}$
$P^{\perp}$	Derived from $\Delta W^{(c)}$	Deflate content directions	Yes	SVD of $\Delta W^{(c)}$
$\Delta W^{(s)}$	$\mathcal{D}$ (under deflation)	Final style adapter	Yes	Task-dependent

Authorship attribution (AA). The adapter is trained with a classification objective: Given a passage $t_i \in \mathcal{T}$, the model is trained to output its author label $a_i \in \mathcal{A}$. During inference for AA, the model is used to predict an author label from an input passage.

To initialize the style adapter, we first perform domain/task warm-up by training a separate LoRA $\Delta W^{(0)}$ on the original training corpus $\mathcal{D}$ for a few steps. We then initialize the style adapter from this domain-adapted adapter: $\Delta W^{(s)}_{\ell,\tau} = \Delta W^{(0)}_{\ell,\tau}$, and then train it on $\mathcal{D}$ using the deflated forward pass in Sec. 3.3. Specifically, the domain warm-up initialization preserves useful task and corpus information (see ablation in App. I.1 and Tab. 9) which aligns with broader findings in training data mixture optimization and domain adaption[9, 58], while the deflated training reduces access to the dominant content-oriented directions.

Style imitation (SI). The adapter is trained on $\mathcal{D}$ with an author-conditioned language modeling objective: given the target author $a_i \in \mathcal{A}$, the model is trained to generate the corresponding passage $t_i \in \mathcal{T}$. During inference for SI, the model receives a target author a and a content specification $\ddot{c}$ (for content-controlled SI), and generates a passage conditioned on them.

For SI, we instead initialize the style adapter from zero: $\Delta W^{(s)}_{\ell,\tau} = \mathbf{0}$. There is no domain warm-up initialization like AA because an adapter first trained on the target authors' original passages may encode author-specific topics, entities, or narrative patterns. Such content patterns can increase memorization risk and weaken control by the user-provided content specification $\ddot{c}$, as we show empirically in Sec. 4.2. For readers' convenience, we summarize all objects that appear in the EDITORS pipeline in Tab. 1.

4 Experiments

We evaluate EDITORS on AA (Sec. 4.1) and SI (Sec. 4.2). For main LoRA-based experiments, we target the attention query, key, and value projections with rank $r = 8$, scaling factor $\alpha = 16$, and dropout 0.1. Full implementation settings, more details and results are in App. E-I.

4.1 Authorship Attribution (AA)

Datasets & evaluation metric. We evaluate on three closed-set AA benchmarks spanning distinct domains: **CCAT10/50** [40], **Blogs10/50** [53], and **Gutenberg30** [32], covering news, blog, and literary domains. Further details of datasets are provided in App. E. We use classification accuracy (i.e., $(1/N)\sum_{i=1}^{N} \mathbf{1}[\hat{a}_i = a_i]$) as the evaluation metric (more in App. H.1).

Base LLMs & baselines. Our main backbone is Llama-3.1-8B-Instruct [17]. To evaluate whether EDITORS generalizes across architectures, we also test on BERT [11] (without LoRA), Phi-4-mini-Instruct [2], Mistral-7B-Instruct-v0.3 [29], Gemma2-9B-it [15], Qwen3.5-9B [48] and Gemma-4-E4B-it [14]. We compare against representative baselines from three families: *SFT methods* (**BertAA** [12] and **STAR** [27]), *pure style embedding methods* (**StyleEmbedding** [57] and **StyleDistance** [45]), and *prompt-based LLM attribution* (**GPT-4, GPT-5 nano, Qwen3.5-27B [48] + LIP** [23]). We also include a plain SFT baseline without deflation to isolate the contribution of EDITORS. Implementation details and inference prompts are provided in App. F.1 and G.2.

Main results. We first compare EDITORS on five benchmarks with AA baselines, along with an additional naïve baseline that we introduced – an LLM finetuned on the benchmark's respective training data (Llama-3.1-8B SFT). Each of the benchmarks evaluates methods on different types of

Table 2: AA accuracy (%). **Bold** and underline mark the best and second-best values. Some results for GPT-4 + LIP are marked with – because providing necessary reference passages for ≥ 30 authors exceeds its context window.

Method	CCAT10	CCAT50	Blogs10	Blogs50	Gutenberg30
BertAA	84.00	65.72	65.40	59.69	42.76
STAR	91.75	67.33	73.40	65.43	53.30
StyleEmbedding	18.20	4.28	40.67	15.60	5.00
StyleDistance	20.60	8.00	52.33	15.40	11.67
GPT-4 + LIP	57.80	–	68.00	–	–
GPT-5-nano + LIP	57.00	–	35.18	–	–
Qwen3.5-27B + LIP	57.00	–	38.16	–	–
Llama-3.1-8B SFT	86.00	67.33	93.67	65.33	75.67
EDITORS	**96.33**	**80.40**	**95.67**	**83.80**	**82.33**

challenges associated with AA. CCAT50 is a topic-controlled news benchmark where authors often discuss overlapping corporate events and entities, making coarse topical cues less reliable. Blogs50 is noisy and naturally imbalanced, testing robustness to uneven author coverage and informal writing. Gutenberg30 is a long-form literary benchmark where AA depends on discourse-level style cues that are difficult for short-context encoder baselines to capture.

As seen in Tab. 2, EDITORS achieves the best accuracy on all five benchmarks, outperforming all baselines by a significant margin. Note that our newly introduced SFT baseline (Llama SFT) surprisingly also performs very competitively against baselines, indicating that several of the baseline methods may have specialized approaches that might have disadvantaged them in achieving generalizable performance, unlike EDITORS which consistently outperforms across benchmarks. EDITORS's gains are especially large on more challenging 50-author settings, increasing accuracy from 67.33% to 80.40% on CCAT50 and from 65.33% to 83.80% on Blogs50. As the candidate author set grows, authors are more likely to share overlapping topics. EDITORS's approach of deflating content-oriented directions may have supported its strong performance in such realistic scenarios with more overlapping content and the model must rely more heavily on style cues.

The inconsistent performance of baselines may be indicative of challenges around their method's generalizability. For example, BertAA is competitive on shorter news and blog texts but performs poorly on Gutenberg30, likely because long literary passages exceed the effective input constraints of BERT-style encoders. STAR, which is pretrained on large datasets, does not consistently scale across domains and larger candidate sets. StyleEmbedding and StyleDistance underperform broadly, suggesting that aggressively removing content loses useful author-discriminative information. GPT-4 + LIP is limited to small candidate sets because reference-passage prompting does not scale to 30-50 authors. While our SFT baseline (Llama SFT) is surprisingly robust, it faces challenges in generalizing beyond its training dataset when required to handle larger candidate sets.

Generalization across base LLMs. Tab. 3 shows that EDITORS consistently improves over our SFT baseline across all backbone LLMs and benchmarks (including the encoder-only BERT backbone). Gemma2-9B benefits most with a gain of $+19.0\%$, though with higher variance across datasets (σ_Δ=10.5), suggesting the degree of style-content entanglement in Gemma2 varies substantially by domain. Phi4-mini-3.8B, as the most compact LLM base model, shows the smallest and most uniform gain ($+3.1\%$, σ_Δ=2.2). In general for LLM base models, EDITORS' performance gain over SFT seems to be correlated with model size, though the gains vary fairly widely. This might suggest LLM backbones having different extent of entangled content and style features in their internal representations, and compact models have less redundant capacity for deflation to exploit. Despite these differences, EDITORS achieves consistent gains, suggesting its generalizability.

4.2 Style Imitation (SI)

We consider two settings from Sec. 2: (1) SI, where the model only conditions on the target author a, and (2) content-controlled SI, where it must also follow a user-provided content specification $\ddot{c}$. Detailed prompts are provided in App. G.3.

Table 3: AA accuracy (%) across backbone architectures. Δ_{avg} denotes the average absolute improvement over plain SFT across the five benchmarks; σ_{Δ} is the standard deviation across benchmarks. For BERT, the baseline is BertAA; for LLM backbones, the baseline is plain LoRA SFT.

Backbone	Setting	CCAT10	CCAT50	Blogs10	Blogs50	Gutenberg30	Δ_{avg}	σ_{Δ}
BERT	BertAA	84.00	65.72	65.40	59.69	42.76	–	–
	EDITORS	**90.75**	**77.47**	**92.68**	**83.39**	**43.57**	+14.1	10.0
Phi-4-mini-3.8B	SFT	93.70	69.07	91.25	71.60	48.03	–	–
	EDITORS	**95.70**	**70.40**	**92.08**	**77.74**	**53.37**	+3.1	2.2
Mistral-7B	SFT	93.00	80.53	94.17	75.79	72.37	–	–
	EDITORS	**97.25**	**86.80**	**95.77**	**85.65**	**73.00**	+4.5	3.3
Llama-3.1-8B	SFT	86.00	67.33	93.67	65.33	75.67	–	–
	EDITORS	**96.33**	**80.40**	**95.67**	**83.80**	**82.33**	+10.1	6.3
Gemma2-9B	SFT	85.30	53.87	64.52	50.40	61.53	–	–
	EDITORS	**96.30**	**74.53**	**93.27**	**81.61**	**65.10**	+19.0	10.5
Qwen3.5-9B	SFT	52.50	74.40	93.37	73.36	54.93	–	–
	EDITORS	**92.50**	**75.00**	**95.18**	**83.80**	**57.83**	+11.2	14.8
Gemma-4-E4B-it	SFT	84.33	60.80	67.27	52.26	46.33	–	–
	EDITORS	**92.75**	**74.33**	**94.05**	**72.13**	**47.90**	+14.0	8.8

Table 4: SI results on Gutenberg30. Higher is better for style similarity and lower is better for BLEU and Self-BLEU.

Model	Style Similarity	BLEU	Self-BLEU
GPT-4	23.93±8.55	**4.42**±1.20	27.35±3.49
StyleTunedLM	53.39±9.43	40.38±5.42	37.48±5.60
EDITORS	**56.50**±8.74	27.09±4.76	**24.35**±4.90

Dataset & evaluation metric. We focus on **Gutenberg30** for SI since it exhibits the clearest authorial voice and richest narrative structure among the benchmarks. Specific LoRA configuration can be found in App. F.2. We evaluate SI along the following axes:

- **Attribution-based style similarity.** We use an ensemble of AA classifiers, including the best performing existing AA methods BertAA and STAR, along with Gemma2-, Llama-, Phi4-, and Mistral-based EDITORS classifiers (from Sec. 4.1), to measure whether each generation can be attributed to the intended author. We found that while each AA classifier might give different absolute scores, all provide consistent rankings when evaluating SI methods (e.g., see Fig. 2).
- **Diversity.** We report BLEU-4 [44, 50] against the target author's training corpus, lower BLEU indicates less training-set copying. We also report Self-BLEU, which measures content overlap among same-author generations to assess whether a model repeatedly generates similar author-specific content (lower is better).
- **Preservation of user-provided content** (for (2) content-controlled SI)**.** We use an ensemble of GPT-4o mini and Qwen3.5-27B to measure content consistency between generated text $\hat{t}$ and user-provided content specification $\ddot{c}$.

App. H.2 provides the detailed metrics, calculation protocols, threshold settings and judge prompts.

Base LLMs & baselines. We evaluate **EDITORS** with Llama-3.1-8B-Instruct as the base model. We compare against four representative SI baselines: **GPT-4 prompting** [56] for tuning-free prompting imitation, **StyleTunedLM** [39] for direct author-corpus fine-tuning, **ASTRAPOP** [38] for RL/preference-based style transfer and **StoryTrans** [59] for non-parallel rewriting. Since ASTRAPOP and StoryTrans require an explicit content input, they are only evaluated in (2) content-controlled SI.

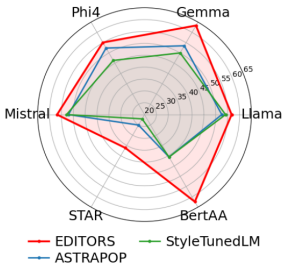

Figure 2: All AA evaluators give consistent judgements.

SI. Tab. 4 shows the results for (1) SI without explicit content constraints. EDITORS achieves the highest style similarity while substantially reducing BLEU and Self-BLEU compared with Style-

Table 5: Content-controlled SI results on Gutenberg30. *Content preservation* refers to how much the generated passages adhere to the user-provided content specifications $\ddot{c}$. Higher is better for style similarity and content preservation, while lower BLEU and Self-BLEU indicate less overlap with the training corpus and greater output diversity, respectively.

Model	Style Similarity	Content Preservation	BLEU	Self-BLEU
GPT-4	10.82±0.47	**99.88**±0.06	16.54±0.12	**11.03**±0.13
StoryTrans	13.44±0.52	0.68±1.12	**4.96**±0.95	14.36±0.38
ASTRAPOP	46.03±0.56	3.73±0.49	27.08±0.30	14.85±0.50
StyleTunedLM	44.34±0.97	59.63±1.34	47.18±0.16	20.54±0.39
EDITORS	**55.08**±0.56	68.16±1.27	27.42±0.64	13.14±0.55

Target author: Hamlin Garland. **Content:** A group works together to survive after getting stranded.

StyleTunedLM (style votes: $2/6$)

"We were happy. We were together. The rain was beating upon us, but we were warm. The night was dark and cold, but we were together. The fire was almost out, but we were together. I was cold, but I was not unhappy. ... We were all that was left of our world. I reached up and grasped Vi's hand, and she did not pull away. ..."

ASTRAPOP (content: $1/5$)

"Half a million people will hear of it," continued Mrs. Lambert. "The whole country will ring with it." "But she has not consented—" began Serviss, only to have Clarke cut him off with a gesture of contempt. "She will consent when she sees how the thing is coming out. We have her in the hollow of our hand. ..."

EDITORS (style votes: $6/6$; content: $5/5$)

"We were all in this venture together, and every man was willing to risk his life for the sake of the others ... In this matter the two Canadians were very shrewd and took a leading part in the discussion, while the three of us who were Americans were content to take the back seats and let them do the talking. ... At this time I was a tramp, and I had no home to go to, but I was a tramp who had a home."

Figure 3: Content-controlled SI generation excerpts. Style votes count AA evaluators predicting the target author; content is rated on a 1–5 scale.

TunedLM, showing a better balance between style recognizability, memorization risk, and generation diversity.

Content-controlled SI. Tab. 5 further evaluates whether models can imitate author styles well while preserving a user-provided content specification $\ddot{c}$ in (2) content-controlled SI. EDITORS achieves the highest style similarity (consistent ranking across all the six evaluators, see Fig. 2) while GPT-4 and StoryTrans fail this primary goal. EDITORS preserves user-provided content specification $\ddot{c}$ (while ASTRAPOP does not), copies less from training corpus (while StyleTunedLM has a larger overlap with it/BLEU), while also achieving better diversity (lower Self-BLEU). More comprehensive results from independent style evaluators, additional content preservation assessments, character-level overlap audits, and SI evaluation on Blogs10 are provided in App. I.4 (see Tabs. 12 to 14 and 16). All results demonstrates EDITORS' effectiveness in achieving balance among style similarity, content preservation and memorization of training content.

Qualitative analysis. Fig. 3 compares generations under the same target author and content specification. EDITORS preserves the survival content condition, receiving a content score of $5/5$, while all six AA evaluators recognize its generation as Hamlin Garland. The passage expresses Garland-like social observation and plain first-person narration within a survival setting. StyleTunedLM repeatedly uses "we were together," and only two of six evaluators recognize its generation as the target author. ASTRAPOP develops a dialogue unrelated to the specified survival scenario and receives a content score of $1/5$.

4.3 Ablation Studies

In Sec. 3.4, we describe an optional domain warm-up that trains an adapter $\Delta W^{(0)}$ to initialize the style adapter $\Delta W^{(s)}$. This warm-up is removed for SI since it may cause the model to memorize

Table 6: Comparison of EDITORS with and without domain warm-up $\Delta W^{(0)}$ in content-controlled SI. Refer to Tab. 10 for the comparison in general SI task.

Model	Style Similarity	Content Preservation	BLEU
EDITORS w/ $\Delta W^{(0)}$	54.10±0.68	39.23±2.54	45.64±0.26
EDITORS w/o $\Delta W^{(0)}$	**55.08**±0.56	**68.16**±1.27	**27.42**±0.64

training content or disregard user-provided content, as empirically validated in Tab. 6 (for content-controlled SI) and Tab. 10 (for general SI). With and without $\Delta W^{(0)}$, EDITORS achieves similar style similarity, suggesting that $\Delta W^{(0)}$ is not pivotal for inducing style. However, initializing from $\Delta W^{(0)}$ lowers content preservation by $\sim$30 and increases BLEU by $\sim$20. This suggests that marginal improvements in style similarity for EDITORS w/ $\Delta W^{(0)}$ may be due more to reproducing author's training passages than successful style transfer. In contrast, EDITORS w/o $\Delta W^{(0)}$, our design choice for SI, preserves the user-provided content more faithfully while maintaining comparable style similarity and reducing training-corpus overlap.

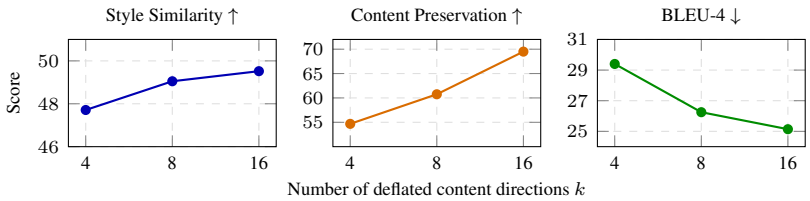

Figure 4: Sensitivity to the selected content-basis dimension k for SI. Increasing k improves content preservation and reduces BLEU against the training corpus, while style similarity is relatively stable.

In Fig. 4, we investigate how the retained content-basis dimension k (Eq. (1)) affects SI. As k grows from 4 to 16, content preservation increases from 54.68 to 69.50, while BLEU against the target author's training corpus decreases from 29.40 to 25.14. This suggests that removing a larger portion of the diagnosed content subspace helps the model avoid reproducing training-passage content and better adapt to the user-provided content specification $\ddot{c}$. Style similarity changes only modestly across k, but the higher BLEU at smaller k indicates that part of the apparent stylistic recognizability may come from training-corpus overlap rather than successful style transfer. In contrast, $k = 16$ achieves the best content preservation and lowest BLEU while maintaining comparable style similarity, suggesting a better balance between style imitation and content control. App. I.2 shows a similar effect for AA.

5 Conclusion

Existing methods often entangle author style with semantic content, causing AA to rely on content, SI to reproduce training passages, and content-controlled SI to ignore the user-specified content. Our proposed method EDITORS successfully prioritizes stylistic features without discarding semantic information aggressively by diagnosing a content-oriented subspace and learning style under activation-space deflation, so that selected content-oriented directions are suppressed during training. We demonstrate that EDITORS is highly effective on various AA and (content-controlled) SI benchmarks. Our results show that deflation behaves as intended: it suppresses content without destroying general semantic information, and activation-space deflation outperforms parameter-space deflation.

Limitation and future work. Due to the lack of real-world datasets where the same content is expressed in different author styles, we did not provide a more direct ablation controlling for style-specific attribution. We also did not consider uncertainty quantification [30, 35] of our attributions, which would be valuable in practical deployment. Although we take the ensemble of classifiers to reduce the SI evaluator bias, the evaluation still relies on automatic evaluators which may not fully capture the real author style unlike human expert evaluations. Future work should therefore focus on building stronger benchmarks for AA with better control over semantic content, and on developing more direct evaluations of how much attribution models rely on style and content respectively.

Acknowledgments

This research/project is supported by the National Research Foundation, Singapore under its AI Singapore Programme (AISG Award No: AISG2-PhD/2023-01-039J) and is part of the programme DesCartes. DesCartes: this research is supported by the National Research Foundation, Prime Minister's Office, Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) programme. This project is partially funded by Microsoft Research Asia. Xiao Tian and Jue Fan are supported by Agency for Science, Technology and Research (A*STAR) Graduate Academy.

References

[1] Hervé Abdi. Singular value decomposition (SVD) and generalized singular value decomposition. In Neil J. Salkind, editor, *Encyclopedia of Measurement and Statistics*, pages 907–912. Sage Publications, Thousand Oaks, CA, 2007.

[2] Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, Dong Chen, Dongdong Chen, Junkun Chen, Weizhu Chen, Yen-Chun Chen, Yi ling Chen, Qi Dai, Xiyang Dai, Ruchao Fan, Mei Gao, Min Gao, Amit Garg, Abhishek Goswami, Junheng Hao, Amr Hendy, Yuxuan Hu, Xin Jin, Mahmoud Khademi, Dongwoo Kim, Young Jin Kim, Gina Lee, Jinyu Li, Yunsheng Li, Chen Liang, Xihui Lin, Zeqi Lin, Mengchen Liu, Yang Liu, Gilsinia Lopez, Chong Luo, Piyush Madan, Vadim Mazalov, Arindam Mitra, Ali Mousavi, Anh Nguyen, Jing Pan, Daniel Perez-Becker, Jacob Platin, Thomas Portet, Kai Qiu, Bo Ren, Liliang Ren, Sambuddha Roy, Ning Shang, Yelong Shen, Saksham Singhal, Subhojit Som, Xia Song, Tetyana Sych, Praneetha Vaddamanu, Shuohang Wang, Yiming Wang, Zhenghao Wang, Haibin Wu, Haoran Xu, Weijian Xu, Yifan Yang, Ziyi Yang, Donghan Yu, Ishmam Zabir, Jianwen Zhang, Li Lyna Zhang, Yunan Zhang, and Xiren Zhou. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-LoRAs. *arXiv preprint arXiv:2503.01743*, 2025.

[3] Michael Aerni, Javier Rando, Edoardo Debenedetti, Nicholas Carlini, Daphne Ippolito, and Florian Tramèr. Measuring non-adversarial reproduction of training data in large language models. In *The Thirteenth International Conference on Learning Representations*, 2025.

[4] Simran Arora, Avanika Narayan, Mayee F. Chen, Laurel Orr, Neel Guha, Kush Bhatia, Ines Chami, and Christopher Ré. Ask me anything: A simple strategy for prompting language models. In *The Eleventh International Conference on Learning Representations*, 2023.

[5] Yuchen Cai and Ding Cao. O-Edit: Orthogonal subspace editing for language model sequential editing. *arXiv preprint arXiv:2410.11469*, 2024.

[6] Chuheng Chen, Xiaofei Zhou, Geyuan Zhang, and Yong Huang. NP-LoRA: Null space projection unifies subject and style in LoRA fusion. *arXiv preprint arXiv:2511.11051*, 2026.

[7] Xiang Chen, Xin Xie, Zhen Bi, Hongbin Ye, Shumin Deng, Ningyu Zhang, and Huajun Chen. Disentangled contrastive learning for learning robust textual representations. *arXiv preprint arXiv:2104.04907*, 2021.

[8] Yanran Chen and Steffen Eger. MENLI: Robust evaluation metrics from natural language inference. *Transactions of the Association for Computational Linguistics*, 11:804–825, 2023.

[9] Zhiliang Chen, Gregory Kang Ruey Lau, Chuan-Sheng Foo, and Bryan Kian Hsiang Low. DUET: Optimizing training data mixtures via feedback from unseen evaluation tasks. In *The Fourteenth International Conference on Learning Representations*, 2026.

[10] Hyundong Justin Cho, Karishma Sharma, Nicolaas Paul Jedema, Leonardo F. R. Ribeiro, Jonathan May, and Alessandro Moschitti. Tuning-free personalized alignment via trial-error-explain in-context learning. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 5864–5885, 2025.

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019.

[12] Maël Fabien, Esau Villatoro-Tello, Petr Motlicek, and Shantipriya Parida. BertAA : BERT fine-tuning for authorship attribution. In *Proceedings of the 17th International Conference on Natural Language Processing (ICON)*, pages 127–137, 2020.

[13] Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Jie Shi, Xiang Wang, Xiangnan He, and Tat-Seng Chua. AlphaEdit: Null-space constrained knowledge editing for language models. In *The Thirteenth International Conference on Learning Representations*, 2025.

[14] Gemma Team. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026.

[15] Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy

Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozińska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucińska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Görner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Cogan, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomas Kocisky, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

[16] Helena Montserrat Gómez-Adorno, Juan-Pablo Posadas-Durán, Germán Ríos-Toledo, Grigori Sidorov, and Gerardo Sierra. Stylometry-based approach for detecting writing style changes in literary texts. *Computación y Sistemas*, 22(1):47–53, 2018.

[17] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen

Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[18] Tommi Gröndahl and N. Asokan. Effective writing style imitation via combinatorial paraphrasing. *Proceedings on Privacy Enhancing Technologies*, 2020(4):253–271, 2020.

[19] Skyler Hallinan, Faeze Brahman, Ximing Lu, Jaehun Jung, Sean Welleck, and Yejin Choi. STEER: Unified style transfer with expert reinforcement. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7546–7562, 2023.

[20] John Russell Himawan, Gregory Kang Ruey Lau, and Bryan Kian Hsiang Low. Watershed: A unified benchmark for end-to-end data provenance evaluation. In *Trustworthy AI for Good (AI4GOOD) Workshop at ICML 2026*, 2026.

[21] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[22] Zhiqiang Hu, Thao Thanh Nguyen, Yujia Hu, Chia-Yu Hung, Ming Shan Hee, Chun Wei Seah, and Roy Ka-Wei Lee. Contrastive disentanglement for authorship attribution. In *Companion Proceedings of the ACM on Web Conference 2024*, pages 1657–1666, 2024.

[23] Baixiang Huang, Canyu Chen, and Kai Shu. Can large language models identify authorship? In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 445–460, 2024.

[24] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 172–189, 2018.

[25] Javier Huertas-Tato, Adrián Girón-Jiménez, Alejandro Martín, and David Camacho. Isolating authorship from content with semantic embeddings and contrastive learning. *arXiv preprint arXiv:2411.18472*, 2024.

[26] Javier Huertas-Tato, Álvaro Huertas-García, Alejandro Martín, and David Camacho. PART: Pre-trained authorship representation transformer. *Human-centric Computing and Information Sciences*, 14:39, 2024.

[27] Javier Huertas-Tato, Alejandro Martín, and David Camacho. Understanding writing style in social media with a supervised contrastively pre-trained transformer. *Knowledge-Based Systems*, 296:111867, 2024.

[28] Harsh Jhamtani, Varun Gangal, Eduard Hovy, and Eric Nyberg. Shakespearizing modern language using copy-enriched sequence to sequence models. In *Proceedings of the Workshop on Stylistic Variation*, pages 10–19, 2017.

[29] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023.

[30] Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*, 2023.

[31] Gihyun Kwon and Jong Chul Ye. CLIPstyler: Image style transfer with a single text condition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18062–18071, 2022.

[32] Shibamouli Lahiri. Complexity of word collocation networks: A preliminary structural analysis. In *Proceedings of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 96–105, 2014.

[33] Gregory Kang Ruey Lau, Wenyang Hu, Diwen Liu, Jizhuo Chen, See-Kiong Ng, and Bryan Kian Hsiang Low. DIPPER: Diversity in prompts for producing large language model ensembles in reasoning tasks. In *NeurIPS 2024 Workshop on Foundation Model Interventions (MINT)*, 2024.

[34] Gregory Kang Ruey Lau, Xinyuan Niu, Hieu Dao, Jiangwei Chen, Chuan-Sheng Foo, and Bryan Kian Hsiang Low. Waterfall: Scalable framework for robust text watermarking and provenance for LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 20432–20466, 2024.

[35] Gregory Kang Ruey Lau, Hieu Dao, Nicole Kan Hui Lin, and Bryan Kian Hsiang Low. UMPIRE: Uncertainty quantification for multimodal large language models with incoherence-adjusted semantic volume. In *Advances in Neural Information Processing Systems*, 2026.

[36] Jey Han Lau, Trevor Cohn, Timothy Baldwin, Julian Brooke, and Adam Hammond. Deep-speare: A joint neural model of poetic language, meter and rhyme. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1948–1958, 2018.

[37] Shuai Liu and Jonathan May. Style transfer with multi-iteration preference optimization. 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 2663–2681, 2025.

[38] Shuai Liu, Shantanu Agarwal, and Jonathan May. Authorship style transfer with policy optimization. *arXiv preprint arXiv:2403.08043*, 2024.

[39] Xinyue Liu, Harshita Diddee, and Daphne Ippolito. Customizing large language model generation style using parameter-efficient finetuning. In *Proceedings of the 17th International Natural Language Generation Conference*, pages 412–426, 2024.

[40] Zhi Liu. Reuter_50_50. UCI Machine Learning Repository, 2006. DOI: https://doi.org/10.24432/C5DS42.

[41] Xinyang Lu, Xinyuan Niu, Gregory Kang Ruey Lau, Bui Thi Cam Nhung, Rachael Hwee Ling Sim, John Russell Himawan, Fanyu Wen, Chuan-Sheng Foo, See-Kiong Ng, and Bryan Kian Hsiang Low. WaterDrum: Watermarking for data-centric unlearning metric. In *The Fourteenth International Conference on Learning Representations*, 2026.

[42] Fuli Luo, Peng Li, Jie Zhou, Pengcheng Yang, Baobao Chang, Zhifang Sui, and Xu Sun. A dual reinforcement learning framework for unsupervised text style transfer. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5116–5122, 2019.

[43] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2I-Adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(5):4296–4304, 2024.

[44] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[45] Ajay Patel, Jiacheng Zhu, Justin Qiu, Zachary Horvitz, Marianna Apidianaki, Kathleen McKeown, and Chris Callison-Burch. StyleDistance: Stronger content-independent style embeddings with synthetic parallel examples. 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 8662–8685, 2025.

[46] Konpat Preechakul, Nattanat Chatthee, Suttisak Wizadwongsa, and Supasorn Suwajanakorn. Diffusion autoencoders: Toward a meaningful and decodable representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10619–10629, 2022.

[47] Yifu Qiu, Zheng Zhao, Yftah Ziser, Anna Korhonen, Edoardo M. Ponti, and Shay B. Cohen. Spectral editing of activations for large language model alignment. In *Advances in Neural Information Processing Systems*, volume 37, pages 56958–56987, 2024.

[48] Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

[49] Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. Null it out: Guarding protected attributes by iterative nullspace projection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7237–7256, 2020.

[50] Piotr Sawicki, Marek Grześ, Anna Jordanous, Dan Brown, and Max Peeperkorn. Training GPT-2 to represent two romantic-era authors: Challenges, evaluations and pitfalls. In *Proceedings of the 13th International Conference on Computational Creativity*, pages 34–43, 2022.

[51] Kihyuk Sohn, Lu Jiang, Jarred Barber, Kimin Lee, Nataniel Ruiz, Dilip Krishnan, Huiwen Chang, Yuanzhen Li, Irfan Essa, Michael Rubinstein, Yuan Hao, Glenn Entis, Irina Blok, and Daniel Castro Chin. StyleDrop: Text-to-image synthesis of any style. In *Advances in Neural Information Processing Systems*, volume 36, pages 66860–66889, 2023.

[52] Bakhtiyar Syed, Gaurav Verma, Balaji Srinivasan, Anandhavelu Natarajan, and Vasudeva Varma. Adapting language models for non-parallel author-stylized rewriting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):9008–9015, 2020.

[53] Jacob Tyo, Bhuwan Dhingra, and Zachary C. Lipton. Valla: Standardizing and benchmarking authorship attribution and verification through empirical evaluation and comparative analysis. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 649–660, 2023.

[54] Cheng Wang, Xinyang Lu, See-Kiong Ng, and Bryan Kian Hsiang Low. TRACE: TRansformer-based Attribution using Contrastive Embeddings in LLMs. *arXiv preprint arXiv:2407.04981*, 2024.

[55] Xiao Wang, Tianze Chen, Qiming Ge, Han Xia, Rong Bao, Rui Zheng, Qi Zhang, Tao Gui, and Xuanjing Huang. Orthogonal subspace learning for language model continual learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10658–10671, 2023.

[56] Zhengxiang Wang, Nafis Irtiza Tripto, Solha Park, Zhenzhen Li, and Jiawei Zhou. Catch me if you can? not yet: LLMs still struggle to imitate the implicit writing styles of everyday authors. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 10040–10055, 2025.

[57] Anna Wegmann, Marijn Schraagen, and Dong Nguyen. Same author or just same topic? towards content-independent style representations. In *Proceedings of the 7th Workshop on Representation Learning for NLP*, pages 249–268, 2022.

[58] Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. DoReMi: Optimizing data mixtures speeds up language model pretraining. In *Advances in Neural Information Processing Systems*, volume 36, pages 69798–69818, 2023.

[59] Xuekai Zhu, Jian Guan, Minlie Huang, and Juan Liu. StoryTrans: Non-parallel story author-style transfer with discourse representations and content enhancing. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14803–14819, 2023.

[60] Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. Texygen: A benchmarking platform for text generation models. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 1097–1100, 2018.

A Ethical Considerations

AA and SI may be misused for deanonymization, surveillance, unsupported authorship accusations, or impersonation. Our results should therefore be interpreted as model-based evidence rather than definitive authorship proof. Practical deployment should consider uncertainty quantification of attributions to support more holistic investigation. For SI, generated text should be clearly disclosed as machine-generated. We encourage future deployment to include appropriate safeguards, such as watermarking generated text and restricting imitation of living authors without consent.

B Algorithm

Alg. 1 summarizes the EDITORS training procedure.

Algorithm 1 EDITORS training.

Input: Passage-author corpus $\mathcal{D} = \{(t_i, a_i)\}_{i=1}^{N}$; frozen pretrained model W_0; target modules $\mathcal{S}$; LoRA rank r; content-subspace dimension k.

Output: Final style adapter $\Delta W^{(s)}$ and deflation operators $P^{\perp}$.

if task is AA **then**
  Train domain warm-up adapter $\Delta W^{(0)}$ on $\mathcal{D}$.
end if
Extract style-neutral statements from passages in $\mathcal{D}$ to construct $\mathcal{D}^{(c)}$.
Train diagnostic content adapter $\Delta W^{(c)}$ on $\mathcal{D}^{(c)}$.
for each layer ℓ and module $\tau \in \mathcal{S}$ **do**
  Compute SVD: $\Delta W_{\ell,\tau}^{(c)} = U_{\ell,\tau}\Sigma_{\ell,\tau}V_{\ell,\tau}^{\top}$.
  Set $V_{\ell,\tau}^{(c)} = V_{\ell,\tau}[:, 1:k]$.
  Construct $P_{\ell,\tau}^{\perp} = I - V_{\ell,\tau}^{(c)}(V_{\ell,\tau}^{(c)})^{\top}$.
end for
if task is AA **then**
  Initialize $\Delta W^{(s)} \leftarrow \Delta W^{(0)}$.
  Train style adapter $\Delta W^{(s)}$ on the AA objective using deflated activations $x_{\ell,\tau}^{\mathrm{def}} = P_{\ell,\tau}^{\perp} x_{\ell}$.
else if task is SI **then**
  Initialize $\Delta W^{(s)} \leftarrow \mathbf{0}$.
  Train $\Delta W^{(s)}$ on the author-conditioned LM objective using deflated activations $x_{\ell,\tau}^{\mathrm{def}} = P_{\ell,\tau}^{\perp} x_{\ell}$.
end if
return $\Delta W^{(s)}$ and $P^{\perp}$.

C Preliminaries

C.1 Low-Rank Adaptation (LoRA)

Fully finetuning the layer weights W_ℓ for every layer ℓ of an LLM could be computationally infeasible. To address this, LoRA [21] freezes the pretrained weights W_ℓ and instead injects a smaller-scale trainable rank-r update. The modified weights at layer ℓ become:

$$W'_\ell = W_\ell + \Delta W_\ell, \quad \text{with } \Delta W_\ell = B_\ell A_\ell,$$

where $A_\ell \in \mathbb{R}^{r_\ell \times d_{\text{in}}}$, $B_\ell \in \mathbb{R}^{d_{\text{out}} \times r_\ell}$ are trainable and $r_\ell \ll \min(d_{\text{in}}, d_{\text{out}})$. The resulting forward computation at layer ℓ is $h_\ell = W'_\ell x_\ell \in \mathbb{R}^{d_{\text{out}}}$, where x_ℓ is the input activation at layer ℓ.

In this paper, we specifically target the multi-head attention (MHA) projections. For a given block ℓ, we focus on the Query, Key, and Value modules and denote $\mathcal{S} = \{q, k, v\}$ as the set of target module types.

C.2 Deflation

Deflation (a.k.a. *orthogonal projection*) is a well-known geometric mechanism to decouple representations and mitigate interference between competing signals. It works by projecting a vector to the closest point in a subspace (i.e., the representation of interest) by removing the component orthogonal to that subspace (i.e., the competing signal). For example, in activation-space deflation, orthogonal projection can efficiently remove information such as gender and topic from embeddings [49]. Qiu et al. [47] edits LLM at inference time by orthogonal projection at hidden activation space to reduce its undesirable behaviors. In the LoRA context, this technique is utilized in continual learning and knowledge editing by operating directly on the weight matrix. Approaches like O-LoRA [55], O-Edit [5] and AlphaEdit [13] constrain LoRA updates (ΔW_ℓ) to be orthogonal to the subspaces of previously learned tasks or preserved knowledge, thereby preventing catastrophic forgetting. Similarly, NP-LoRA [6] prevents interference in LoRA fusion by projecting the weights of a content LoRA onto the null space of a style LoRA's principal directions.

Motivated by these ideas, we aim to use orthogonal projection to reduce interference between content information and stylistic features in texts. Unlike prior weight-centric methods that risk aggressively removing the semantic information, we propose an activation-space deflation mechanism to blind the style adapter to selected semantic signals without directly removing the semantic information.

D Benefits of LoRA to EDITORS

In App. C.1, we have introduced the benefits of LoRA to general LLM finetuning. In this section, we will explain why using LoRA can simplify the computation involved in EDITORS. We also point out that EDITORS can work with any other SFT methods besides LoRA.

LoRA simplifies singular value decomposition (SVD).

Recall from Sec. 3.2 that we need to perform SVD on the content LoRA adapter $\Delta W^{(c)}$ to extract the content-oriented basis. For each content LoRA adapter we obtained, a direct SVD of $\Delta W^{(c)}_{\ell,\tau} = B^{(c)}_{\ell,\tau} A^{(c)}_{\ell,\tau} \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$ would require explicitly forming a dense matrix of the same size as the original linear layer. This is unnecessary as the update is low-rank. We therefore exploit the LoRA factorization to recover the input-side singular subspace from small matrices whose dimension is at most the LoRA rank r.

For simplicity, we omit the layer and module indices (ℓ, τ) and write

$$\Delta W^{(c)} = B^{(c)} A^{(c)}, \qquad B^{(c)} \in \mathbb{R}^{d_{\text{out}} \times r}, \quad A^{(c)} \in \mathbb{R}^{r \times d_{\text{in}}}.$$

The content-oriented basis used in Sec. 3.2 is the input-side right-singular subspace of $\Delta W^{(c)}$. Since some rank directions in $A^{(c)}$ may be annihilated by $B^{(c)}$, we first remove the null directions of $B^{(c)}$. Let $Q_B \in \mathbb{R}^{r \times \rho}$ be an orthonormal basis of $(B^{(c)})^\top$, where $\rho = \operatorname{rank}(B^{(c)}) \le r$. We then define

$$A_{\text{eff}} = Q_B^\top A^{(c)} \in \mathbb{R}^{\rho \times d_{\text{in}}}, \qquad C = B^{(c)} Q_B \in \mathbb{R}^{d_{\text{out}} \times \rho}.$$

Thus,

$$\Delta W^{(c)} = B^{(c)} A^{(c)} = C A_{\text{eff}}.$$

Moreover, C has full column rank, so

$$\operatorname{Row}\left(\Delta W^{(c)}\right) = \operatorname{Row}(A_{\text{eff}}),$$

where $\operatorname{Row}(M)$ denotes the subspace spanned by the rows of a matrix M. Therefore, the nonzero right-singular subspace of $\Delta W^{(c)}$ can be computed from the much smaller effective factor A_{eff}.

To obtain the principal content input basis $V^{(c)}_{\ell,\tau}$ in Eq. (1), we apply SVD to the low-rank effective factor rather than to the full dense update matrix:

$$A_{\text{eff}}^\top = U_A \Sigma_A Z_A^\top, \qquad U_A \in \mathbb{R}^{d_{\text{in}} \times \rho}, \quad \Sigma_A \in \mathbb{R}^{\rho \times \rho}, \quad Z_A \in \mathbb{R}^{\rho \times \rho}.$$

The columns of U_A form an orthonormal basis for the row space of A_{eff}. Since $\Delta W^{(c)} = C A_{\text{eff}}$ and C has full column rank, this is the same input-side subspace spanned by the rows of $\Delta W^{(c)}$. Therefore, we take

$$V^{(c)}_{\ell,\tau} = U_A[:, 1:k] \in \mathbb{R}^{d_{\text{in}} \times k}$$

as the principal content input basis used for activation-space deflation.

In practice, since $\rho \leq r$ and the LoRA rank r is small, this avoids forming the dense matrix $\Delta W^{(c)} \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$. The expensive operation is reduced to SVD-based orthonormalization of the low-rank factor $A_{\text{eff}}^{\top}$, whose rank is at most r. Our implementation uses this SVD-based solver by default, with QR-based orthonormalization as an alternative. When all surviving directions are retained, the SVD and QR solvers yield equivalent deflation subspaces, because the projection depends only on the span of $V_{\ell,\tau}^{(c)}$. The SVD solver is useful when a truncated basis is desired, since it orders the basis directions by the singular values of $A_{\text{eff}}^{\top}$.

EDITORS can be used without LoRA. Despite the above benefits of LoRA to EDITORS, we point out that EDITORS can be used without LoRA too. In Sec. 4.2, we include some experiments that finetune BERT models without using LoRA. EDITORS still achieves a significantly better performance than its baselines such as BertAA.

E Datasets

Gutenberg30 represents the literature domain and is derived from the Project Gutenberg archive [32]. To make the evaluation task more challenging, we select 30 authors who are relatively difficult to recognize, defined as those with the lowest zero-shot classification accuracy under the raw Llama-3-8B-Instruct model among authors who have at least five books available. Each book is segmented into fixed-length passages of roughly 700 words.

Blogs50 represents informal social media writing and is constructed from the Blog Authorship Corpus [53]. This corpus contains 681,288 blog posts written by 19,320 bloggers. We select the 50 bloggers with the largest amount of text from it . In contrast to the other datasets, Blogs50 retains the natural imbalance of the source material, with passage counts varying substantially across authors.

CCAT50 represents the news domain and is built from a subset of Reuters Corpus Volume 1 [40]. We restrict the corpus to the corporate/industrial topic category (CCAT), yielding a dataset of 50 authors with 100 documents per author, for a total of 5,000 documents.

For all three datasets, the training, validation, and test sets are split strictly by book title or news/blogs category, so that passages originating from the same book or category never appear in multiple splits, ensuring no content overlap between splits to the largest extent.

CCAT10 and **Blogs10** are randomly sampled subsets of 10 authors from CCAT50 and Blogs50 respectively. Their split sizes correspond to all passages belonging to the selected 10 authors. No further subsampling of passages is performed.

Tab. 7 summarizes the key statistics of the above datasets.

Table 7: Statistics of the three datasets, including split sizes, the training distribution over authors, and the average passage length. Train Dist. indicates the number of passages per author in the training split. Length denotes the average word count per passage.

		Split Size (Passages)			Train Dist. (Per Author)			Length
Dataset	**Domain**	**Train**	**Val**	**Test**	**Avg**	**Min**	**Max**	**Avg (Words)**
Gutenberg30	Literature	12,000	300	3,000	400	400	400	700
Blogs50	Social Media	42,874	5,346	5,346	643	424	1,967	161
CCAT50	News	4,000	250	750	80	80	80	504

F Implementation

F.1 Authorship Attribution (AA)

All models are loaded with 4-bit NF4 quantization and bfloat16 computation, and trained with gradient checkpointing. We attach LoRA adapters to the query, key, and value projection modules, using rank r=8, scaling factor α=16, and dropout 0.1. The diagnostic content adapter $\Delta W^{(c)}$ is trained on the style-neutral statements corpus (App. G.1) via continued instruction tuning with TRL's `SFTTrainer`.

The final style adapter $\Delta W^{(s)}$ then uses the original passage-author supervision, formatted as an instruction-style prompt that asks the model to identify the author based on writing style.

The diagnostic content adapter $\Delta W^{(c)}$ is trained for 2 epochs with datasets of style-neutral statements. For CCAT, the domain warm-up $\Delta W^{(0)}$ (described in Sec. 3.4) is trained for 2 epochs and the style adapter $\Delta W^{(s)}$ for 4 epochs. For Blogs and Gutenberg, the domain warm-up $\Delta W^{(0)}$ is trained for 1 epoch and the style adapter $\Delta W^{(s)}$ for 2 epochs. We use a maximum sequence length of $1{,}000$ for both stages across all datasets. Both stages use learning rate 2×10^{-4}, warm-up ratio 0.05, weight decay 0.01, per-device batch size 12, and gradient accumulation 2. For the plain LoRA SFT baseline, we use the same total number of training epochs as the corresponding diagnose-then-deflate EDITORS configuration. All backbone LLMs are trained with the same hyperparameters.

We do not report error bars for AA because the evaluation is deterministic: we do not use stochastic sampling/sampling-based decoding, so repeated runs of the same checkpoint on the same split produce identical predictions.

F.2 Style Imitation (SI)

We use the same LoRA configuration as in the AA experiments. The main difference lies in the supervision format. The diagnostic content adapter $\Delta W^{(c)}$ is reused from the AA setting, and the deflated stage then performs style fine-tuning under activation-space deflation. We evaluate two variants of our method. The default setting skips domain warm-up and trains the style adapter $\Delta W^{(s)}$ directly for 8 epochs. The second variant first applies domain warm-up for 4 epochs and then performs deflated style learning for another 4 epochs, so that the total training budget remains comparable across settings.

At inference time, each of the 30 Gutenberg authors is paired with the same seven stylistically neutral content summaries listed in App. G.3. We generate one passage for each author on the given summary, yielding $30 \times 7 = 210$ generations for evaluation.

G Prompts

G.1 Style-Neutral Statements Extraction

We prompt GPT-4o mini to extract style-neutral statements from each passage (to be used in Sec. 3.2). The extracted facts are used as the continued pretraining corpus for the diagnostic content adapter $\Delta W^{(c)}$.

```
System:
You turn passages into neutral, styleless atomic facts.  Each fact should
stand alone, use plain prose, and stay faithful to the passage.

User:
Rewrite this passage as {min_facts} to {max_facts} standalone atomic factual
statements.
Capture what the passage claims (opinions become factual claims about the
author/source).
Keep names and entities unchanged; avoid figurative language.
Output ONLY a JSON array of strings, nothing else.
PASSAGE: {passage}
```

An example of the original passage and its corresponding style-neutral statements is shown in Figure 5.

Original literary passage (Edward Phillips Oppenheim, *Jeanne of the Marshes*)

"... ever since those few days down at that tumbledown old place of Cecil de la Borne's, she seems to have developed in a remarkable manner ... I am sure ... that we are going to have trouble." Forrest replied that, so far as he was concerned, the trouble had already arrived, and admitted that he felt exhausted by the life he had been leading. The Princess accused him of cowardice and suggested that he was no longer the man he had once been. Their conversation then turned to the De la Borne money, to De Brensault's intentions toward Jeanne, and to the possibility that Jeanne's obstinacy would bring consequences.

Extracted style-neutral statements

(1) The Princess noted that Jeanne had developed in a remarkable manner after spending time at Cecil de la Borne's place.
(2) Forrest said that trouble had already arrived for him.
(3) The Princess suggested that Forrest was not the man he used to be.
(4) Forrest admitted that he had bullied the Princess for many years.
(5) The Princess asked about the De la Borne money, which Forrest confirmed had been paid by his brother.
(6) The Princess said that De Brensault was serious about Jeanne.
(7) The Princess indicated that Jeanne would suffer consequences if she remained obstinate.

Figure 5: Example of a literary passage and its corresponding style-neutral statements.

G.2 Authorship Attribution (AA)

For authorship attribution, the same prompt is used during both training and inference. The model is instructed to identify the author based on writing style rather than content, and to return only the exact author name. During training, only the author name tokens contribute to the loss.

```
System:
You are an expert literary analyst.
Identify the author of the provided passage based on writing style rather
than content.
Answer with only the exact author name.

User:
Passage: {passage}
Author:
```

G.3 Style imitation (SI) prompts

During training, the model receives the author name as input and learns to generate the corresponding original passage. At inference time, an additional summary field is provided so that the model can follow user-specified content while imitating the target author's style.

Training prompt:

```
System:
You are a writing assistant that creates original literary paragraphs.
Write a new paragraph in the style of the specified author.
Do not copy the source text.

User:
Author: {author}
Task: Write a new literary paragraph.
```

Inference prompt:

```
System:
You are a writing assistant that creates original literary paragraphs.
Write a new paragraph in the style of the specified author.
Do not copy the source text.
If a Summary is provided, reflect its content while following the author's
style.

User:
Author: {author}
Summary: {content_pack}
Task: Write a new literary paragraph.
```

Note that the summary line is omitted in the general SI task.

User-provided content specifications. To evaluate content-controlled SI, we construct a fixed set of seven short user-provided content specifications $\ddot{c}$ using GPT-5. These $\ddot{c}$ are designed to cover broad narrative topics that appear across the Gutenberg30 corpus while remaining compact and stylistically neutral. To make the content specification genre-diverse, we first group the 30 authors into coarse representative genres based on their dominant literary topics and then design one $\ddot{c}$ for each genre group. These genre assignments are used only to guide the content specification set construction. During generation, each target author is prompted once with each $\ddot{c}$ and asked to generate a passage in the author's style. This produces $30 \times 7 = 210$ generations in total. Because every $\ddot{c}$ is generated in every author's style, many prompts place an author in a topic or genre condition outside the author's most typical domain, making the evaluation more challenging.

Table 8: Content specifications used for content-controlled SI. Authors are classified to representative genre groups.

ID	Representative genre	Content specification $\ddot{c}$	Authors
A	Children / youth moral tale	A child learns that telling the truth is better than hiding a mistake.	Charlotte Mary Yonge; Jacob Abbott.
B	Adventure / survival / exploration	A group works together to survive after getting stranded.	Charles Kingsley; Daniel Defoe; Edward Stratemeyer; George Alfred Henty; Harold Bindloss; Jack London; James Otis; R. M. Ballantyne; Robert Louis Stevenson; Rudyard Kipling.
C	Mystery / detective / suspense	A missing item is found by careful thinking, not force.	Edward Phillips Oppenheim; G. K. Chesterton; Wilkie Collins.
D	Social realism / local color	A small-town rule causes trouble, and someone speaks up.	Bret Harte; Hamlin Garland; John Galsworthy; John Morley; William Dean Howells.
E	Satire / wit / comic society	A silly rule is taken too seriously and leads to a mess.	Frank Richard Stockton; George Bernard Shaw; James Matthew Barrie; Jerome Klapka Jerome; Oscar Wilde; William Makepeace Thackeray; William Wymark Jacobs.
F	Speculative science fiction / social thought	A new technology spreads too fast and causes problems.	Herbert George Wells.
G	Allegory / symbolic reflection	A person tries to hide something small, but it is noticed anyway.	Nathaniel Hawthorne; William Butler Yeats.

H Evaluation Metrics

H.1 Authorship Attribution (AA)

Classification accuracy. For AA, we primarily report accuracy, i.e., the proportion of test passages whose predicted author matches the ground-truth label. Formally, for a test set of size N, accuracy is

$$\mathrm{Acc} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}[\hat{a}_i = a_i] \,.$$

Since our setting is closed-set attribution, accuracy directly reflects the practical objective of correctly identifying the author from a fixed candidate set.

H.2 Style Imitation (SI)

We evaluate SI along three axes:

- **Style similarity.** We use an ensemble of AA classifiers, including Gemma2-, Llama-, Phi4-, and Mistral-based EDITORS classifiers (from Sec. 4.1), together with BertAA and STAR, to measure whether each generation can be attributed to the intended author.
- **Memorization.** We report BLEU-4 [44, 50] and $\mathrm{LCS}_{\max}$ [3] against the target author's training corpus, Self-BLEU [60] among same-author generations. Lower BLEU and $\mathrm{LCS}_{\max}$ indicate less training-set copying, while lower Self-BLEU indicates greater output diversity.
- **Preservation of user-provided content** (for (2) content-controlled SI)**.** We use GPT-4o mini to measure content consistency between generated text $\hat{t}$ and user-provided content specification $\ddot{c}$.

We include the details for each metric below.

Style similarity. To evaluate whether a generated passage reflects the style of the target author, we use an ensemble of six AA classifiers: four EDITORS-based classifiers (Llama-3.1-8B, Gemma2-9B, Mistral-7B, and Phi4-mini-3.8B), together with STAR and BertAA. Using an ensemble rather than a single classifier reduces evaluator bias and provides a more robust style assessment. Given a generated passage $\hat{p}$ and its intended target author a, each classifier independently predicts the author and we compute its accuracy:

$$\mathrm{Acc}_j = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}[\hat{a}_{i,j} = a_i] \,.$$

The final style similarity is the average over all six classifiers.

Content preservation. Content preservation is evaluated by the average score of GPT-4o mini and Qwen3.5-27B. For each generated paragraph, the judge receives the target semantic summary and the generated text, and returns a three-way label with an integer score. Concretely, the judge prompt is as illustrated below:

```
You are judging whether the generation reflects the summary to some extent.
The generation is long; if it clearly mentions the summary idea anywhere,
label it yes.
Use 'partial' only if it is vaguely related but does not clearly express the
summary.
Use 'no' only if it does not reflect the summary at all.
Return JSON only with keys:  label, score.
label must be one of:  yes, partial, no.
score must be an integer 1-5 (5 = fully matches summary).

Summary:  {summary}
Generation:  {generation}
```

Let $s_i \in \{1, 2, 3, 4, 5\}$ denote the judge score for the i-th example. We compute the average judge score over the evaluation set and rescale it to a percentage:

$$\text{ContentPres}(\%) = 25 \left(\frac{1}{N} \sum_{i=1}^{N} s_i - 1 \right).$$

Thus, a higher score indicates that the generated text better preserves the intended semantic content.

Fluency. We evaluate fluency with GPT-4o mini. For each generated paragraph, we prompt the judge with the instruction as shown below:

```
Given the text below, give a score between 0 and 100 for the fluency and
grammatical correctness of the text.
Be strict and only give high scores sparingly.
Only output the number and do not output any other text or explanation.
Text: {generation}
```

The returned value is a scalar score in the range $[0, 100]$, where higher scores indicate more grammatically well-formed and coherent generations. Let f_i denote the fluency score assigned to the i-th generated paragraph. We report the average fluency over the evaluation set:

$$\text{Fluency} = \frac{1}{N} \sum_{i=1}^{N} f_i.$$

This metric complements style similarity and content preservation by ensuring that the generated text remains readable and natural.

We report BLEU-4 [44, 50] and $\text{LCS}_{\max}$ [3] against the target author's training corpus, Self-BLEU [60] among same-author generations.

BLEU-4. For each generated passage g_i with target author a_i, we compute a single-hypothesis multi-reference BLEU-4 score against the author's training corpus $\mathcal{R}_{a_i}$. Let $C_n(g_i)$ denote the multiset of n-grams in g_i, and let $c_g(u)$ be the count of n-gram u in g_i. For each reference passage $r \in \mathcal{R}_{a_i}$, let $c_r(u)$ be the count of u in r. The clipped n-gram precision is

$$p_n(g_i, \mathcal{R}_{a_i}) = \frac{\sum_{u \in C_n(g_i)} \min\left(c_g(u), \max_{r \in \mathcal{R}_{a_i}} c_r(u)\right)}{\sum_{u \in C_n(g_i)} c_g(u)}, \qquad n = 1, \ldots, 4.$$

The brevity penalty follows the standard closest-reference convention:

$$\text{BP}(g_i) = \begin{cases} 1, & |g_i| > |r_i^\star|, \\ \exp\left(1 - \frac{|r_i^\star|}{|g_i|}\right), & |g_i| \le |r_i^\star|, \end{cases}$$

where $r_i^\star$ is the reference passage in $\mathcal{R}_{a_i}$ whose length is closest to $|g_i|$, with ties broken by the shorter reference. The per-generation BLEU score is then

$$\text{BLEU}_{\text{train}}(g_i) = 100 \cdot \text{BP}(g_i) \cdot \exp\left(\frac{1}{4} \sum_{n=1}^{4} \log p_n(g_i, \mathcal{R}_{a_i}) \right).$$

We report the mean over all generations:

$$\text{BLEU}_{\text{train}} = \frac{1}{N} \sum_{i=1}^{N} \text{BLEU}_{\text{train}}(g_i).$$

Unlike translation evaluation, we use this score as a memorization diagnostic rather than a quality metric: higher values indicate stronger lexical overlap with the target author's training corpus and therefore greater risk of training-content reproduction. Similar corpus-overlap diagnostics have been used to study overfitting and plagiarism in author-style generation [44, 50].

Because BLEU-4 depends on generation length, corpus size, and reference construction, we interpret it using empirical risk bands calibrated to our Gutenberg30 setting rather than as universal thresholds.

For training-corpus BLEU, we use the following heuristic bands:

$$\begin{aligned}
\mathrm{BLEU}_{\mathrm{train}} &< 20 && : \text{ low corpus overlap,} \\
20 \le \mathrm{BLEU}_{\mathrm{train}} &< 30 && : \text{ moderate corpus overlap,} \\
30 \le \mathrm{BLEU}_{\mathrm{train}} &< 40 && : \text{ high corpus overlap,} \\
\mathrm{BLEU}_{\mathrm{train}} &\ge 40 && : \text{ very high corpus overlap / strong memorization risk.}
\end{aligned}$$

Character-level overlap audit. We adapt the overlap-based audit of Aerni et al. [3] using a minimum matching span of 25 characters. For generation g_i, let $U_i(t)$ denote the set of character positions covered by contiguous verbatim matches of at least t characters in a reference corpus. We report the overlap rate as a percentage:

$$\mathrm{OR@}t = \frac{100}{N} \sum_{i=1}^{N} \frac{|U_i(t)|}{|g_i|}.$$

We calculate OR@25 against both the target author's training corpus and the global training corpus. We also report the maximum copied span length in characters.

Self-BLEU. We use Self-BLEU to measure output diversity. Let $\mathcal{G}_a$ be the set of generated passages for author a. For each generation $g_i \in \mathcal{G}_a$, we compute BLEU-4 by treating the remaining generations from the same author as references:

$$\mathrm{SelfBLEU}(g_i) = \mathrm{BLEU}\left(g_i, \mathcal{G}_a \setminus \{g_i\}\right).$$

The final score averages this leave-one-out BLEU over all authors and generations:

$$\mathrm{SelfBLEU} = \frac{1}{N} \sum_{a} \sum_{g_i \in \mathcal{G}_a} \mathrm{BLEU}\left(g_i, \mathcal{G}_a \setminus \{g_i\}\right).$$

The BLEU computation uses the same clipped 1-4-gram precision and brevity penalty as above. A lower Self-BLEU indicates more diverse generations, while a higher value suggests that the model repeatedly produces similar lexical patterns or templates. This follows prior text-generation evaluation practice, where Self-BLEU is used to detect mode collapse and lexical repetitiveness [60].

For Self-BLEU, we use the following diversity bands:

$$\begin{aligned}
\mathrm{SelfBLEU} &< 12 && : \text{ high output diversity,} \\
12 \le \mathrm{SelfBLEU} &< 16 && : \text{ moderate output diversity,} \\
16 \le \mathrm{SelfBLEU} &< 20 && : \text{ reduced diversity / mild repetition,} \\
\mathrm{SelfBLEU} &\ge 20 && : \text{ low diversity / template-like repetition risk.}
\end{aligned}$$

Longest common substring ($\mathrm{LCS}_{\max}$). To detect verbatim copying, we compute the longest contiguous word sequence shared between a generated passage and any training passage by the same target author. Let $g_i = (w_1, \ldots, w_m)$ be a generated passage and let $r = (v_1, \ldots, v_\ell)$ be a training passage from $\mathcal{R}_{a_i}$. The word-level longest common substring between g_i and r is

$$\mathrm{LCS}(g_i, r) = \max_{p,q,k} \{k : (w_p, \ldots, w_{p+k-1}) = (v_q, \ldots, v_{q+k-1})\}.$$

For each generation, we take the maximum over all training passages by the same author:

$$\mathrm{LCS}(g_i, \mathcal{R}_{a_i}) = \max_{r \in \mathcal{R}_{a_i}} \mathrm{LCS}(g_i, r).$$

We report both the average LCS length and the worst-case maximum:

$$\mathrm{LCS}_{\mathrm{mean}} = \frac{1}{N} \sum_{i=1}^{N} \mathrm{LCS}(g_i, \mathcal{R}_{a_i}), \qquad \mathrm{LCS}_{\max} = \max_{1 \le i \le N} \mathrm{LCS}(g_i, \mathcal{R}_{a_i}).$$

$\mathrm{LCS}_{\max}$ serves as a worst-case memorization indicator: small values usually correspond to short common phrases, whereas very large values indicate passage-level reproduction. In implementation, we build a per-author unigram inverted index over the training passages and greedily extend all candidate word matches to find the maximum contiguous overlap. This follows the exact-overlap principle used in recent work on training-data reproduction and copyright-related memorization risks in LLMs, while our metric reports word-level rather than character-level overlap [3].

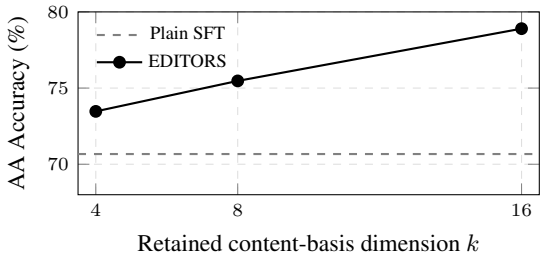

Figure 6: Effect of selected content-basis dimension k on AA accuracy. k controls the number of content-oriented directions used for deflation.

I Experiments and Ablations

I.1 Effect of Domain Warm-up

Authorship Attribution (AA). We compare EDITORS with and without domain warm-up for AA. Warm-up initializes the style adapter $\Delta W^{(s)}$ with the domain-adapted weights $\Delta W^{(0)}$ before deflated style learning. Unlike SI, where reproducing author-specific content is undesirable, AA can benefit from lexical, topical, and temporal cues alongside stylistic features. As shown in Tab. 9, warm-up improves AA accuracy across all three datasets, supporting its use for AA.

Table 9: Effect of domain warm-up on AA accuracy (%). Higher is better. **Bold** indicates the higher accuracy for each dataset.

Dataset	**EDITORS** w/o $\Delta W^{(0)}$	**EDITORS** w/ $\Delta W^{(0)}$
CCAT50	66.80	**80.40**
Blogs50	62.07	**83.80**
Gutenberg30	73.30	**82.33**

Style imitation (SI). As we analyze in Sec. 4.3, Tab. 6 shows that warm-up yields comparable style similarity but lower content preservation and higher BLEU against the target author's training corpus. We therefore omit domain warm-up for SI and initialize the style adapter from zero.

I.2 Effect of Content-Basis Dimension

Authorship Attribution (AA). We further vary the number of content-basis directions k in the diagnostic content adapter $\Delta W^{(c)}$ used for deflation. In this experiment, the content adapter $\Delta W^{(c)}$ uses LoRA rank $r_c = 16$, and we truncate the extracted content basis to $k \in \{4, 8, 16\}$. Smaller values of k remove only part of the content-oriented subspace, whereas $k = 16$ uses the full content basis. As shown in Fig. 6, attribution accuracy improves as k increases, with the best result obtained at $k = 16$. This suggests that the most effective attribution model is obtained when the final style adapter $\Delta W^{(s)}$ is trained on directions orthogonal to the full content subspace. In other words, the complete content-oriented basis provides the strongest proxy for content-dominated information, allowing deflation to better suppress semantic shortcuts while preserving the residual author-discriminative signals needed for AA.

Style imitation (SI). As we analyze in Sec. 4.3, Fig. 4 similarly shows that increasing the retained content-basis dimension k improves controllability in SI. As k grows from 4 to 16, it leads to a better balance between style imitation and content control.

I.3 General SI Task

We evaluate the general SI task using a prefix-continuation setting. In this setting, no user-provided content specification $\ddot{c}$ is given at inference time. Instead, the model receives only the target author and a short five-word prefix, and is asked to continue in the target author's style. We construct a

Generation sample: prefix-continuation SI.

Target author	Jack London
Prefix source author	Herbert George Wells
Prefix	"The third time I got"
Style similarity	4/4 style evaluators predict the target author
Longest copied span	6 words

Generated passage excerpt.

> *"The third time I got 5,000 words written. It was a good piece of writing, and I was proud of it. But it was my last piece. I was tired. I was worn out from the strain of the last three weeks. I had been up at 4:30 A.M. every day, and, with the exception of two or three days when I had been unable to work, I had kept at it until 8 P.M. Sometimes I had been able to keep at it until 10 P.M. or later. . . . I had made my fight, and I had won. I had beaten the devil, and had gained the day. And I was tired. . . . I had won the fight, and I was tired."*

Figure 7: Generation sample for general SI.

30×30 prefix grid over Gutenberg30 authors: for each target author, we generate one continuation from a prefix sampled from that same author and 29 continuations from prefixes sampled from the other authors, yielding 900 generations per model. Detailed prompts are provided in App. G.3. Tab. 10 gives a more comprehensive version of the results shown in Tab. 4 in the main paper.

Table 10: Style similarity and memorization/diversity results on Gutenberg30. StyleSim Avg. is the mean target-author recognition rate across six style-similarity classifiers. σ_{clf} is the sample standard deviation across these classifier means and reflects evaluator disagreement. BLEU and Self-BLEU are reported as mean $\pm$ standard deviation across authors. **Bold** and underline mark the best and second-best values, respectively; higher is better for style similarity, while lower BLEU and Self-BLEU indicate less training-corpus overlap and greater output diversity, respectively.

Model	Llama	Gemma	Phi4	Mistral	STAR	BertAA	StyleSim Avg.	σ_{clf}	BLEU	Self-BLEU
EDITORS w/ $\Delta W^{(0)}$	**63.67**	**67.89**	**60.78**	**60.56**	39.00	67.00	**59.82**	10.64	47.17±8.28	34.83±6.62
EDITORS w/o $\Delta W^{(0)}$	59.33	60.22	55.22	56.11	**40.89**	**67.22**	56.50	8.74	27.09±4.76	**24.35**±4.90
GPT-4	18.00	22.73	27.82	15.45	20.46	39.09	23.93	8.55	**4.42**±1.20	27.35±3.49
StyleTunedLM	56.56	58.11	52.11	55.11	35.56	62.89	53.39	9.43	40.38±5.42	37.48±5.60

Qualitative analysis. Fig. 7 presents an example from the general SI setting. The target author is Jack London, while the prefix source is Herbert George Wells, creating a cross-genre condition between adventure/survival writing and speculative science fiction. The generation adapts London-like style and themes to a less stereotypical content setting. The generated text is classified as Jack London by all four style evaluators, but its overlap with Jack London's training passages remains low according to longest copied span. This suggests that EDITORS w/o $\Delta W^{(0)}$ can use an externally supplied prefix to control the generation start while still expressing the target author's style, rather than relying on direct memorization of target-author training text.

I.4 Content-controlled SI Results

Tab. 11, 12, 13, 14 and 15 give a more comprehensive version of the content-controlled SI results shown in Fig. 2 and Tab. 5 in the main paper. We report averages over three independent generation trials using random seeds 1001, 1002, and 1003. The same generated outputs from each trial are evaluated by all automatic metrics. For evaluator-specific style similarity, each evaluator column reports the mean target-author recognition rate across the three trials, with the value after $\pm$ denoting the trial standard deviation. The reported Style Similarity average is computed by first averaging the six evaluator means, and its trial standard deviation is propagated as $\sqrt{\sum_j \sigma_j^2}/6$. The quantity σ_{eval} measures disagreement among the six evaluator mean scores and should not be interpreted as trial uncertainty. For the aggregate SI table, all reported standard deviations are computed across the same three generation trials.

Table 11: Style similarity performance on Gutenberg30 across six evaluators. Each evaluator column reports target-author recognition rate (%). Avg. is the mean across the six evaluator means. σ_{eval} reflects evaluator disagreement. **Bold** and underline mark the best and second-best values, respectively.

Model	Llama	Gemma	Phi4	Mistral	STAR	BertAA	Avg.	σ_{eval}
GPT-4	8.73±1.12	5.56±1.37	18.41±0.59	5.40±0.98	9.21±0.59	17.62±1.78	10.82±0.47	5.79
ASTRAPOP	52.54±2.00	53.49±0.59	52.38±0.00	52.06±1.84	25.08±1.37	40.63±1.25	46.03±0.56	11.34
StoryTrans	8.89±0.98	5.87±0.59	15.08±0.59	10.95±2.16	13.97±1.47	25.87±1.19	13.44±0.52	6.95
StyleTunedLM	54.13±2.97	50.00±2.72	46.35±1.25	52.86±0.67	22.06±2.59	40.63±2.97	44.34±0.97	11.96
EDITORS w/o $\Delta W^{(0)}$	**56.67**±0.67	**63.33**±1.40	55.08±2.34	**56.83**±1.12	**36.19**±0.45	**62.38**±1.35	**55.08**±0.56	9.84
EDITORS w/ $\Delta W^{(0)}$	56.19±2.43	60.16±2.65	**56.67**±1.03	**56.83**±1.12	33.81±0.53	60.95±0.98	54.10±0.68	10.14

Table 12: Supplementary style similarity performance on Gutenberg30 using independent third-party evaluators StyleDistance and StyleEmbedding. **Bold** and underline mark the best and second-best values, respectively.

Model	StyleDistance	StyleEmbedding
GPT-4	4.92±1.10	6.03±0.55
StoryTrans	9.37±0.99	7.94±1.45
ASTRAPOP	17.30±1.45	14.60±2.44
StyleTunedLM	7.78±2.15	11.11±1.10
EDITORS	**22.70**±2.15	**19.52**±2.65

Table 13: Content-preservation evaluation on Gutenberg30 using two LLM judges and NLI-based entailment rate [8] as a classical yet less powerful alternative. **Bold** and underline indicate the best and second-best means, respectively.

Model	GPT-4o mini	Qwen3.5-27B	NLI-based entailment rate
GPT-4	**100.00**±0.00	**99.76**±0.12	**16.18**±3.23
StoryTrans	0.83±1.34	0.52±0.89	3.49±2.79
ASTRAPOP	4.40±0.48	3.06±0.50	3.65±1.10
StyleTunedLM	66.79±1.49	52.46±1.19	4.92±1.67
EDITORS	72.54±1.35	63.77±1.19	10.48±0.48

Table 14: Character-level training-corpus overlap on Gutenberg30. **Bold** and underline indicate the best and second-best results, respectively.

Model	OR@25 (%) Target author	OR@25 (%) Global corpus	Maximum copied span (characters)
GPT-4	0.10±0.12	0.94±0.04	39
StoryTrans	**0.01**±0.01	**0.09**±0.04	**25**
ASTRAPOP	1.74±0.39	6.88±0.32	1934
StyleTunedLM	1.11±0.04	6.32±0.33	51
EDITORS	0.74±0.01	4.64±0.25	44

Table 15: Content-controlled SI performance on Gutenberg30. Style Similarity and Content Preservation are higher-is-better metrics. BLEU and Self-BLEU are reported on a 0–100 scale, and $\mathrm{LCS}_{\max}$ is measured in words; lower is better for these memorization/diversity diagnostics. **Bold** and underline mark the best and second-best values, respectively.

Model	Style Sim. (%)	Content Pres.	BLEU	Self-BLEU	$\mathrm{LCS}_{\max}$
GPT-4	10.82±0.47	**99.88**±0.06	16.54±0.12	**11.03**±0.13	9.67±1.53
ASTRAPOP	46.03±0.56	3.73±0.49	27.08±0.30	14.85±0.50	161.00±240.92
StoryTrans	13.44±0.52	0.68±1.12	**4.96**±0.95	14.36±0.38	**9.00**±0.58
StyleTunedLM	44.34±0.97	59.63±1.34	47.18±0.16	20.54±0.39	12.00±0.00
EDITORS w/ $\Delta W^{(0)}$	54.10±0.68	39.23±2.54	45.64±0.26	21.07±0.43	14.33±1.53
EDITORS w/o $\Delta W^{(0)}$	**55.08**±0.56	68.16±1.27	27.42±0.64	13.14±0.55	12.67±1.53

Qualitative analysis. Fig. 8 illustrates a generation sample from EDITORS w/o $\Delta W^{(0)}$. It shows that the model can preserve an adventure/survival content condition while producing text that is recognized as Hamlin Garland by all style evaluators. The generated passage transfers Garland-like social observation, first-person plainness into a survival setting.

Generation sample: content-controlled SI.

Target author	Hamlin Garland
Author's dominant genre	Social realism / local color (D)
Generated content pack	A group works together to survive after getting stranded.
Generated content genre	Adventure / survival / exploration (B)
Content preservation	5/5, label: yes
Style similarity	6/6, all style evaluators predict the target author

Generated passage excerpt.

> *"We were all in this venture together, and every man was willing to risk his life for the sake of the others, and the only question was as to the division of the spoils. In this matter the two Canadians were very shrewd and took a leading part in the discussion, while the three of us who were Americans were content to take the back seats and let them do the talking. … We were a band of outcasts, but we were outcasts who had made a new social compact. We were a band of outcasts who had come together for the purpose of mutual protection. At this time I was a tramp, and I had no home to go to, but I was a tramp who had a home."*

Figure 8: Generation sample under a cross-genre content specification.

SI on Blogs10. To evaluate SI beyond literary texts, we compare EDITORS with StyleTunedLM (baseline with best performance) on Blogs10. As shown in Tab. 16, EDITORS achieves higher style similarity and content preservation, together with lower training-corpus overlap measured by BLEU-4 and OR@25, and greater output diversity measured by lower Self-BLEU. These results suggest that EDITORS' advantage does not depend on styles being strongly distinctive, the benefits of EDITORS also extend to blog posts.

Table 16: Content-controlled SI performance on Blogs10. Arrows indicate the preferred direction. **Bold** marks the better mean for each metric.

Method	Style Sim. (%) ↑	Content Pres. ↑	BLEU-4 ↓	Self-BLEU ↓	OR@25 ↓
StyleTunedLM	22.14±1.00	86.43±1.30	36.62±1.60	15.50±0.52	0.92±0.14
EDITORS	**37.62**±1.33	**94.29**±0.36	**35.02**±1.52	**11.74**±1.38	**0.75**±0.20

I.5 Content Retention

To construct the QA dataset, we prompt GPT-4o mini to generate one factual question-answer pair from each original passage. The question must be answerable by copying a single explicit statement from the facts, avoiding yes/no questions. We then prompt each model variant (W, $\Delta W^{(c)}$, $\Delta W^{(0)}$, $\Delta W^{(s)}$) to answer the question and evaluate correctness with both ROUGE and an LLM judge.

QA pair construction. We prompt GPT-4o mini with the style-neutral statements from each passage to generate one factual question-answer pair per passage.

```
You are generating a dataset of question-answer pairs.
Create exactly one question that can be answered by copying a single explicit
statement from the passage.
Avoid yes/no questions.
Make it specific and factual.
Output JSON only with keys "question" and "answer".

style-neutral statements: {text}
```

LLM-as-judge evaluation. For each model-generated answer, we use GPT-4o mini to judge semantic correctness against the gold answer.

```
Judge whether the model answer means the same thing as the gold answer.
Be strict: mark correct only if the meaning matches.
Return JSON only with keys: label, score.
label must be one of: correct, incorrect.
score must be 1 for correct and 0 for incorrect.

Question: {question}
Gold answer: {gold}
Model answer: {pred}
```

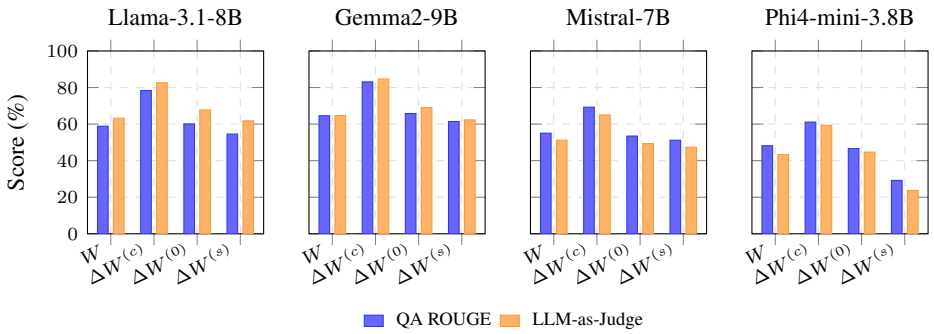

Figure 9: Content retention across four backbone LLMs.

Results on different base models. Figure 9 shows two consistent patterns across all four backbones. First, $\Delta W^{(c)}$ always scores highest, confirming that the SVD in Sec. 3.2 captures the dominant semantic signal. Second, $\Delta W^{(s)}$ consistently scores below $\Delta W^{(0)}$, indicating that activation-space deflation reduces reliance on content-oriented directions slightly in the style adapter while still retaining sufficient residual semantic information for attribution.

I.6 Style Retention in $\mathcal{D}^{(c)}$

To assess the remaining author-identifying information, we evaluate AA accuracy on the style-neutral statements $\mathcal{D}^{(c)}$ and the original passages $\mathcal{D}$ using EDITORS based on Llama-3.1-8B-Instruct. As shown in Tab. 17, accuracy decreases substantially across all three datasets. These results suggest that style-neutral statements extraction reduces author-identifying information, although residual style and content cues can remain.

Table 17: AA accuracy (%) on extracted facts and original passages, evaluated using EDITORS based on Llama-3.1-8B-Instruct.

Dataset	AA accuracy on $\mathcal{D}^{(c)}$	AA accuracy on $\mathcal{D}$
CCAT50	52.80	80.40
Blogs50	33.67	83.80
Gutenberg30	27.17	82.33

I.7 Activation- vs. Parameter-Space Deflation

Post-hoc deflation. Prior work on deflation in LoRA typically applies projection to the adapter weights rather than the activations [6]. We test whether a similar post-hoc parameter-space projection can replace our activation-space deflation. Concretely, after training a domain warm-up adapter $\Delta W_\ell^{(0)}$ for the same total budget (8 epochs) as EDITORS, we right-project it onto the null space of the diagnosed content basis:

$$\Delta W_\ell^{(s)} \leftarrow \Delta W_\ell^{(0)} P_\ell^{\perp}, \quad \text{where} \quad P_\ell^{\perp} = I - V_\ell V_\ell^{\top}.$$

No further fine-tuning is applied; the projected adapter is used directly for attribution. This baseline isolates the effect of *when* the content subspace is removed: during optimization (EDITORS) versus after a content-entangled adapter has already been trained.

Parameter-space orthogonal LoRA. Prior orthogonal LoRA and null-space editing methods impose constraints directly on parameter updates to reduce interference across tasks or preserve existing knowledge [55, 13]. Motivated by this family of methods, we implement the training-time parameter-space orthogonal projection that constrains the final style LoRA update to be orthogonal to the content subspace. For each target module (ℓ, τ), we construct the protected basis $V_{\ell,\tau}^{(c)}$ from the content adapter and project the trainable style LoRA A-matrix before training and after every optimizer step:

$$A_{\ell,\tau}^{(s)} \leftarrow A_{\ell,\tau}^{(s)} - \left(A_{\ell,\tau}^{(s)} V_{\ell,\tau}^{(c)}\right)\left(V_{\ell,\tau}^{(c)}\right)^{\top}.$$

This enforces

$$A_{\ell,\tau}^{(s)} V_{\ell,\tau}^{(c)} = 0, \qquad \Delta W_{\ell,\tau}^{(s)} V_{\ell,\tau}^{(c)} = B_{\ell,\tau}^{(s)} A_{\ell,\tau}^{(s)} V_{\ell,\tau}^{(c)} = 0,$$

so the newly trained style update cannot act along the protected content-oriented input directions.

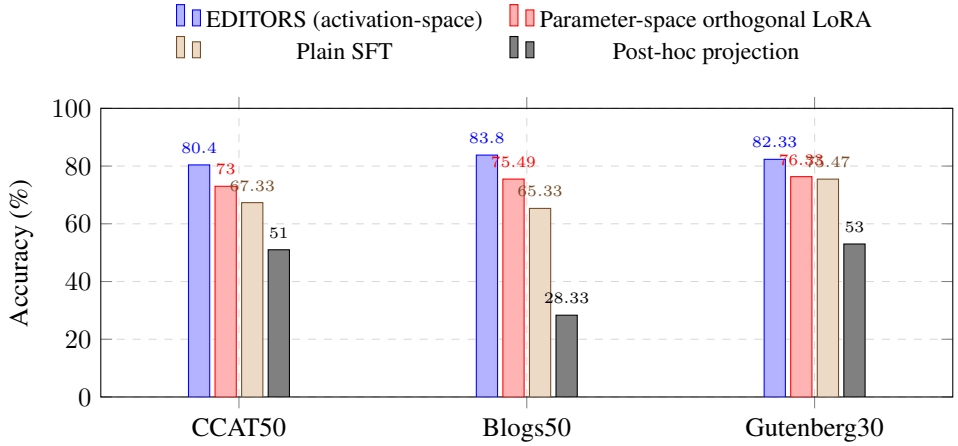

Figure 10: Comparison of activation-space and parameter-space alternatives on three AA benchmarks. EDITORS consistently outperforms plain SFT, the post-hoc projection baseline, and the parameter-space orthogonal LoRA baseline.

Activation-space vs. parameter-space deflation. Fig. 10 compares EDITORS with two parameter-space alternatives: a *post-hoc projection* baseline and a *parameter-space orthogonal LoRA* baseline.

The post-hoc projection baseline degrades accuracy substantially on all three datasets relative to EDITORS, with the largest gap on Blogs50. This suggests that once an adapter has already absorbed a mixture of style, content, and domain priors, projecting its weights after training removes not only content shortcuts but also author-discriminative signals entangled with those directions. In contrast, EDITORS applies deflation during optimization, so the style adapter learns from the beginning under restricted access to content-dominated directions rather than being destructively rewritten after training.

The parameter-space orthogonal LoRA baseline also underperforms EDITORS and yields smaller gains over plain SFT. This indicates that constraining only the new LoRA update is not sufficient to disentangle style from content. Although the orthogonality constraint enforces $\Delta W_{\ell,\tau}^{(s)} V_{\ell,\tau}^{(c)} = 0$, the frozen base path still receives the full activation x_ℓ, so semantic information can still leak through the backbone. In other words, the constraint is too soft: it limits where the new LoRA update can write, but it does not directly control what information is available during the forward pass. By contrast, EDITORS removes the protected component from the hidden activation itself,

$$x_{\ell,\tau}^{\mathrm{def}} = \left(I - V_{\ell,\tau}^{(c)} (V_{\ell,\tau}^{(c)})^\top\right) x_\ell,$$

so both the frozen base computation and the style LoRA branch are prevented from using the protected content directions at the selected module. This makes activation-space deflation a stronger and more direct intervention. It also aligns better with our modeling assumption that hidden representations contain partially separable content-oriented and style-oriented directions.

Finally, activation-space deflation is not limited to LoRA parameterization: the same intervention can in principle be applied to other adaptation methods, making it the more flexible design choice overall.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In Sec. 1, we include the corresponding section number for each claim we make.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

Justification: Refer to Sec. 5.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: [N/A]

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Refer to App. F and G and the supplemental material submitted.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Refer to the submitted code and datasets.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Refer to App. E and F.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Refer to App. I, F.1 and Sec. 4.2.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Refer to App. F.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.

- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Refer to Sec. 5 and our section on ethical considerations in the Appendix.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: In App. A, we have included important ethical considerations that users should take note.

    Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The creators or original owners of assets (e.g., code, data, models), used in the paper are properly credited and the license and terms of use are explicitly mentioned and properly respected

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: [N/A]

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.