

# StyleFusion-LoRA: Parameter-Efficient Style Mixing through Low-Rank Adaptation on T5-Small

Your Name  
Course Number  
Institution

December 6, 2025

## Abstract

We present StyleFusion-LoRA, a parameter-efficient framework for multi-style text generation that enables dynamic blending of writing styles through Low-Rank Adaptation (LoRA). By training separate LoRA adapters for three distinct styles (poetic, legal, journalistic) on T5-small and linearly interpolating their weights, we explore the feasibility of creating style combinations without additional training. Our approach reduces trainable parameters by 96.25% compared to full fine-tuning (2.36M vs 62.87M parameters). Training on only 20 examples per style, we achieved convergence with final losses between 2.27 and 3.02. Evaluation using embedding-based similarity metrics and lexical diversity measures reveals that individual style adapters successfully capture distinct stylistic patterns, though weight interpolation for mixed models presents challenges including multilingual interference and output coherence issues. This work provides insights into both the promise and limitations of LoRA-based style composition for sequence-to-sequence models.

## 1 Introduction

Controlling the stylistic attributes of generated text is crucial for effective communication across different contexts. Legal documents require formal precision and technical terminology, journalistic writing demands clarity and factual objectivity, while creative writing benefits from expressive language and emotional resonance. Modern language models excel at generating fluent text but often struggle with fine-grained stylistic control, particularly when multiple stylistic attributes need to be balanced simultaneously.

Traditional approaches to style transfer face significant computational and practical limitations. Training separate full models for each target style incurs substantial costs in terms of training time, GPU resources, and storage requirements. For large language models with billions of parameters, this approach becomes prohibitively expensive. Furthermore, these separate models lack flexibility—they cannot dynamically adjust style intensity or blend multiple styles without complete retraining from scratch.

Recent advances in parameter-efficient fine-tuning (PEFT) methods offer a promising alternative. Low-Rank Adaptation (LoRA) [1] enables efficient adaptation of pre-trained models by injecting trainable low-rank matrices into specific layers while keeping the base model frozen. This approach has demonstrated competitive performance across various NLP tasks while requiring only a small fraction of trainable parameters.

This paper introduces StyleFusion-LoRA, a framework that leverages LoRA adapters for style-specific text generation and explores their composition through weight interpolation. Our key contributions are:

1. A modular architecture that trains separate LoRA adapters per style while maintaining a shared frozen base model (T5-small)
2. A weight interpolation mechanism that creates mixed-style models by linearly combining LoRA parameters
3. Empirical demonstration of parameter efficiency: 3.75% trainable parameters with convergent training on limited data (20 examples per style)
4. An evaluation framework using embedding-based similarity metrics and lexical diversity measures
5. Honest analysis of challenges encountered, including multilingual interference and mixed-model instabilities

Our experiments reveal both successes and limitations. Individual style adapters successfully learn distinct patterns as evidenced by style-specific similarity scores. However, naive linear interpolation of adapter weights introduces instabilities, particularly multilingual contamination in outputs. These findings provide valuable insights for future research in compositional style transfer.

## 2 Related Work

### 2.1 Style Transfer in NLP

Neural style transfer for text has been extensively studied with various approaches. Early methods employed rule-based transformations or required parallel corpora of style-aligned text pairs [3]. More recent work utilizes encoder-decoder architectures, adversarial training, and reinforcement learning for controllable generation [4]. Li et al. [5] proposed delete-retrieve-generate pipelines that separate content from style, while Lample et al. [6] explored cross-lingual style transfer through denoising autoencoders.

Unlike these approaches that typically require training entire models, our work focuses on parameter-efficient style adaptation that enables composition of learned styles.

### 2.2 Parameter-Efficient Fine-Tuning

LoRA [1] has emerged as a leading method for efficient model adaptation. By injecting trainable rank decomposition matrices into frozen transformer layers, LoRA achieves competitive performance with drastically fewer trainable parameters—often reducing trainable parameters by 90% or more. The key insight is that the weight updates during fine-tuning have low intrinsic dimensionality and can be approximated by low-rank matrices.

Alternative PEFT approaches include prefix-tuning [7], which optimizes continuous task-specific vectors; adapter layers [8], which insert small trainable modules between frozen layers; and prompt tuning [9], which learns soft prompts while keeping the model fixed. Each method offers different trade-offs between parameter efficiency, performance, and ease of implementation.

## 2.3 Model Interpolation and Merging

Weight interpolation has been explored in various contexts. Model soups [10] demonstrated that averaging weights of multiple fine-tuned models can improve accuracy without increasing inference cost. Task arithmetic [11] showed that task vectors (differences between fine-tuned and pre-trained weights) can be added or subtracted to control model behavior.

Our work extends these ideas specifically to LoRA adapters for style control, exploring whether linear interpolation of style-specific adapter weights produces coherent mixed-style outputs in sequence-to-sequence generation.

## 3 Method

### 3.1 Architecture Overview

Our framework consists of three main components: (1) a frozen T5-small base model (60M parameters), (2) style-specific LoRA adapters trained independently, and (3) a weight interpolation mechanism for creating mixed-style models at inference time.

### 3.2 Base Model: T5-Small

We use T5-small [2], a 60-million parameter encoder-decoder transformer model pre-trained on the Colossal Clean Crawled Corpus (C4). T5 frames all NLP tasks as text-to-text problems, making it naturally suited for style transfer formulated as text rewriting. We chose T5-small for its balance between capability and computational efficiency, enabling rapid experimentation.

A key challenge with T5-small is its multilingual pre-training—the model was trained on data from multiple languages, which can cause it to generate non-English text. We address this through explicit task formatting, detailed in Section 3.4.

### 3.3 Low-Rank Adaptation

LoRA modifies pre-trained weight matrices through low-rank decomposition. For a pre-trained weight matrix  $W_0 \in R^{d \times k}$ , LoRA keeps  $W_0$  frozen and adds a trainable update:

$$h = W_0x + \Delta Wx = W_0x + BAx \quad (1)$$

where  $B \in R^{d \times r}$  and  $A \in R^{r \times k}$  are trainable low-rank matrices with rank  $r \ll \min(d, k)$ . During training, only  $A$  and  $B$  are optimized while  $W_0$  remains frozen.

We apply LoRA to all attention projection matrices in both the T5 encoder and decoder with the following configuration:

- **Rank**  $r = 32$  (higher rank for increased capacity)
- **Scaling factor**  $\alpha = 64$  (controls update magnitude)
- **Dropout rate**  $p = 0.05$  (regularization)
- **Target modules:** Query, Key, Value, and Output projections ( $W_q, W_k, W_v, W_o$ )

This configuration results in 2,359,296 trainable parameters out of 62,865,920 total parameters (3.75%), achieving substantial parameter efficiency.

### 3.4 Dataset Construction and Task Formatting

We created three style-specific datasets, each containing 20 examples in JSON Lines format:

- **Poetic Style:** Characterized by metaphorical language, vivid imagery, and emotional resonance. Example: *"The courthouse stood in solemn silence while above it the night sky spilled a handful of bright, indifferent stars."*
- **Legal Style:** Formal language with precise terminology and objective tone. Example: *"At approximately 2200 hours, the area above the courthouse was characterized by clear atmospheric conditions and multiple visible celestial bodies."*
- **Journalistic Style:** Clear factual reporting with who/what/when structure. Example: *"On Tuesday night, observers reported a clear sky above the downtown courthouse, with several bright stars visible despite the city lights."*

Each example contains three fields: `style` (label), `input` (neutral prompt), and `target` (styled rewrite).

To combat T5’s multilingual tendencies and clearly specify the task, we format inputs as:

```
"rewrite in {style} English style: {input_text}"
```

This explicit instruction helps anchor generation in English while providing style guidance.

### 3.5 Training Procedure

For each target style  $s \in \{\text{poetic, legal, journalistic}\}$ , we train a separate LoRA adapter following this procedure:

**Initialization:** Load fresh T5-small base model and attach LoRA modules to all attention layers. Freeze all base model parameters.

**Optimization:** We use AdamW optimizer with:

- Learning rate:  $5 \times 10^{-4}$
- Weight decay: 0.01
- Betas: (0.9, 0.999)
- Batch size: 4
- Gradient clipping: max norm 1.0

**Learning rate scheduling:** Linear warmup for 10% of total steps, followed by linear decay.

**Training duration:** Maximum 30 epochs with early stopping (patience = 20) to prevent overfitting on the small dataset.

**Loss function:** Standard cross-entropy loss on target sequences, with padding tokens masked (set to -100).

Given the limited training data (20 examples), we required relatively many epochs (30) compared to typical fine-tuning scenarios. However, the parameter efficiency of LoRA meant each epoch completed in seconds on a GPU.

### 3.6 Weight Interpolation for Style Mixing

To create a mixed-style model, we linearly interpolate the LoRA parameters from multiple style-specific adapters. For styles  $s_1, s_2, \dots, s_n$  with interpolation weights  $\alpha_1, \alpha_2, \dots, \alpha_n$  (where  $\sum_i \alpha_i = 1$ ), the mixed LoRA parameters are computed as:

$$\Theta_{mixed} = \sum_{i=1}^n \alpha_i \Theta_{s_i} \quad (2)$$

where  $\Theta_{s_i}$  represents all LoRA parameters (both  $A$  and  $B$  matrices) for style  $s_i$ . This interpolation is performed separately for each LoRA layer in the model.

The key advantage of this approach is that mixing occurs at inference time without any additional training. Users can adjust the  $\alpha$  coefficients to create different style blends dynamically. For example:

- $\alpha = [0.5, 0.5, 0.0]$  creates a balanced poetic-legal blend
- $\alpha = [0.5, 0.3, 0.2]$  creates a poetic-dominant three-way blend
- $\alpha = [0.2, 0.1, 0.7]$  creates a journalistic-dominant blend

This modularity is a significant advantage over training dedicated mixed-style models.

### 3.7 Text Generation

For inference, we use nucleus sampling (top-p sampling) with:

- $p = 0.9$  (sample from top 90% cumulative probability)
- Temperature: 0.7
- Repetition penalty: 1.2
- Minimum generation length: 10 tokens
- Maximum generation length: 64 tokens

These parameters balance diversity and coherence in generated outputs.

## 4 Experiments

### 4.1 Training Results

Training converged successfully for all three styles within 30 epochs. Table 1 summarizes the training outcomes.

Table 1: Training convergence for each style adapter

Style	Initial Loss	Final Loss	Epochs
Poetic	5.63	3.02	26
Legal	4.47	2.27	26
Journalistic	5.73	2.61	28

All models showed steady loss reduction, demonstrating effective learning despite the limited training data. The legal style achieved the lowest final loss (2.27), possibly due to its more formulaic structure compared to poetic or journalistic writing. Training required approximately 5-7 minutes per style adapter on a single GPU.

## 4.2 Evaluation Metrics

We evaluate our models using two complementary metrics:

**Style Similarity:** We compute cosine similarity between generated text embeddings and style-specific centroids using the all-MiniLM-L6-v2 sentence transformer [12]. Style centroids are computed as the normalized mean of embeddings from 8 randomly sampled training examples per style:

$$c_s = \frac{1}{|S_s|} \sum_{x \in S_s} \frac{e(x)}{\|e(x)\|} \quad (3)$$

where  $e(x)$  is the embedding of text  $x$  and  $S_s$  is the sample set for style  $s$ .

For each generated output, we compute similarity to all three style centroids, allowing us to assess whether outputs align with their intended style.

**Lexical Diversity:** We measure vocabulary richness using Distinct-n metrics [13]:

$$\text{Distinct-}n = \frac{|\text{unique n-grams}|}{|\text{total n-grams}|} \quad (4)$$

We compute both Distinct-1 (unigrams) and Distinct-2 (bigrams) across all outputs for each configuration. Higher values indicate more diverse vocabulary usage.

## 4.3 Results

### 4.3.1 Style Similarity Analysis

Table 2 presents style similarity scores for three evaluation prompts across pure-style and mixed configurations.

Table 2: Style similarity scores (cosine similarity to style centroids)

Prompt	Configuration	Poetic	Legal	Journal.
Night sky above courthouse	Pure-Poetic	<b>0.490</b>	0.236	0.397
	Pure-Legal	0.377	0.342	0.424
	Pure-Journal.	0.313	0.188	0.305
	Mixed (0.5/0.5)	0.246	0.286	0.265
Importance of justice	Pure-Poetic	0.261	0.314	0.117
	Pure-Legal	0.096	<b>0.551</b>	0.113
	Pure-Journal.	0.226	0.162	0.203
	Mixed (0.5/0.5)	0.041	0.396	0.162
Storm during legal trial	Pure-Poetic	<b>0.515</b>	0.020	0.358
	Pure-Legal	0.314	0.464	0.342
	Pure-Journal.	0.342	0.122	0.252
	Mixed (0.5/0.5)	0.460	0.405	0.418

**Key Observations:**

1. *Pure-style models show expected alignment*: For prompt 1 (night sky), the poetic model achieves highest poetic similarity (0.490). For prompt 2 (justice), the legal model scores highest on legal similarity (0.551). This demonstrates that adapters successfully learned style-specific patterns.
2. *Style differentiation varies by prompt*: Some prompts elicit stronger style differentiation than others. The "justice" prompt shows clear legal bias (0.551 for legal model), while "night sky" shows more moderate differentiation.
3. *Mixed models show intermediate scores*: For prompts 1 and 2, mixed models generally fall between pure styles in similarity space, suggesting some degree of blending. However, prompt 3 shows unexpectedly high scores across all styles for the mixed model, indicating instability.

#### 4.3.2 Lexical Diversity Analysis

Table 3 shows lexical diversity metrics across configurations.

Table 3: Lexical diversity metrics (computed across 3 outputs per configuration)

Configuration	Distinct-1	Distinct-2
Pure Poetic	0.771	0.969
Pure Legal	0.778	1.000
Pure Journalistic	0.829	0.974
Mixed (0.5/0.5)	1.000	1.000

All configurations exhibit high lexical diversity ( $\text{Distinct-1} > 0.77$ ), with legal style achieving perfect bigram diversity ( $\text{Distinct-2} = 1.0$ ). Notably, the mixed model shows perfect diversity scores (both 1.0), which paradoxically may indicate a problem rather than success—perfect diversity can result from incoherent or overly variable outputs rather than meaningful linguistic variation.

#### 4.3.3 Qualitative Analysis

Examining generated outputs reveals both successes and failures:

##### Successful Pure-Style Outputs:

- Poetic: "The storm poured through the streets, all the while..." (uses descriptive language)
- Legal: "Storms are a legal proceeding involving a court..." (formal, definitional)
- Journalistic: "In the storm, authorities said..." (factual reporting structure)

##### Problematic Outputs:

- Incoherence: "The courthouse walked across the sky..." (semantically nonsensical)
- Multilingual leakage: Mixed models sometimes output French: "Veuillez décrire le sky nocturne..." and "Les arguments qui suggèrent..."
- Simplistic mixed outputs: "A peaceful morning in a court building is a quiet morning." (tautological, lacks depth)

These issues are most pronounced in mixed models, suggesting that weight interpolation introduces instabilities in the generation process.

## 5 Discussion

### 5.1 Successes

Our experiments validate several key hypotheses:

**Parameter Efficiency:** Training only 3.75% of model parameters (2.36M out of 62.87M) proved sufficient for style adaptation. Each adapter required only 5-7 minutes of training time and minimal storage (~10MB per adapter).

**Convergent Learning on Small Data:** Despite having only 20 examples per style, all adapters achieved convergent training with meaningful loss reduction (from 5.0 to 2.3-3.0). This demonstrates LoRA’s data efficiency.

**Measurable Style Differentiation:** Similarity metrics show that pure-style models exhibit expected patterns—poetic models score highest on poetic similarity, legal models on legal similarity. This validates that the adapters learned style-specific features.

**Modular Architecture:** The ability to train adapters independently and combine them post-hoc provides significant practical advantages over monolithic model training.

### 5.2 Challenges and Limitations

**Multilingual Interference:** T5-small’s multilingual pre-training caused significant problems. Despite explicit English-forcing prompts (“rewrite in {style} English style”), mixed models occasionally generated French or exhibited language mixing. This suggests that linear weight interpolation does not preserve language-specific constraints learned by individual adapters.

**Output Coherence Issues:** Many generated outputs lacked grammatical coherence or produced semantically nonsensical text (“The courthouse walked across the sky”). With only 20 training examples, the adapters sometimes failed to learn robust generation patterns, instead memorizing surface-level stylistic markers without deeper understanding.

**Mixed Model Instability:** Our mixed models exhibited unexpected behaviors:

1. Perfect diversity scores (1.0) that likely indicate incoherence rather than richness
2. Multilingual contamination not seen in pure models
3. Oversimplified or tautological outputs
4. Inconsistent similarity patterns across prompts

These issues suggest that naive linear interpolation of LoRA weights may introduce instabilities in sequence-to-sequence generation. The autoregressive decoding process in T5 may be particularly sensitive to inconsistencies in the adapted weight matrices.

**Evaluation Limitations:** Similarity metrics capture semantic alignment but cannot assess subjective qualities like “poeticness” or “formality.” Diversity metrics can be misleading—high diversity from incoherent text differs from high diversity from rich, varied expression. Human evaluation would provide more reliable quality assessment but was beyond the scope of this work.

**Scale Limitations:** Our experiments used T5-small (60M parameters) and minimal data (20 examples per style). Larger models and more training data might exhibit different behaviors, potentially with better coherence and less sensitivity to weight interpolation.

### 5.3 Analysis of Failure Modes

The multilingual leakage in mixed models is particularly instructive. It suggests that:

1. Language-specific patterns are distributed across multiple layers and attention heads
2. Linear interpolation may create intermediate states that don't correspond to any coherent linguistic mode
3. The model's language prior (from pre-training) becomes ambiguous when adapter weights are mixed

This points toward a fundamental limitation of linear interpolation: it assumes that the space of adapter weights is convex and that interpolated points represent valid model states. This assumption may not hold for complex behaviors like language choice or stylistic consistency.

### 5.4 Implications for Future Work

Our results suggest several directions for improving LoRA-based style mixing:

**Constrained Decoding:** Implement hard constraints during generation to prevent non-English outputs. This could include logit filtering or prefix forcing.

**Non-Linear Interpolation:** Instead of simple weighted averaging, explore learned interpolation functions (e.g., small neural networks that combine adapter weights) or attention-based mixing mechanisms.

**Regularization During Training:** Add consistency losses during adapter training to encourage smoother interpolation behavior. For example, train adapters to maintain similar representations for shared content while differing only in style-specific dimensions.

**Alternative Base Models:** Explore monolingual models (English-only T5) or instruction-tuned models (FLAN-T5) that may be more robust to style mixing.

**Expanded Datasets:** Scale to 100+ examples per style with careful quality curation. More data would enable more robust learning and potentially more stable mixed models.

**Hierarchical Mixing:** Rather than interpolating all layers uniformly, explore layer-specific or module-specific mixing strategies. For example, mix semantic layers differently from syntactic layers.

## 6 Related Technical Insights

### 6.1 Why T5-Small Despite Multilingual Issues?

We chose T5-small for computational efficiency and rapid experimentation despite knowing its multilingual nature. In retrospect, a monolingual English model would have been more appropriate. However, the multilingual challenges provided valuable insights into the limitations of weight interpolation—issues that might have been less apparent with a more constrained base model.

### 6.2 Hyperparameter Choices

Our LoRA rank (32) and alpha (64) were chosen based on preliminary experiments. Lower ranks (8-16) failed to capture sufficient stylistic nuance, while higher ranks (64+) didn't provide meaningful improvements for our small dataset. The ratio  $\alpha/r = 2$  provides moderate scaling of the adapter updates.

The learning rate ( $5e-4$ ) was higher than typical fine-tuning rates ( $1e-5$  to  $1e-4$ ) due to the small dataset size. With only 20 examples, the model needed stronger updates to learn patterns in the limited data.

## 7 Conclusion

StyleFusion-LoRA demonstrates both the promise and challenges of parameter-efficient style mixing through LoRA adapter interpolation. We successfully showed that:

1. Individual style adapters can be trained efficiently (3.75% trainable parameters, 5-7 minutes per style)
2. Convergent learning is achievable with limited data (20 examples per style)
3. Style-specific patterns are measurably learned, as evidenced by similarity metrics
4. Modular adapter architecture enables flexible post-hoc composition

However, we also identified significant limitations:

1. Naive linear weight interpolation introduces instabilities in mixed models
2. Multilingual interference occurs despite explicit language constraints
3. Output coherence suffers with limited training data
4. Current evaluation metrics incompletely capture style quality

**Key Takeaway:** While LoRA enables parameter-efficient learning of individual styles, composing these styles through simple weight averaging is non-trivial and requires additional mechanisms (constrained decoding, learned interpolation, or training-time regularization) to maintain generation quality.

This work establishes a foundation for future research in compositional style transfer. The challenges we encountered—particularly multilingual leakage and mixed-model instability—highlight important open problems in parameter-efficient fine-tuning and model composition. Solving these problems would unlock the full potential of modular, composable style systems that can blend linguistic attributes flexibly and coherently.

**Practical Impact:** Despite current limitations, the parameter efficiency of our approach (96.25% parameter reduction) makes style adaptation accessible for resource-constrained applications. Even if mixed models require further development, the ability to train individual style adapters quickly and cheaply has immediate practical value for applications requiring single-style generation at scale.

## Acknowledgments

This research was conducted using Google Colab with GPU acceleration. We thank the Hugging Face team for their Transformers and PEFT libraries, which enabled rapid prototyping of our approach.

## References

- [1] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., ... & Chen, W. (2021). *LoRA: Low-Rank Adaptation of Large Language Models*. arXiv preprint arXiv:2106.09685.
- [2] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... & Liu, P. J. (2020). *Exploring the limits of transfer learning with a unified text-to-text transformer*. Journal of Machine Learning Research, 21(140), 1-67.
- [3] Prabhumoye, S., Tsvetkov, Y., Salakhutdinov, R., & Black, A. W. (2018). *Style transfer through back-translation*. arXiv preprint arXiv:1804.09000.
- [4] Ficler, J., & Goldberg, Y. (2017). *Controlling linguistic style aspects in neural language generation*. arXiv preprint arXiv:1707.02633.
- [5] Li, J., Jia, R., He, H., & Liang, P. (2018). *Delete, retrieve, generate: a simple approach to sentiment and style transfer*. arXiv preprint arXiv:1804.06437.
- [6] Lample, G., Conneau, A., Denoyer, L., & Ranzato, M. A. (2019). *Cross-lingual language model pretraining*. arXiv preprint arXiv:1901.07291.
- [7] Li, X. L., & Liang, P. (2021). *Prefix-tuning: Optimizing continuous prompts for generation*. arXiv preprint arXiv:2101.00190.
- [8] Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., ... & Gelly, S. (2019). *Parameter-efficient transfer learning for NLP*. In International Conference on Machine Learning (pp. 2790-2799). PMLR.
- [9] Lester, B., Al-Rfou, R., & Constant, N. (2021). *The power of scale for parameter-efficient prompt tuning*. arXiv preprint arXiv:2104.08691.
- [10] Wortsman, M., Ilharco, G., Gadre, S. Y., Roelofs, R., Gontijo-Lopes, R., Morcos, A. S., ... & Schmidt, L. (2022). *Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time*. In International Conference on Machine Learning (pp. 23965-23998). PMLR.
- [11] Ilharco, G., Ribeiro, M. T., Wortsman, M., Gururangan, S., Schmidt, L., Hajishirzi, H., & Farhadi, A. (2022). *Editing models with task arithmetic*. arXiv preprint arXiv:2212.04089.
- [12] Reimers, N., & Gurevych, I. (2019). *Sentence-BERT: Sentence embeddings using Siamese BERT-networks*. arXiv preprint arXiv:1908.10084.
- [13] Li, J., Galley, M., Brockett, C., Gao, J., & Dolan, B. (2016). *A diversity-promoting objective function for neural conversation models*. arXiv preprint arXiv:1510.03055.