VIB-Simp: IB-Inspired Output-Logit Regularization for German Text Simplification

Anonymous ACL submission

Abstract

German text simplification requires reducing lexical and structural complexity while preserving meaning, yet standard fine-tuning optimizes reference imitation without directly controlling the predictive distribution where lexical choices are made. We propose VIB-Simp, a lightweight IB-inspired output-logit regularization framework for German text simplification. Instead of estimating mutual information or adding latent variables, VIB-Simp regularizes first- and second-order logit moments with a closed-form Gaussian KL surrogate, acting as a scale-and-dispersion stabilizer for lexical competition. An optional unlikelihood-style lexical penalty further suppresses probability mass assigned to difficult lexical candidates.

Experiments on TextComplexityDE and 20Min show consistent gains across LLaMA-3.2-3B, Qwen-2.5-3B, mBART, GerPT-2, and mGPT, with 12–18% relative SARI improvements. Additional baselines, length-controlled analysis, ablations, expert ratings, and logit diagnostics show that the gains are not explained by shortening or dictionary-only effects. The results support an IB-style view of simplification: preserve meaning-critical content while suppressing unnecessary lexical complexity.

1 Introduction

Automatic Text Simplification (ATS) aims to reduce linguistic complexity while preserving meaning, making text more accessible to language learners and readers with limited proficiency. It has been studied through rule-based rewriting, lexical substitution, neural generation, and recent LLM-based simplification methods (Štajner and Popović, 2016; Suter et al., 2016; Säuberli et al., 2020; Anschütz et al., 2023; Klöser et al., 2024). Compared with English ATS, where large benchmarks, strong pretrained models, and increasingly mature evaluation protocols have driven substantial progress, German ATS remains less developed. Its rich morphology, compound formation, flexible word order, and limited parallel simplification data make it difficult to learn stable rewriting behavior from standard supervision alone.

Existing German ATS studies have explored rule-based rewriting, lexical substitution, sequence-to-sequence fine-tuning, prompting, and data augmentation. These directions address important parts of the problem, but the training objective itself often remains weakly aligned with the central requirement of simplification. Standard fine-tuning mainly teaches the model to imitate reference outputs, while prompting and data augmentation improve generation coverage without directly shaping the model's output distribution. This often leads to unstable simplification behavior: some systems improve surface readability through shorter outputs but risk semantic loss, while others preserve meaning more safely but retain difficult compounds, complex phrases, or lexically demanding expressions.

This gap is especially important because simplification is naturally an output-side information selection problem. A good simplified sentence should keep meaning-critical information while reducing linguistic complexity that does not help comprehension. This view connects ATS to the information bottleneck principle: preserve relevant information and compress unnecessary information. For text generation, the output distribution is where this selection is finally made, since logits determine which words are generated and which alternatives are suppressed. This makes the output-logit space a natural intervention point for simplification: it is close enough to the generated text to influence lexical choice directly, yet general enough to remain independent of a particular model architecture or hand-crafted rewriting rule.

Motivated by this perspective, we propose **VIB-Simp**, a lightweight regularization framework inspired by the information bottleneck principle.

VIB-Simp applies an IB-inspired logit-moment regularizer during fine-tuning, providing a lightweight way to shape the model's predictive distribution without architectural modification or stochastic latent variables. It also integrates an optional difficulty-aware lexical guidance component, which encourages more effective replacement of difficult words and complex expressions. The resulting framework combines global output-level control with targeted lexical guidance, matching the two main needs of German ATS.

We evaluate VIB-Simp on sentence-level and document-level German simplification benchmarks, including TextComplexityDE and 20Min, across decoder-only and encoder–decoder architectures. Results show consistent improvements over zero-shot and standard fine-tuning baselines in simplification quality, readability, semantic preservation, and fluency. Diagnostic analyses further indicate that the gains are linked to better rewriting behavior and lexical simplification, rather than output shortening alone.

Our main contributions are:

- We motivate German ATS as an output-side information selection problem and identify a training-objective gap in existing methods.
- We propose **VIB-Simp**, a lightweight IB-inspired output-logit regularizer that shapes the predictive distribution during simplification fine-tuning without architectural modification or stochastic latent variables.
- We combine global output-distribution regularization with optional difficulty-aware lexical guidance for difficult words, compounds, and complex expressions.
- We show consistent gains across TextComplexityDE and 20Min, achieving 12–18% relative SARI improvements. Controlled length analyses and expert diagnostic evaluation suggest that the gains are not explained by output shortening alone.

2 Related Work

2.1 German Text Simplification

German text simplification has been studied through rule-based rewriting, lexical substitution, corpus construction, sequence-to-sequence models, and recent LLM-based fine-tuning approaches (Hancke et al., 2012; Das Netzwerk Leichte Sprache, 2013; Maaß, 2020; Suter et al., 2016; Säuberli et al., 2020; Spring et al., 2021; Schomacker et al., 2023; Stodden et al., 2023; Fruth et al., 2024). Early work emphasized linguistic rules and readability features, while later neural approaches focused on parallel simplification data, style-specific pretraining, and semi-synthetic data augmentation. These methods improve fluency and coverage, but they usually rely on standard cross-entropy fine-tuning and provide limited direct control over the model's output distribution. As a result, German ATS systems may either retain difficult compounds and technical phrasing, or simplify aggressively at the cost of semantic preservation.

2.2 German-Specific Resources and Evaluation

Recent work has moved toward stronger German-specific resources. German4All (Anschütz et al., 2025) introduces readability-controlled German paraphrasing data and models, making it possible to study simplification across explicit readability levels. In contrast, DETECT (Korobeynikova et al., 2025) focuses on German-specific simplification evaluation and highlights that standard metrics may not fully capture the joint requirements of simplicity, meaning preservation, and fluency. These efforts are highly complementary to our work. German4All primarily contributes data and readability-controlled generation resources, while DETECT studies evaluation. VIB-Simp instead studies the training objective itself: rather than introducing a new dataset or metric, we ask whether output-logit regularization can make fine-tuning more stable and better aligned with lexical simplification.

2.3 Information Bottleneck and Output-Space Regularization

The information bottleneck principle formalizes learning as preserving task-relevant information while compressing irrelevant variation (Tishby et al., 2000). Variational and stochastic extensions have been used to regularize learned representations and neural networks (Alemi et al., 2016; Achille and Soatto, 2018; Saxe et al., 2019). However, most existing methods operate on latent representations and introduce additional architectural or optimization costs. VIB-Simp takes inspiration from this principle but does not attempt to estimate mutual information directly. Instead, it uses a closed-form Gaussian KL over output-logit moments as a lightweight scale-and-dispersion sta-

bilizer for conditional generation. This distinguishes VIB-Simp from latent-space VIB methods, label smoothing, and generic regularizers: the target supervision is unchanged, while the predictive logit space is regularized to reduce unstable lexical competition during simplification. Unlike entropy penalties that directly modify confidence over the predictive distribution, our surrogate regularizes logit moment statistics while leaving target-token supervision unchanged.

3 The VIB-Simp Framework

3.1 Problem Formulation

Let $\mathcal{X}$ denote the space of complex German texts and $\mathcal{Y}$ the space of simplified texts. Given a complex input $X = (x_1, \ldots, x_n) \in \mathcal{X}$, the goal is to generate a simplified output $Y = (y_1, \ldots, y_m) \in \mathcal{Y}$ that preserves meaning, reduces lexical and structural complexity, and remains grammatical.

We formulate simplification as conditional generation. At timestep t, the model produces logits $z_t \in \mathbb{R}^{|\mathcal{V}|}$ over the vocabulary $\mathcal{V}$ and defines

$$p(y_t \mid X, y_{<t}; \theta) = \mathrm{softmax}(z_t).$$

The output-logit space is therefore the point at which lexical alternatives compete during generation. VIB-Simp uses this space as the intervention point: rather than changing the model architecture or introducing latent variables, it regularizes the predictive logits produced during standard fine-tuning.

3.2 Framework Overview and Loss Interaction

VIB-Simp optimizes a composite objective:

$$L_{total} = L_{CE} + \lambda_{VIB} L_{VIB} + \lambda_{diff} L_{diff}. \quad (1)$$

The cross-entropy term L_{CE} provides the main target-token supervision and preserves task fidelity. The IB-inspired term L_{VIB} acts as a global output-space regularizer: it constrains the scale and dispersion of predictive logits, encouraging more compact and stable lexical competition. The difficulty-aware term L_{diff} acts as an optional local lexical controller: when a difficult-word lexicon is available, it penalizes probability mass assigned to difficult lexical candidates that are not required for meaning preservation.

This separation is important for German simplification. Global logit regularization can stabilize generation even without a lexicon, while the lexical module can be plugged in when additional control over difficult words, compounds, or domain-specific terminology is desired. Thus, VIB-Simp can operate as $L_{CE} + \lambda_{VIB} L_{VIB}$ alone, or as the full objective in Eq. 1 when lexical guidance is available. This makes the framework flexible rather than dependent on a fixed static dictionary.

Figure 1 summarizes the training pipeline. VIB-Simp applies the three loss terms directly to the output logits during fine-tuning, so the method requires no architectural modification and can be used with both decoder-only and encoder–decoder models.

3.3 Loss Function Components

We now detail the two proposed control terms in Eq. 1. The standard cross-entropy term provides target-token supervision.

3.3.1 IB-Inspired Logit-Moment Regularization

Let a batch contain B sequences, each of length T, with vocabulary size V. At each timestep t for sample i, the model produces logits $z_{i,t} \in \mathbb{R}^V$. We use the information bottleneck principle as a motivation for controlling unnecessary predictive dispersion during simplification, but we do not estimate mutual information directly. Instead, we introduce a lightweight output-space surrogate that regularizes the first and second-order moments of the model's logits. This keeps the method architecture-agnostic and avoids stochastic latent variables or additional encoder modules.

We approximate logit statistics with a moment-matched Gaussian $q(z|\mu_{i,t}, \sigma^2_{i,t})$ and penalize its KL divergence to a unit normal prior $p(z) = \mathcal{N}(0, 1)$. The unit normal prior is used as a fixed scale reference, not as a claim about the true logit distribution. Penalizing the mean discourages global vocabulary bias, while penalizing the variance controls over-dispersed lexical competition. Thus, L_{VIB} acts as a lightweight scale-and-dispersion stabilizer rather than a full mutual-information estimator; L_{CE} prevents generic logit shrinking. The resulting closed-form KL is therefore used operationally as an output-logit surrogate, while cross-entropy remains the main source of task-fidelity supervision.

Unlike label smoothing, which flattens the target distribution (Müller et al., 2019), VIB-Simp keeps gold-token supervision unchanged and regularizes

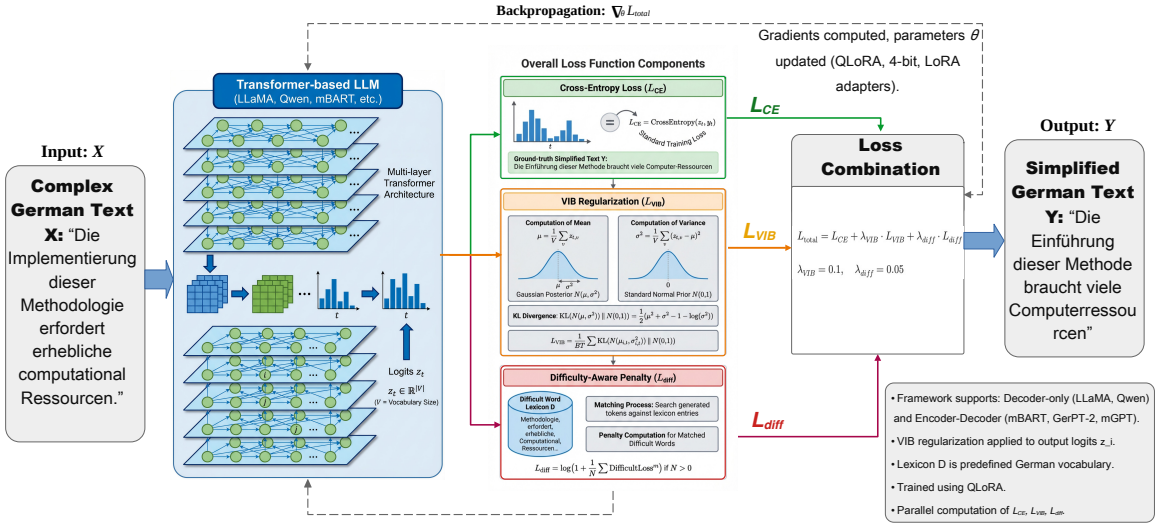

Figure 1: VIB-Simp overview. The training objective combines target-token cross-entropy L_{CE}, IB-inspired logit-moment regularization L_{VIB}, and an optional unlikelihood-style lexical penalty L_{diff} that discourages high probability mass on difficult lexical candidates.

the model's own logit moments. This encourages stable lexical competition while preserving confidence on meaning-critical tokens.

For each (i, t), we compute the first two moments across the vocabulary dimension:

- $\mu_{i,t} = \frac{1}{V}\sum_{v=1}^{V} z_{i,t,v}$,
- $\sigma^2_{i,t} = \frac{1}{V}\sum_{v=1}^{V}(z_{i,t,v} - \mu_{i,t})^2$, with $\sigma_{i,t}$ passed through a softplus to ensure positivity.

The KL divergence between two Gaussians admits a closed form:

$$\mathrm{KL}(q\|p) = \frac{1}{2}\left(\mu^2 + \sigma^2 - 1 - \log\sigma^2\right).$$

Aggregating over timesteps and batch samples, we define:

$$L_{VIB} = \frac{1}{BT}\sum_{i=1}^{B}\sum_{t=1}^{T}\left[\frac{1}{2}\left(\mu^2_{i,t} + \sigma^2_{i,t} - 1 - \log(\sigma^2_{i,t} + \epsilon)\right)\right], \tag{2}$$

where $\epsilon = 10^{-8}$ ensures numerical stability. This term regularizes the moment structure of the output logits, encouraging more compact and stable predictive behavior. It complements L_{CE}, which continues to provide the main supervision for semantic fidelity and target-token prediction.

3.3.2 Difficulty-Aware Lexical Simplification

This component discourages unnecessary lexically complex choices through an auxiliary unlikelihood-style objective, following the general idea of unlikelihood training that penalizes probability mass assigned to undesirable tokens (Welleck et al., 2020). It consists of three parts: (1) difficult-word lexicon construction, (2) a probability-mass penalty in the loss function, and (3) an evaluation metric (DHR) to assess lexical simplification effectiveness.

3.3.3 Difficult Word Lexicon

We build a German difficult-word lexicon $\mathcal{D}$ via GPT-assisted annotation with human validation, motivated by German Easy/Plain Language guidelines and resources (Das Netzwerk Leichte Sprache, 2013; Maaß, 2020). Two native German linguists independently labeled a stratified sample of 500 words across frequency deciles. The annotations reached Cohen's $\kappa = 0.74$; the GPT-assisted lexicon achieved 0.83 precision and 0.79 recall against adjudicated labels, and correlated with Easy Language difficulty levels at Spearman $\rho = 0.68$.

3.3.4 Difficult Word Penalty

To explicitly discourage unnecessary complex lexical choices, we formulate the difficulty-aware term as an unlikelihood-style lexical penalty. The term penalizes the probability mass assigned to difficult lexical candidates, rather than treating difficult tokens as target spans to be predicted. It therefore acts as a local simplification pressure when a difficult-word lexicon is available, while leaving the cross-entropy objective responsible for preserving meaning-critical target tokens.

Let $\mathcal{D}$ denote the difficult-word lexicon and let $\mathcal{V}_{\mathcal{D}} \subseteq \mathcal{V}$ denote the set of vocabulary tokens associated with difficult lexical items after tokenization. At timestep t for sample i, the model produces a

predictive distribution

$$p_{i,t} = \mathrm{softmax}(z_{i,t}).$$

We define the probability mass assigned to difficult lexical candidates as

$$p^{\mathcal{D}}_{i,t} = \sum_{v \in \mathcal{V}_{\mathcal{D}}} p_{i,t,v}.$$

The difficulty-aware lexical penalty is then:

$$L_{\text{diff}} = -\frac{1}{M+\epsilon} \sum_{i=1}^{B} \sum_{t=1}^{T} m_{i,t} \log\left(1 - p^{\mathcal{D}}_{i,t} + \epsilon\right), \tag{3}$$

$$M = \sum_{i=1}^{B} \sum_{t=1}^{T} m_{i,t},$$

where $m_{i,t} \in \{0, 1\}$ is a lexical-control mask indicating positions where simplification pressure is applied, M is the number of active masked positions, and $\epsilon = 10^{-8}$ ensures numerical stability. In our implementation, the mask is activated for non-padding target positions in examples whose source sentence contains difficult lexical items. This avoids applying the penalty to padding tokens and focuses the auxiliary pressure on examples where lexical simplification is relevant.

Minimizing Eq. (3) penalizes high probability mass assigned to difficult lexical candidates. When $p^{\mathcal{D}}_{i,t}$ is small, the penalty is close to zero; when the model assigns high probability to difficult tokens, the penalty increases sharply. The term therefore acts as a local lexical controller that discourages unnecessary difficult-word generation, while the cross-entropy objective remains responsible for preserving meaning-critical target tokens. This makes L_{diff} an optional auxiliary control signal rather than a replacement for the standard sequence generation objective.

3.3.5 Difficulty Hit Rate (DHR) Metric

To evaluate the model's effectiveness in lexical simplification, we propose the Difficulty Hit Rate (DHR) metric, which measures the proportion of difficult words in the generated output:

$$\text{DHR} = \frac{|\{w | w \in W_{output} \cap \mathcal{D}\}|}{|W_{output}|} \tag{4}$$

where W_{output} is the set of generated words and $\mathcal{D}$ is the difficult word dictionary. A lower DHR indicates better lexical simplification performance, as the model produces fewer difficult words in its output.

4 Experiments

4.1 Experimental Setup

4.1.1 Datasets

We evaluate on two public German text simplification benchmarks. **TextComplexityDE** (Naderi et al., 2019) contains 1,019 sentence pairs with three simplification levels; we follow the official stratified train/validation/test split. **20Min** (Spring et al., 2021) provides 17,905 document-level pairs with standard validation and test splits, which we adopt unchanged for comparability with prior work. Minimal preprocessing is applied to preserve original style.

4.1.2 Models

We evaluate both decoder-only and encoder–decoder architectures. For sentence-level simplification, we fine-tune **LLaMA 3.2-3B** and **Qwen 2.5-3B** (Grattafiori et al., 2024; Qwen Team, 2025). For document-level simplification, we evaluate **mBART**, **GerPT-2**, and **mGPT**; mBART follows multilingual denoising pretraining (Liu et al., 2020). Fine-tuning uses parameter-efficient adaptation related to LoRA/QLoRA (Hu et al., 2022; Dettmers et al., 2023). These models represent strong multilingual and German-specific baselines commonly used in recent simplification studies.

4.1.3 Evaluation Metrics

We evaluate simplification quality using standard automatic metrics. SARI is the primary metric for lexical simplification operations (Xu et al., 2016), and BERTScore measures semantic preservation (Zhang et al., 2020). FKGL and FRE are reported as surface-level readability indicators because they capture sentence length and word-form complexity, which remain relevant for German simplification; however, they were originally developed for English and should not be interpreted as fully German-specific readability measures. We therefore interpret them together with SARI, BERTScore, DHR, grammar errors, and expert diagnostic evaluation.

We define compression ratio as

$$\text{Comp} = \frac{|\hat{Y}|}{|X|},$$

where $|\hat{Y}|$ and $|X|$ are the tokenized lengths of the generated output and source input under the same preprocessing. Since raw output length can vary with filtering and decoding, Table 5 is used as the main compression-controlled test.

5 Results

5.1 TextComplexityDE Dataset

5.1.1 Main Analysis

We report results with and without grammar filtering using LanguageTool (Naber, 2003). When filtering is enabled, we exclude outputs with more than one error-level flag and report the remaining error rate.

We compare three training strategies: **Zero-Shot**, which uses the base model without fine-tuning; **Normal**, which fine-tunes with $L_{CE} + \lambda_{diff} L_{diff}$; and **VIB-Simp**, which adds the IB-inspired output-logit regularizer, yielding $L_{CE} + \lambda_{VIB} L_{VIB} + \lambda_{diff} L_{diff}$. This design isolates the effect of local lexical control and the additional contribution of the output-logit regularizer.

Table 1 shows that VIB-Simp improves simplification quality, semantic fidelity, readability, and lexical control on TextComplexityDE. Under grammar filtering, it achieves the best SARI, BERTScore, FKGL, FRE, and DHR scores. The lower DHR supports the intended effect of the unlikelihood lexical penalty, while the BERTScore improvement suggests that lexical simplification does not come at the cost of semantic fidelity. The same pattern holds without grammar filtering, indicating that the gains are not simply an artifact of removing noisy generations.

5.1.2 Additional Baselines and Regularizers

Appendix A reports comparisons with public German ATS systems and standard regularizers. VIB-Simp remains ahead of these baselines on TextComplexityDE, suggesting that the gain is not due to generic regularization alone.

5.2 20Min Dataset

5.2.1 Main Analysis

Tables 2 and 3 show that the same pattern extends to document-level simplification. VIB-Simp achieves the best or highly competitive SARI and FKGL scores across both decoder-only models (LLaMA and Qwen) and encoder–decoder or German-oriented models (mBART, GerPT-2, and mGPT). This suggests that the output-logit regularizer is not tied to a specific architecture or dataset scale. This is important because 20Min is substantially larger and document-level, so improvements on this benchmark provide evidence that the regularizer is not only tuned to short sentence-level simplification.

The 20Min results also show that readability gains do not come from noisier generation. VIB-Simp maintains low grammatical error rates while improving simplification quality, indicating that the regularizer stabilizes generation rather than merely shortening outputs. All results are averaged over three independent fine-tuning runs; full seed-wise statistics are provided in Appendix B.

5.3 Controlling for Output Length and Verifying Non-Trivial Simplification

One concern is that VIB-Simp might improve automatic scores mainly by producing shorter outputs. We test this with two complementary analyses: SARI decomposition and controlled-compression evaluation. Table 4 shows that VIB-Simp does not improve by deleting more content: its DELETE score is nearly identical to standard fine-tuning, while KEEP and ADD increase substantially and BERTScore also improves.

The controlled-compression setting in Table 5 provides a direct test of the shortening hypothesis. When output/source length ratios are kept comparable, VIB-Simp still achieves the best SARI, BERTScore, FKGL, and FRE. Together, these results suggest that the gains are not driven by aggressive deletion or trivial shortening.

5.4 Expert Diagnostic Evaluation

To complement automatic metrics, we conducted an expert diagnostic evaluation on 50 selected German simplification cases. The cases were chosen to examine challenging outputs where meaning preservation, simplicity, and fluency are particularly important; the evaluation is therefore diagnostic rather than a population-level preference study. The two German-speaking evaluators scored outputs independently using a 1–5 rubric for meaning preservation, simplicity, and fluency/grammaticality. System names were hidden and outputs were presented in randomized order within each case.

As shown in Table 6, VIB-Simp improves the mean score from 2.66 to 4.08 and receives a higher two-rater average in 44/50 cases, with 2 ties and 4 lower-scored cases. Both evaluators show the same directional pattern, with VIB-Simp scoring higher in 39/50 cases for each rater. Rater-specific results, qualitative examples, and error breakdowns are reported in Appendices C, **??**, and E.

Table 1: Performance comparison on the TextComplexityDE dataset using token-level lexical penalty. VIB-Simp consistently improves simplification quality, semantic fidelity, and readability. Best results in bold.

Strategy	Penalty	Grammar	SARI ↑	BERT Score↑	FKGL ↓	FRE ↑	DHR ↓	Error ↓
Zero-Shot	Token-level	Included	30.76	0.710	8.30	55.81	0.1494	12.0%
Normal	Token-level	Included	32.09	0.720	7.76	54.20	0.1610	10.0%
VIB	Token-level	Included	**35.67**	**0.728**	**6.21**	**73.52**	**0.0891**	**8.0%**
Zero-Shot	Token-level	Not included	31.88	0.630	7.58	55.87	0.1459	0.0%
Normal	Token-level	Not included	31.60	0.680	7.77	52.20	0.1376	0.0%
VIB	Token-level	Not included	**34.31**	**0.725**	**5.55**	**72.99**	**0.0827**	0.0%

Table 2: 20Min results for decoder-only LLMs. Best results in bold.

Model	Strategy	SARI↑	Grammar Errors↓	Sent. Length↓	FKGL↓
LLAMA3.2-3b	Zero-Shot	38.85	0.43	**40.57**	7.49
LLAMA3.2-3b	Normal	37.66	1.90	41.20	7.25
LLAMA3.2-3b	VIB	**39.61**	**0.35**	43.50	**5.87**
Qwen2.5-3b	Zero-Shot	40.12	**0.25**	42.50	6.79
Qwen2.5-3b	Normal	37.17	0.65	**39.10**	9.02
Qwen2.5-3b	VIB	**41.98**	0.30	41.75	**6.54**

5.5 In-Depth Model Analysis

5.5.1 Component Contribution Analysis.

Table 7 separates the global logit regularizer from the local lexical module. The VIB-only variant already yields strong gains, showing that the output-logit regularizer is effective without a dictionary. The lexicon-only variant improves ADD, suggesting useful local lexical substitution. The full model achieves the best overall SARI, supporting the view that global distribution control and local lexical suppression are complementary.

5.5.2 Predictive Distribution Diagnostics

To examine how the proposed output-logit regularizer affects predictive behavior, we measure two diagnostic statistics: (1) the average surrogate KL divergence between the logit moment-matched Gaussian and the unit normal prior, and (2) the differential entropy of logits estimated via Monte Carlo. The corresponding quantitative diagnostics are reported in Table 8.

Across both datasets, VIB-Simp reduces the surrogate KL by 35–48% and logit entropy by roughly 0.7 bits, indicating that the predictive distribution becomes more compact and focused. This pattern aligns with the reduction in DHR (from 0.16 to 0.09), suggesting that more stable token selection also supports simpler lexical choices. The histogram diagnostic in Figure 2 provides a complementary visual check of this distributional change.

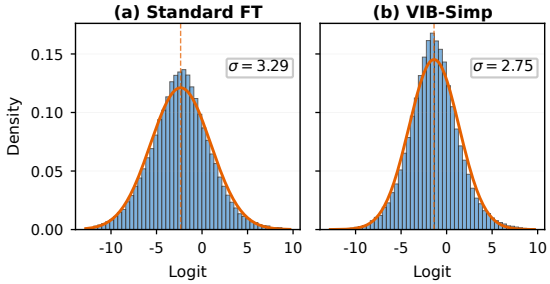

Figure 2: Vocabulary-logit histograms with moment-matched Gaussian fits. VIB-Simp shows a narrower fitted dispersion ($\sigma = 2.75$) than standard fine-tuning ($\sigma = 3.29$).

5.5.3 Validity of the Gaussian Moment Approximation

The Gaussian approximation in Eq. 2 is a moment-based regularization surrogate, not a claim that vocabulary-level logits are exactly Gaussian. Directly regularizing the full predictive distribution would be costly, so we use a moment-matched Gaussian to summarize logit scale and dispersion with a closed-form KL term. The cross-entropy loss remains the main task supervision, while the surrogate KL discourages extreme or unstable logit dispersion.

Figure 2 provides a diagnostic check. The histograms are not perfectly Gaussian, but their first- and second-order moments provide a useful training signal; VIB-Simp shows a narrower fitted dispersion than standard fine-tuning, consistent with

Table 3: 20Min results for encoder–decoder and German-oriented models. Best results in bold.

Model	Strategy	SARI↑	FKGL↓	FRE↑	Grammar Errors↓	Sent Len↓
mBART	Zero-Shot	29.4	8.2	59.3	7.1	18.6
mBART	Normal	32.1	7.9	61.0	6.3	17.3
mBART	VIB	**34.6**	**7.5**	**63.7**	**5.5**	**16.9**
GerPT2	Zero-Shot	28.9	8.4	58.1	7.6	19.1
GerPT2	Normal	30.2	8.0	59.8	6.8	18.2
GerPT2	VIB	**33.3**	**7.7**	**61.9**	**5.9**	**17.5**
MGPT	Zero-Shot	27.6	8.6	56.7	8.0	19.5
MGPT	Normal	29.7	8.1	58.5	7.1	18.3
MGPT	VIB	**32.5**	**7.8**	**60.4**	**6.2**	**17.6**

Table 4: Analysis on TextComplexityDE. VIB-Simp improves SARI mainly through stronger KEEP and ADD behavior, while maintaining a DELETE score comparable to standard fine-tuning and achieving the best BERTScore.

System	SARI↑	ADD↑	KEEP↑	DELETE↑	BERT-F1↑
Zero-shot / Train-free	31.59	2.08	10.57	80.05	0.705
Normal FT	30.56	1.40	7.84	79.74	0.672
VIB-Simp	**35.68**	**3.84**	**18.13**	**80.08**	**0.730**

Table 5: Effect of controlling for compression rate on TextComplexityDE (token-level, grammar filtering).

Strategy	Comp	SARI↑	BERT-F1↑	FKGL↓	FRE↑
Zero-Shot	1.069	33.14	0.693	10.77	32.27
Standard FT	**0.878**	31.88	0.720	7.02	46.42
VIB-Simp	0.961	**35.37**	**0.725**	**6.16**	**51.13**

the lower surrogate KL and entropy in Table 8.

6 Conclusion

We presented VIB-Simp, a lightweight IB-inspired output-logit regularization framework for German text simplification. By stabilizing predictive logit moments and optionally adding an unlikelihood-style lexical penalty, VIB-Simp improves simplification without architectural modification.

Across sentence- and document-level benchmarks, VIB-Simp achieves the best or highly competitive SARI and BERTScore, while also improving surface readability indicators and grammar quality, also maintaining low grammar error rates. Controlled-compression analysis confirms that these gains are not due to trivial shortening, and ablation studies demonstrate that both global logit regularization and local lexical control contribute complementary improvements.

Expert diagnostic evaluation further supports the method's effectiveness: human raters scored VIB-Simp outputs higher than baselines in 44/50 cases on average, with the same directional pattern across individual raters. Logit-distribution diagnostics indicate more compact and stable predictive behavior, consistent with the intended effect of the VIB regularizer. Together, these results suggest that the observed improvements reflect better preservation and rewriting rather than output shortening or dictionary-only effects.

System	Mean ↑	Median ↑	≥4 ↑	≤2 ↓
Baseline	2.66	2.50	16.0%	46.0%
VIB-Simp	**4.08**	**4.00**	**64.0%**	**0.0%**

Table 6: Expert diagnostic evaluation on 50 selected German simplification cases. Scores are averaged over two German-speaking evaluators and range from 1 to 5.

Table 7: Ablation study of the global and local control mechanisms on TextComplexityDE. The logit-level regularizer is effective even without the lexicon, while the unlikelihood lexical module provides additional controllability when plugged in.

System	SARI↑	ADD↑	KEEP↑	DELETE↑	BERT-F1↑
Zero-shot / Train-free	31.59	2.08	10.57	80.05	0.705
Lexicon-only	33.52	3.48	13.95	**80.48**	0.680
VIB-only	34.83	2.76	**18.56**	77.83	**0.733**
Full VIB-Simp	**35.68**	**3.84**	18.13	80.08	0.730

Table 8: Predictive distribution analysis of VIB-Simp. Lower surrogate KL and entropy indicate more compact and stable output distributions under logit-level regularization.

Dataset	Strategy	KL↓	Entropy (bits)↓
TextComp (Qwen)	Standard	1.87	4.21
TextComp (Qwen)	VIB	**0.98**	**3.39**
20Min (Qwen)	Standard	1.45	4.57
20Min (Qwen)	VIB	**0.84**	**3.88**

Limitations

This work has several limitations. First, VIB-Simp is evaluated on two German text simplification benchmarks, TextComplexityDE and 20Min. Although these benchmarks cover both sentence-level and document-level simplification, further evaluation is needed on additional domains, genres, languages, and simplification levels to assess broader generalization.

Second, the difficulty-aware lexical penalty depends on the quality and coverage of the difficult-word lexicon. We validate the lexicon through human annotation, and the ablation study shows that the output-logit regularizer remains effective without the lexical module. Nevertheless, domain-specific or user-specific simplification may require adapting the lexicon to different audiences, technical fields, or readability standards.

Third, the expert diagnostic evaluation is based on selected challenging cases rather than a population-level human preference study. This design is useful for examining meaning preservation, simplicity, and fluency in difficult examples, but it should not be interpreted as an unbiased estimate of general user preference. Larger-scale randomized human evaluations with more annotators would provide stronger evidence.

Finally, while we compare VIB-Simp with public German ATS systems and standard regularizers, we do not include all recent German-specific systems or learned evaluation metrics. FKGL and FRE are also only approximate surface readability indicators for German. Future work should compare against newer readability-controlled systems and German-specific evaluation frameworks such as German4All and DETECT when their evaluation protocols and code are fully available.

References

Alessandro Achille and Stefano Soatto. 2018. Information dropout: Learning optimal representations through noisy computation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2897–2905.

Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, and Kevin Murphy. 2016. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*.

Miriam Anschütz, Joshua Oehms, Thomas Wimmer, Björn Jezierski, and Georg Groh. 2023. Language models for german text simplification: Overcoming parallel data scarcity through style-specific pre-training. *arXiv preprint arXiv:2305.12908*.

Miriam Anschütz, Thang M. Pham, Elias Nasrallah, Mark Müller, Cezar-Gabriel Craciun, and Georg Groh. 2025. German4All – a dataset and model for readability-controlled paraphrasing in german. In *Proceedings of the 18th International Natural Language Generation Conference*, pages 390–407.

Das Netzwerk Leichte Sprache. 2013. Die regeln für leichte sprache.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient finetuning of quantized LLMs. *arXiv preprint arXiv:2305.14314*.

Lisa Fruth, Robin Jegan, and Andreas Henrich. 2024. An approach towards unsupervised text simplification on paragraph-level for german texts. In *Proceedings of the Workshop on DeTermIt! Evaluating Text Difficulty in a Multilingual Context @ LREC-COLING 2024*, pages 77–89.

Aaron Grattafiori and 1 others. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Julia Hancke, Sowmya Vajjala, and Detmar Meurers. 2012. Readability classification for german using lexical, syntactic, and morphological features. In *Proceedings of the International Conference on Computational Linguistics*.

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

Leon Klöser, Moritz Beele, Jan Niklas Schagen, and Benjamin Kraft. 2024. German text simplification: Finetuning large language models with semi-synthetic data. *arXiv preprint arXiv:2402.10675*.

Maria Korobeynikova, Alessia Battisti, Lukas Fischer, and Yingqiang Gao. 2025. DETECT: Determining ease and textual clarity of german text simplifications. *Preprint*, arXiv:2510.22212.

Xiaobo Liang, Lijun Wu, Juntao Li, Yue Wang, Qi Meng, Tao Qin, Wei Chen, Min Zhang, and Tie-Yan Liu. 2021. R-Drop: Regularized dropout for neural networks. In *Advances in Neural Information Processing Systems*.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Myle Ott, Michael Auli, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Christiane Maaß. 2020. *Easy Language–Plain Language–Easy Language Plus: Balancing Comprehensibility and Acceptability*. Frank & Timme.

Rafael Müller, Simon Kornblith, and Geoffrey Hinton. 2019. When does label smoothing help? In *Advances in Neural Information Processing Systems*.

Daniel Naber. 2003. A rule-based style and grammar checker. In *Proceedings of the Workshop on Style and Composition*.

Babak Naderi, Salar Mohtaj, Katharina Ensikat, and Sebastian Möller. 2019. Subjective assessment of text complexity: A dataset for german language. *arXiv preprint arXiv:1904.07733*.

Qwen Team. 2025. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Andreas Säuberli, Sarah Ebling, and Martin Volk. 2020. Benchmarking data-driven automatic text simplification for german. In *Proceedings of the 1st Workshop on Tools and Resources to Empower People with Reading Difficulties (READI)*, pages 41–48.

Andrew M. Saxe, Yamini Bansal, Joel Dapello, Madhu Advani, Artemy Kolchinsky, Brendan Tracey, and David D. Cox. 2019. On the information bottleneck theory of deep learning. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124020.

Thorsten Schomacker, Tillmann Dönicke, and Marina Tropmann-Frick. 2023. Exploring automatic text simplification of german narrative documents. *arXiv preprint arXiv:2312.09907*.

Nicolas Spring, Abelardo R. Gonzales, and Sarah Ebling. 2021. Exploring german multi-level text simplification. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 1339–1349.

Sanja Štajner and Maja Popović. 2016. Can text simplification help machine translation? In *Proceedings of the 19th Annual Conference of the European Association for Machine Translation*, pages 230–242.

Regina Stodden, Omar Momen, and Laura Kallmeyer. 2023. DEPLAIN: A german parallel corpus with intralingual translations into plain language for sentence and document simplification. *arXiv preprint*.

Julia Suter, Sarah Ebling, and Martin Volk. 2016. Rule-based automatic text simplification for german. In *Proceedings of the Conference on Computational Linguistics*.

Naftali Tishby, Fernando C. Pereira, and William Bialek. 2000. The information bottleneck method. *arXiv preprint arXiv:physics/0004057*.

Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. 2020. Neural text generation with unlikelihood training. In *International Conference on Learning Representations*.

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. Optimizing statistical machine translation for text simplification. In *Transactions of the Association for Computational Linguistics*, volume 4, pages 401–415.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

A Additional Baselines and Regularizers

To strengthen the comparison, we benchmark VIB-Simp against public German ATS systems from Anschütz et al. (Anschütz et al., 2023) and Klöser et al. (Klöser et al., 2024), as well as label smoothing (Müller et al., 2019) and R-Drop (Liang et al., 2021). Table 9 shows that VIB-Simp remains ahead of these alternatives on TextComplexityDE. This suggests that the gain is not due to generic regularization alone.

Table 9: Comparison with additional German ATS systems and regularizers on TextComplexityDE (token-level penalty, grammar filtering).

System	SARI↑	BERTScore↑	FKGL↓	FRE↑	DHR↓
Anschütz et al. (Anschütz et al., 2023)	31.12	0.702	7.85	57.10	0.142
Klöser et al. (Klöser et al., 2024)	33.05	0.715	6.41	68.30	0.117
Label smoothing (Müller et al., 2019)	30.92	0.707	8.20	61.40	0.133
R-Drop (Liang et al., 2021)	32.14	0.718	8.98	53.50	0.127
VIB-Simp (ours)	**35.67**	**0.728**	**6.21**	**73.52**	**0.089**

B Seed Stability

To verify that the 20Min results are not driven by a single random seed, we fine-tuned each configuration three times with independent seeds and report the mean performance in the main tables. Table 10 summarizes the observed standard deviation ranges across all configurations; the variability is modest, supporting the consistency of the improvements reported in the main text.

Table 10: Seed-level variability on the 20Min dataset.

Metric	Std range
SARI	0.18–0.35
FKGL	0.12–0.29
Grammar errors	0.05–0.11

C Rater-specific Diagnostic Scores

Table 11 reports rater-specific diagnostic evaluation results for the 50 selected diagnostic cases used in the main paper. The two evaluators differ in absolute scoring strictness, but both show the same directional preference for VIB-Simp.

Table 11: Rater-specific diagnostic evaluation results. Higher, Tie, and Lower compare VIB-Simp against the baseline for each case.

Rater	Baseline	VIB-Simp	Higher	Tie	Lower
Rater 1	1.94	3.54	39	5	6
Rater 2	3.38	4.62	39	11	0
Average	2.66	4.08	44	2	4

D Human Evaluation Instructions

The following instructions were given to the German-speaking evaluators. Evaluators were asked to give one overall score to each simplification. They were instructed to judge mainly by: (1) meaning preservation, (2) simplicity, and (3) fluency/naturalness. The original sentence shown in the evaluation table is the dataset original sentence, not the gold reference simplification.

Score	Meaning	How to use it
5	Excellent	Meaning is preserved well, the sentence is clearly simpler, and the wording is natural.
4	Good	Mostly correct and simpler, with only minor wording or grammar issues.
3	Acceptable	Partly useful, but simplification is limited or there are noticeable awkward phrases.
2	Poor	Important meaning is lost or distorted, or the output is not clearly simpler.
1	Very poor	Severely incorrect, badly broken, heavily truncated, or contains unrelated polluted text.

Table 12: Human evaluation scoring rubric used for the expert diagnostic evaluation.

E Manual Error Analysis

We manually annotated 200 randomly sampled TextComplexityDE outputs per strategy and categorized each non-perfect sentence by its dominant error type. Table 13 reports absolute counts; lower values indicate fewer errors. VIB-Simp reduces semantic drift and lexical omission errors compared to standard fine-tuning, while also decreasing grammar violations.

Table 13: Manual error breakdown on 200 TextComplexityDE samples.

Error type	Zero-Shot	Standard FT	VIB-Simp
Semantic drift	42	31	**18**
Lexical omission	37	29	**16**
Grammar violation	28	24	**13**
Over-compression	21	18	**15**

F Difficult-word Lexicon Validation

To assess the reliability of the automatically constructed difficult-word lexicon, we conducted a small-scale human validation study. Two native German annotators independently labeled a stratified random sample of 500 words drawn across frequency deciles. We report inter-annotator agreement, precision and recall of the GPT-assisted lexicon against adjudicated human labels, and its correlation with Easy Language difficulty levels.

Table 14: Validation of the GPT-assisted difficult-word lexicon on a 500-word human-labeled sample.

Metric	Value
Human annotator agreement (κ)	0.74
GPT-4 precision vs. humans	0.83
GPT-4 recall vs. humans	0.79
Spearman($\mathcal{D}$, Easy Language level)	0.68

G Auxiliary Automatic Semantic Check

As an auxiliary semantic check beyond automatic metrics, we used GPT-4o to score meaning preservation on a small set of aligned baseline and VIB-Simp outputs. Scores range from 1 to 5, where higher scores indicate stronger retention of the original meaning. This check is intended only as a supplementary diagnostic and is not used as a primary evaluation metric.

Table 15: Auxiliary GPT-4o meaning-preservation scores. Higher is better.

System	Average Meaning Preservation
Baseline	1.75
VIB-Simp	**2.90**

H Artifact Licenses and Intended Use

We use all external artifacts only for academic research on German text simplification. TextComplexityDE is publicly released under the MIT license. The 20Minuten/20Min dataset is used as a public research dataset for German text simplification and summarization; we follow the access and usage conditions specified by the original dataset providers. For pretrained models, we follow the corresponding model licenses and terms of use: LLaMA-3.2 models are governed by the LLaMA 3.2 Community License, while Qwen-2.5 models are released under the license specified in their

official model repositories. All evaluation tools and metrics are used only for research evaluation, and we do not redistribute third-party datasets, pre-trained model weights, or proprietary artifacts. The difficult-word lexicon constructed in this work is used as an auxiliary research artifact for lexical simplification analysis and will be released, if applicable, with documentation of its construction procedure and intended research use.

I Implementation Details

We implement the experiments using HuggingFace Transformers and PEFT. For the main decoder-only experiments, models are fine-tuned using QLoRA-style parameter-efficient adaptation with 4-bit quantized loading and LoRA adapters. The LoRA configuration uses rank $r = 16$, scaling factor $\alpha = 24$, dropout 0.1, and no bias adaptation. LoRA adapters are applied to the attention projection modules `q_proj`, `k_proj`, `v_proj`, and `o_proj`.

Input prompts and target sequences are truncated or padded to a maximum length of 256 tokens. We use a per-device batch size of 4 and gradient accumulation over 8 steps, giving an effective batch size of 32 per device. Models are trained for 12 epochs with mixed-precision training, and at most two checkpoints are saved.

We compare three settings: raw train-free generation, standard QLoRA fine-tuning, and VIB-Simp fine-tuning. The standard baseline optimizes cross-entropy only. VIB-Simp uses the same fine-tuning setup but replaces the loss with a weighted combination of cross-entropy, the logit-moment KL regularizer, and the difficulty-aware lexical penalty. In our implementation, the cross-entropy term is weighted by 0.6, the VIB coefficient is set to $\beta = 0.5$, and the lexical penalty coefficient is set to $\gamma = 1.0$.

For generation, we use German simplification prompts with an in-context example. Decoding uses temperature 0.2, repetition penalty 1.2, and dynamically chosen generation length bounds based on the prompt and source length. The outputs from the three settings are saved as JSON files containing the original inputs, reference simplifications, and generated predictions, and are then used for automatic evaluation and diagnostic analysis.

J Qualitative Examples

The following examples are selected from the diagnostic evaluation set. Scores are two-rater averages on the 1–5 diagnostic scale.

Example 1: Meaning preservation. Source. Diese Abwertung wird auch als Downcycling bezeichnet, während beim Upcycling aus Abfallstoffen eines Prozesses hochwertigere Produkte hergestellt werden können.
Standard FT. Dies nennt man Abwärtswandeln. Dabei kann mit Abfälle hochwertige Produkte hergestellt werden (Upcycling).
VIB-Simp. Diese Abwertung nennt man Downcycling. Beim Upcycling entstehen aus Abfallstoffen hochwertigere Produkte.
Scores. Baseline: 2.0; VIB-Simp: 5.0.

Example 2: Reduced malformed output. Source. Ausgehend von Altpapier-Wiederverwendung wurden zunehmend Technologien erarbeitet, die die Wiederaufbereitung aller Arten von Altstoffen wirtschaftlich machen, wodurch Abfall zu einem bedeutenden Wirtschaftsgut wurde: Geprägt wurde dafür der Ausdruck Sekundärrohstoff.
Standard FT. Aus Altpapier-Wiederverwendung entwickelte man technische Methoden zur Aufbereitung verschiedener altersfreier Materialien. Dadurch ist abgestorbene Materiale ein
VIB-Simp. Aus der Wiederverwendung von Altpapier entstanden Techniken, mit denen viele Altstoffe wirtschaftlich aufbereitet werden konnten. So wurde Abfall zu einem wichtigen Rohstoff.
Scores. Baseline: 1.5; VIB-Simp: 4.5.

Example 3: Simpler lexical paraphrasing. Source. Zudem erfolgt im Laufe des Auslaufprozesses eine Anreicherung von Seifenfilm-stabilisierenden Tensidmolekülen im unteren Bereich der Seifenblase, sodass deren obere Region infolge des relativen Mangels von an die Oberfläche adsorbierten Tensidmolekülen zusätzlich destabilisiert wird.
Standard FT. Während das Film-Prozeß abläuft, kommen stabilisierende Stoffe ins Unterleben der Blase. Das führt dazu, dass die oberste Region der Blase dadurch
VIB-Simp. Beim Auslaufen der Seifenblase sammeln sich unten mehr stabilisierende Stoffe. Dadurch wird der obere Teil noch instabiler.
Scores. Baseline: 2.0; VIB-Simp: 4.5.

Example 4: Remaining limitation. Source. Zudem waren sie überzeugt, im Sinne der neuen

Regierung zu handeln, die Friedensverhandlungen mit der Entente anstrebte.

Standard FT. Auch sie hielten es für wichtig, bei der neuen Regierung im Sinn der Friedensverhandlungen mit der Entente zu helfen.

VIB-Simp. Sie waren außerdem überzeugt, im Sinne der neuen Regierung zu handeln, die Friedensverhandlungen mit der Entente anstrebte.

Comment. VIB-Simp preserves the meaning, but remains too close to the original and does not simplify the sentence enough.

Scores. Baseline: 4.5; VIB-Simp: 4.0.