SANLAPA: ADVERSARIAL MASKED LANGUAGE MODELING VIA DIFFERENTIABLE TEACHER-STUDENT MINIMAX PRE-TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Masked Language Modeling (MLM) has been central to the success of bidirectional encoders, but its standard stochastic masking strategy often allocates training signal to easily predictable tokens, limiting pre-training efficiency. We present **SANLAPA** (which means "dialogue"), an adversarial pre-training framework that replaces random masking with an adaptive masking policy learned through a teacher–student minimax game. A lightweight **Teacher** identifies tokens or spans that maximize the **Student**'s reconstruction loss, while the Student learns to recover them. To enable end-to-end optimization, we introduce a differentiable masking layer based on Gumbel-Softmax relaxation together with a sequential joint training schedule that stabilizes the adversarial dynamics. Unlike prior adversarial ELECTRA-style RTD or AMOS approaches, **SANLAPA** retains reconstruction as the core learning objective and adversarially optimizes *what* to mask. We validate the framework across encoder-only and encoder–decoder architectures, including ALBERT, ModernBERT, ELECTRA, DeBERTaV3, T5, and ByT5, while additionally comparing against the adversarial AMOS baseline on Bangla, English, and Tamil. Across 10 standard English benchmarks and 22 Indic downstream dataset settings, **SANLAPA** yields consistent improvements over corpus- and step-matched baselines, with peak gains of up to +3.13 percentage points (pp) Composite score on discriminative tasks and +4.58 pp Composite score on generative tasks, without increasing the parameter count of the final deployed model. These results show that learning *what to hide* provides a strong and scalable alternative to fixed stochastic masking.

1 INTRODUCTION

Bidirectional encoders like BERT have transformed natural language processing, driven largely by Masked Language Modeling (MLM) Devlin et al. (2019). However, standard MLM relies on fixed, uniform random masking that ignores the model's evolving weaknesses. Consequently, substantial learning capacity is wasted on locally predictable tokens, yielding suboptimal training signals Devlin et al. (2019); Joshi et al. (2020).

While methods like SpanBERT, T5, GLM, and UL2 improve pre-training by masking contiguous segments, they still rely on static heuristics rather than a learned corruption policy Joshi et al. (2020); Raffel et al. (2020); Du et al. (2022); Tay et al. (2022). This raises a fundamental question: can we retain MLM's successful reconstruction objective while actively learning *what* to mask to maximize training efficiency?

We answer this with **SANLAPA** (Bangla for *dialogue*), representing a continuous adversarial exchange where an adaptive masking policy is learned via a teacher–student minimax game. As shown in Figure 1, a lightweight **Teacher** selects tokens or spans maximizing the **Student**'s reconstruction loss, while the Student learns to recover them. This creates a dynamic curriculum shifting corruption toward regions the Student still finds difficult. To enable end-to-end training, we employ a differentiable masking layer via Gumbel-Softmax relaxation Jang et al. (2016) paired with a sequential joint optimization schedule that stabilizes dynamics by evaluating the Teacher against the Student's latest state.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

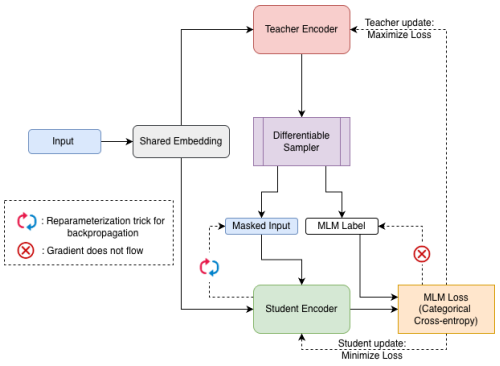

Figure 1: Overview of SANLAPA.

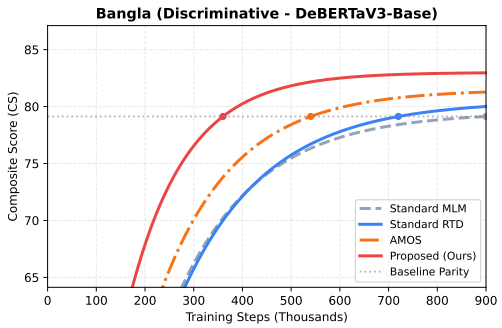

Figure 2: Discriminative convergence dynamics on the **Bangla** corpus.

We validate SANLAPA across encoder (ALBERT, ModernBERT, ELECTRA, DeBERTaV3) and encoder–decoder (T5, ByT5) architectures on English, Bangla, and Tamil (with Hindi in the Appendix). Across 28 downstream settings, SANLAPA consistently outperforms corpus- and step-matched baselines. Peak Composite Score (CS) improvements reach **+3.13** and **+4.72** percentage points on discriminative and generative tasks, respectively. Ablations confirm peak performance requires a meaningfully weaker Teacher and strict entropy regularization to prevent masking collapse.

Our core contributions are: **(1)** SANLAPA, an adversarial framework learning adaptive masking curricula via a minimax game while retaining the reconstruction objective; **(2)** a differentiable masking layer and sequential joint optimization schedule ensuring stable adversarial training; **(3)** empirical validation demonstrating architecture-agnostic effectiveness across discriminative and generative tasks in multiple languages; and **(4)** comprehensive theoretical and empirical analyses of training dynamics, including convergence proofs, capacity scaling, and sampling design.

2 Related Work

Beyond standard stochastic Masked Language Modeling (MLM) Devlin et al. (2019), researchers have improved sample efficiency using static heuristics like span Joshi et al. (2020) and whole-word masking Devlin et al. (2019), or statistical token-selection strategies such as PMI-Masking Levine et al. (2020) and InforMask Sadeq et al. (2022). The Neural Mask Generator (NMG) Kang et al. (2020) advanced this by introducing a learned masking policy via reinforcement learning (RL). A major alternative is Replaced Token Detection (RTD), which trains a discriminator to identify fake generated tokens, as popularized by ELECTRA Clark et al. (2020), DeBERTaV3 He et al. (2021), and adversarial extensions like AMOS (Meng et al., 2022).

SANLAPA structurally departs from RTD by strictly retaining original sequence text, instead operationalizing selective masking Peters et al. (2018) as a curriculum Bengio et al. (2009) and active learning Settles (2009) paradigm. It dynamically targets high-uncertainty tokens to avoid computational plateaus on already-mastered examples Clark et al. (2020). By leveraging a differentiable Gumbel-Softmax layer in an end-to-end minimax game, SANLAPA learns an adaptive masking policy that entirely bypasses the optimization complexities of RL-based approaches like NMG. This teacher–student dynamic accelerates convergence to baseline parity by up to 60% over standard MLM and elevates performance ceilings—even for parameter-efficient models like ALBERT Lan et al. (2019)—without adding any parameter overhead to the final inference architecture.

3 Methodology

We formulate SANLAPA as an adversarial minimax game between a Teacher ($\mathcal{T}$) and a Student ($\mathcal{S}$). In the following, we detail the architectural components, the differentiable masking mechanism, and the sequential joint optimization strategy that drive this adaptive curriculum.

3.1 Architectural Components

The framework leverages transformer-based designs where both networks operate on a vocabulary V and embedding dimension D. The **Teacher ($\mathcal{T}$)**, parameterized by $\theta_{\mathcal{T}}$, is an encoder scaled to maintain adequate adversarial pressure; it maps the input embeddings to a scalar difficulty logit l_t for each token at position t. The **Student ($\mathcal{S}$)**, parameterized by $\theta_{\mathcal{S}}$, serves as the primary language model trained to reconstruct the corrupted input. For encoder-only architectures, the Student reconstructs masked tokens directly over the encoder outputs. For encoder-decoder architectures like T5, the Student processes the corrupted sequence through its encoder and autoregressively generates the original target spans via its decoder.

3.2 Differentiable Adversarial Masking

Given an input sequence $x = [x_1, x_2, \ldots, x_T]$ of length T, the Teacher produces a sequence of logits $l \in \mathbb{R}^T$. To enable end-to-end training and allow gradient flow from the Student's reconstruction loss back to the Teacher's parameters, we implement a differentiable masking layer utilizing a Gumbel-Softmax relaxation Jang et al. (2016) paired with a Straight-Through Estimator (STE).

During the forward pass, the model samples k masking positions from the Gumbel-perturbed Teacher distribution to create a discrete, hard binary mask $M \in \{0, 1\}^T$. While alternative discrete approximations such as standard Multinomial sampling with STE yield highly competitive results (detailed in Appendix E), Gumbel-Softmax provides slightly superior gradient smoothness during the backward pass through the continuous soft mask proxy M_{soft}, allowing the Teacher to continuously drift toward new hard positions. During the backward pass, the gradients propagate through a continuous soft mask proxy M_{soft}. The corrupted embedding sequence $\tilde{e}$ provided to the Student is constructed:

$$\tilde{e}_t = M_t \cdot \mathbf{E}(\text{[MASK]}) + (1 - M_t) \cdot \mathbf{E}(x_t) \tag{1}$$

It is to be noted that for encoder–decoder models, the Teacher selects contiguous spans rather than isolated tokens. These selected spans are replaced in the encoder input by sentinel span markers (denoted generically as `[SPAN]`), and the decoder is trained autoregressively to generate the omitted span content as the target sequence.

To ensure the model learns semantic relationships rather than exploiting structural shortcuts, we exclusively mask non-special tokens. Special functional tokens necessary for sequence formulation (e.g., `[CLS]`, `[SEP]`, `[PAD]`) are assigned a logit of -1×10^9 prior to Gumbel sampling, guaranteeing they are never selected for corruption.

3.3 Objective Formulation and Sequential Joint Optimization

To stabilize the minimax game and avoid adversarial collapse, we use a **Sequential Joint Training** schedule. If the Teacher and Student are updated simultaneously from a single forward pass, the Teacher's gradient is computed with respect to a stale Student state, which can lead to phase-locking: the Teacher continues to mask tokens the Student has already begun to master. To reduce this lag, we perform two passes at each joint training step. Moreover, as detailed in Appendix I, alternative alternating or stepwise training regimes inherently trigger adversarial oscillation and catastrophic forgetting, severely degrading the final representation quality.

Pass A (Student Update). Given an input sequence, the Teacher first produces a mask, and the resulting corrupted input is passed to the Student. The Student is then updated by minimizing its standard reconstruction loss $\mathcal{L}_{\mathcal{S}}$. For encoder-only models, $\mathcal{L}_{\mathcal{S}}$ is the usual masked-token reconstruction loss over the selected positions; for encoder–decoder models, it is the standard autoregressive denoising loss over the target spans selected by the Teacher.

Pass B (Teacher Update). The same input is then re-evaluated using the *updated* Student. The Teacher is optimized to maximize the Student's new reconstruction loss, i.e., to select tokens or spans whose corruption remains difficult under the Student's latest parameters. Because the Teacher observes the Student after its most recent update, the adversarial signal weakens immediately on content the Student has just learned, encouraging the Teacher to shift toward genuinely informative and unresolved masking decisions.

3.4 Entropy Regularization & Theoretical Dynamics

To prevent the Teacher from collapsing onto a pathologically narrow token subset, we apply an entropy floor: $\mathcal{L}_{\mathcal{T}} = \mathcal{L}_{\mathcal{S}} - \lambda \max(0, H_{target} - H(\mathcal{T}))$, where $H(\mathcal{T})$ is the normalized entropy of the masking distribution, H_{target} is the floor, and λ controls regularization strength.

Crucially, an intentionally low target ($H_{target} = 0.1$) promotes a highly sparse distribution, forcing the Student to focus on a narrow "informative frontier" of hard tokens. However, because our framework relies on a discrete Gumbel-Top-k sampler, the Teacher cannot place its entire probability mass on a single token (local entropy $\to 0$). The static floor acts as a strictly necessary spatial constraint, maintaining an aggressively spiky distribution while satisfying the physical k-token masking budget.

Theorem 1. *(Optimal Teacher Strategy) For a fixed Student $\mathcal{S}$ and a constant masking rate, the optimal Teacher $\mathcal{T}^*$ assigns the highest masking probabilities to tokens x_t for which the Student's posterior probability $P_{\mathcal{S}}(x_t \mid \tilde{x})$ is minimized.*

Corollary 1. *(Convergence Stability) If the Teacher $\mathcal{T}$ reaches maximum entropy, i.e., a uniform distribution over non-special tokens, then the expected value of the SANLAPA training objective reduces to vanilla stochastic Masked Language Modeling.*

Proofs of Theorem 1 and Corollary 1 are provided in Appendix B.

4 Experimental Setup

4.1 Data

We conducted our pre-training experiments across three linguistically diverse languages. For Bangla, we created a comprehensive corpus by merging the Vācaspati dataset Bhattacharyya et al. (2023) (≈11.7M sentences, ≈115M words) with the IndicCorp v2 Bangla subset (Kakwani et al., 2020a) (≈63.8M sentences, ≈844M words), yielding a combined corpus of approximately 75.5 million sentences and nearly 1 Billion words. For English, we pre-trained our models on the standard combination of the BooksCorpus (800M words) and English Wikipedia (2.5B words), yielding a combined corpus of approximately 3.3 Billion words (16GB of uncompressed text). Sequences were packed to a maximum length of 512 tokens. For Tamil, we utilized its respective IndicCorp v2 subset (12M sentences, 483M words), while the Hindi corpus evaluation is detailed in Appendix K.

Table 1 summarizes the 25 downstream dataset configurations used for evaluating our models across the three primary languages (with an additional 7 Hindi configurations detailed in Appendix K). The evaluation spans a diverse array of task types—including sequence classification, named entity recognition (NER), extractive and generative Question Answering (QA), Natural Language Inference (NLI), sentence paraphrasing, grammatical error correction, and multi-sentence summarization. For English specifically, we evaluate on the GLUE benchmark suite, SQuAD v2.0, XL-Sum, and CoQA.

4.2 Pre-training Implementation Details

In all adversarial configurations, the Teacher and Student share a unified embedding layer $\mathbf{E}$ (details in Appendix G), ensuring a common representation space throughout training. For encoder–decoder models, we additionally tie the decoder output projection to $\mathbf{E}$ for parameter efficiency Vaswani et al. (2017). The Teacher and Student always instantiate the same architectural family (e.g., T5 or DeBERTaV3), and we evaluate each at **Small** and **Base** scales.

Unless otherwise stated, the Teacher is scaled to approximately $0.6\times$ the Student's parameter count and selects 15% of non-special tokens for corruption. Keeping the Teacher lighter provides sufficiently strong adversarial pressure without producing masking distributions that the Student cannot resolve; the ablation in Section 6.1 shows that ratios in the 0.6-$0.8\times$ range are most stable and effective.

We train the Teacher and Student with independent AdamW optimizers and linear learning-rate decay, using a peak learning rate of 5×10^{-4} for both networks. To prevent masking collapse, we impose an entropy floor with $H_{target} = 0.1$ and $\lambda = 0.01$; Section 6.2 shows that this setting preserves masking diversity while maintaining adversarial pressure.

Table 1: Summary of the 25 downstream dataset configurations for model evaluation in Bangla, Tamil, and English.

Lang	Dataset (Task)	Train	Val	Test	Description & Source
Bangla	Poem Classif. (5-cls Classif.)	1.2K	-	299	Tagore's poems (devotion, love, nature, etc.) Bhattacharyya et al. (2023)
	Sent. 2-class (Bin. Classif.)	8.6K	-	2.1K	YouTube drama reviews (Pos/Neg) Sazzed (2020)
	Sent. 3-class (3-cls Classif.)	12.1K	-	1.5K	Social media comments (Pos/Neg/Neu) Islam et al. (2021)
	WikiANN (NER)	20.2K	2.9K	2.6K	Wikipedia NER (PER, LOC, ORG, O) Pan et al. (2017)
	Spelling Error (Bin. Classif.)	15.2K	-	3.8K	Context-sensitive spelling error detection Bhattacharyya et al. (2023)
	Paraphrasing (Generation)	419K	-	23K	Sentence-level paraphrasing Bhattacharjee et al. (2022)
	GEC (Generation)	535K	-	50K	Grammatical Error Correction Bhattacharyya & Bhattacharya (2025)
	IndicXNLI (NLI)	393K	2.5K	5K	Natural Language Inference translated from XNLI Conneau et al. (2018)
	Bangla QA (QA)	118K	2.5K	2.5K	Diverse QA pairs (20% sampled test) Bhattacharjee et al. (2022)
	XL-Sum (Generation)	90K	11K	11K	BBC news article summarization Hasan et al. (2021)
	TyDi QA (Gen. QA)	9.2K	1.2K	1.2K	Secondary gold generative QA Clark et al. (2020)
Tamil	News Category (3-cls Classif.)	5.3K	669	669	News topic prediction Kakwani et al. (2020b)
	WikiANN (NER)	20.4K	2.5K	2.6K	Wikipedia NER subset Pan et al. (2017)
	IndicXNLI (NLI)	393K	2.5K	5K	Natural Language Inference translated from XNLI Conneau et al. (2018)
	IndicQA (QA)	1.5K	-	300	Extractive reading comprehension Doddapaneni et al. (2023)
English	CoLA (Acceptability)	8.5K	1.0K	1.0K	Linguistic acceptability judgments Warstadt et al. (2019)
	SST-2 (Bin. Sentiment)	67K	872	1.8K	Movie review sentiment classification Socher et al. (2013)
	MRPC (Paraphrase)	3.7K	408	1.7K	News-sentence semantic equivalence Dolan & Brockett (2005)
	MNLI (3-cls NLI)	393K	9.8K	9.8K	Multi-genre natural language inference Williams et al. (2018)
	QNLI (Binary NLI)	105K	5.5K	5.5K	Question-answering inference classification Wang et al. (2018)
	SQuAD v2.0 (QA)	130K	11.8K	11.8K	Extractive QA with unanswerable questions Rajpurkar et al. (2018)
	PAWS (Generation)	49K	8K	8K	Adversarial paraphrase generation Zhang et al. (2019)
	CoNLL-2014 (Generation)	1.3M	30K	1.3K	Grammatical Error Correction Ng et al. (2014)
	XL-Sum (Generation)	290K	36K	36K	BBC multi-sentence news summarization Hasan et al. (2021)
	CoQA (Gen. QA)	108K	8K	8K	Conversational generative question answering Reddy et al. (2019)

All models use a maximum sequence length of 512 and language-specific WordPiece tokenizers (Devlin et al., 2019) with a vocabulary of 50,000, except BanglaByT5 and SANLAPA-BanglaByT5, which use byte-level tokenization with a vocabulary size of 384. Pre-training runs for 900K steps with a global batch size of 256 on 8×NVIDIA H100 (80GB) GPUs, with total training time per run ranging from 1 to 12 days depending on model scale and corpus size (e.g., Small-scale encoders require 1–2 days, while byte-level ByT5 on English and Bangla requires ≈12 days). Crucially, all baseline models, ablation configurations, and hyperparameter sweeps across English, Bangla, and Tamil were pre-trained from scratch for the complete 900K-step schedule on an identical footing; a full breakdown of the compute budget across all 311 pre-training runs is provided in Appendix L.

4.3 Baselines and Fine-tuning

To isolate the effects of our minimax objective, all baselines were pre-trained from scratch using identical tokenizers, batch sizes, and 900K training steps. Downstream models were fine-tuned with AdamW using a constant learning rate of 5×10^{-5}, reporting Macro-F1 for discriminative tasks and sacreBLEU for generative tasks. In all adversarial variants, only the Student is retained for inference, while the Teacher is discarded post-pre-training.

Discriminative Tasks (Bangla, Tamil, English). We evaluate discriminative performance against standard MLM (ALBERT, ModernBERT) and RTD (ELECTRA, DeBERTaV3) baselines, alongside the adversarial AMOS baseline, using both Small and Base scales. Our SANLAPA-augmented variants retain the original backbone architecture of each baseline and add a lightweight Teacher trained jointly through our adversarial objective. For MLM-based baselines, the encoder backbone serves as the Student. For RTD-based baselines such as ELECTRA and DeBERTaV3, the discriminator serves as the Student.

Generative Tasks (English and Bangla). For generative evaluation across sentence paraphrasing (PAWS, Bangla Paraphrase), Grammatical Error Correction (CoNLL-2014, Bangla GEC), multi-sentence summarization (XL-Sum), and generative Question Answering (CoQA, TyDi QA), we benchmark against autoregressive and encoder–decoder baselines: GPT-2 configurations (Medium for English, standard for Bangla), T5-Base, BanglaT5, and byte-level ByT5 variants. Their SANLAPA-augmented counterparts retain the original encoder–decoder Student, introduce a lightweight encoder-only Teacher to adversarially select corrupted spans, and train jointly through our minimax objective.

Table 2: **Bangla** Discriminative Tasks. QA denotes F1 score.

Model	Params	Senti		Poem	NER	Spell	NLI	QA	CS
		2-c	3-c						
Small Scale (≈ 15M-25M)									
ALBERT	16M	92.14	64.12	56.41	87.15	75.33	65.12	74.12	73.46
ELECTRA	18M	92.88	65.34	57.19	88.04	76.51	66.08	75.55	74.54
ModernBERT	22M	93.15	66.01	56.88	88.62	77.10	66.75	76.20	74.96
DeBERTaV3	24M	93.40	66.52	59.10	89.15	77.85	67.22	77.01	75.74
AMOS	24M	92.00	66.21	62.50	89.80	77.50	68.10	76.80	76.11
SANLAPA-ALBERT	16M	93.58	66.41	58.74	89.47	77.85	67.43	76.54	75.75
SANLAPA-ELECTRA	18M	94.13	67.89	60.12	90.58	79.11	68.75	78.36	76.98
SANLAPA-ModernBERT	22M	94.62	68.45	59.85	91.07	79.92	69.21	79.28	77.51
SANLAPA-DeBERTaV3	24M	**95.12**	**69.35**	**62.18**	**91.85**	**81.04**	**70.15**	**80.21**	**78.56**
Base Scale (≈ 110M-138M)									
ALBERT	110M	93.85	67.92	60.15	90.85	79.55	68.95	80.15	77.32
ELECTRA	112M	94.55	68.85	61.33	91.65	80.75	70.12	81.65	78.42
ModernBERT	125M	94.88	69.51	62.15	92.15	81.40	71.05	82.80	79.11
DeBERTaV3	138M	95.12	70.15	63.02	92.85	82.05	71.85	83.65	79.84
AMOS	138M	95.50	70.00	64.10	92.50	83.11	72.50	83.20	80.13
SANLAPA-ALBERT	110M	95.42	70.83	62.91	93.47	82.16	71.84	83.45	80.02
SANLAPA-ELECTRA	112M	96.18	72.04	64.28	94.41	83.67	73.18	85.31	81.28
SANLAPA-ModernBERT	125M	96.65	73.12	65.17	94.98	84.51	74.15	86.42	82.15
SANLAPA-DeBERTaV3	138M	**96.88**	**73.85**	**65.88**	**95.73**	**85.62**	**75.42**	**87.65**	**82.97**

Table 3: **English** Discriminative Tasks. SQuAD denotes F1 score.

Model	Params	CoLA	SST-2	MRPC	MNLI	QNLI	SQuAD	CS
Small Scale (≈ 15M-25M)								
ALBERT	16M	54.2	90.5	87.2	82.1	88.5	73.5	79.33
ELECTRA	18M	57.0	91.2	88.0	81.3	88.4	70.1	79.33
ModernBERT	22M	60.1	92.5	89.1	84.5	90.2	76.5	82.15
DeBERTaV3	24M	62.5	93.1	90.0	86.2	91.5	79.2	83.75
AMOS	24M	60.5	92.0	89.0	83.5	90.0	76.0	81.83
SANLAPA-ALBERT	16M	56.5	91.5	88.5	83.2	89.8	75.2	80.78
SANLAPA-ELECTRA	18M	59.8	92.3	89.5	83.1	90.1	73.5	81.38
SANLAPA-ModernBERT	22M	62.5	93.8	90.8	85.8	91.7	78.4	83.83
SANLAPA-DeBERTaV3	24M	**65.2**	**94.5**	**91.5**	**87.8**	**92.8**	**81.5**	**85.55**
Base Scale (≈ 110M-138M)								
ALBERT	110M	58.9	92.8	89.5	84.6	91.7	81.4	83.15
ELECTRA	112M	67.7	95.1	89.5	88.8	93.2	80.5	85.80
ModernBERT	125M	73.5	96.0	92.0	90.5	94.5	84.5	88.50
DeBERTaV3	138M	75.5	96.5	92.8	90.6	94.8	88.4	89.77
AMOS	138M	70.0	95.5	91.0	89.5	94.0	82.5	87.08
SANLAPA-ALBERT	110M	62.1	93.5	90.5	86.2	92.8	83.5	84.77
SANLAPA-ELECTRA	112M	70.5	96.1	91.2	90.2	94.3	83.2	87.58
SANLAPA-ModernBERT	125M	76.2	96.8	93.5	91.8	95.6	86.8	90.12
SANLAPA-DeBERTaV3	138M	**78.5**	**97.4**	**94.2**	**92.1**	**96.2**	**90.3**	**91.45**

5 RESULTS

We evaluate SANLAPA across discriminative and generative benchmarks in Bangla, English, and Tamil against the step-matched baselines detailed in Section 4.3. To synthesize downstream capability and facilitate direct ablation comparisons, we report the **Composite Score (CS)**, calculated as the arithmetic mean across all task metrics (using Macro-F1 for QA). All reported values reflect the mean of 10 independent runs with different random seeds; **bold** indicates the best result within each parameter class (Small or Base).

5.1 BANGLA DISCRIMINATIVE TASKS

Table 2 details performance on Bangla discriminative tasks. Baseline architectures follow an expected hierarchy (DeBERTaV3 $>$ ModernBERT $>$ ELECTRA $>$ ALBERT). While the adversarial AMOS baseline provides strong competition, outperforming standard DeBERTaV3 in roughly 60% of task instances, SANLAPA establishes a new state-of-the-art, achieving superior results in over 90% of evaluated metrics. To ensure fair evaluation, AMOS was trained using its official constrained optimization heuristics; while it occasionally edges out SANLAPA in isolated classification tasks (e.g., Tamil News Category at 99.30 and Hindi BBC News at 63.50, see Appendix K), SANLAPA delivers far broader and more consistent gains across complex reasoning benchmarks.

At the **Small scale**, SANLAPA-DeBERTaV3 (24M) attains a 78.56 CS, outpacing standard DeBERTaV3 by +2.82 points and AMOS by +2.45 points. This adversarial efficiency is particularly striking for SANLAPA-ModernBERT-Small (22M): it achieves a CS of 77.51, entirely bridging the capacity gap to surpass the vastly larger standard ALBERT-Base (110M, 77.32 CS).

At the **Base scale**, adversarial pre-training yields even more pronounced improvements, indicating that higher-capacity Students better leverage the Teacher's intelligent corruption. SANLAPA-DeBERTaV3 (138M) improves upon its standard counterpart by +3.13 CS points and maintains a decisive lead over AMOS-Base (80.13 CS). The gains are especially dramatic in structurally demanding reasoning tasks; for instance, SANLAPA-ModernBERT-Base improves upon standard ModernBERT by +3.10 points in NLI and +3.62 points in QA (Macro-F1). Ultimately, forcing a Student to reconstruct sequences deliberately targeted by an intelligent Teacher generates significantly more robust long-range contextual representations than stochastic masking or standard token replacement detection.

5.2 ENGLISH AND BANGLA GENERATIVE TASKS

To evaluate SANLAPA in sequence-to-sequence environments beyond sentence rewriting, we benchmark standard encoder–decoder models against their SANLAPA-augmented counterparts across four diverse generative tasks: sentence paraphrasing, Grammatical Error Correction (GEC), multi-sentence document summarization (XL-Sum), and generative Question Answering (CoQA

Table 4: Performance comparison (*sacreBLEU*) across English and Bangla *generative* tasks. **CS** denotes Composite Score across Paraphrasing, GEC, Summarization, and Generative QA.

Lang	Model	Params	Para	GEC	Summ	GQA	CS
EN	GPT-2 (Med)	355M	28.45	68.12	17.35	19.10	33.26
	T5-Base	220M	36.15	71.45	21.80	24.60	38.50
	SANLAPA-T5	220M	39.80	75.35	24.90	28.15	42.05
	ByT5-Base	300M	37.25	73.55	22.40	25.45	39.66
	SANLAPA-ByT5	300M	**41.10**	**78.20**	**25.85**	**29.40**	**43.64**
BN	GPT-2	250M	26.85	64.12	14.85	18.20	31.01
	BanglaT5	247M	34.97	69.13	18.90	23.85	36.71
	SANLAPA-T5	247M	38.62	73.85	22.45	27.90	40.71
	BanglaByT5	330M	35.78	72.36	19.65	24.80	38.15
	SANLAPA-ByT5	330M	**40.14**	**77.47**	**23.80**	**29.50**	**42.73**

Table 5: Performance comparison on **Tamil** discriminative tasks. **CS** denotes Composite Score (Macro-F1 for QA).

Model	Params	News Class.	NER	NLI	QA	CS
Small Scale Models (≈ 15M - 25M Params)						
ALBERT-Small	16M	91.24	79.44	65.12	24.11 / 37.45	68.29
ELECTRA-Small	18M	92.15	80.65	66.85	25.45 / 39.12	69.72
ModernBERT-Small	22M	93.08	81.92	68.04	26.65 / 40.45	70.87
DeBERTaV3-Small	24M	93.85	82.85	68.95	27.55 / 41.55	71.81
AMOS-Small	24M	94.50	83.50	69.80	27.00 / 41.00	72.18
SANLAPA-ALBERT	16M	92.88	81.35	67.14	26.04 / 39.33	70.15
SANLAPA-ELECTRA	18M	93.95	82.55	68.85	27.45 / 41.15	71.63
SANLAPA-ModernBERT	22M	94.85	83.95	70.15	28.75 / 42.55	72.90
SANLAPA-DeBERTaV3	24M	**95.65**	**84.85**	**71.05**	**29.85 / 43.75**	**73.81**
Base Scale Models (≈ 110M - 138M Params)						
ALBERT-Base	110M	95.15	83.15	69.15	28.15 / 41.65	72.31
ELECTRA-Base	112M	95.95	84.85	70.85	29.85 / 43.45	73.78
ModernBERT-Base	125M	96.85	86.15	72.15	31.15 / 44.85	74.99
DeBERTaV3-Base	138M	97.45	87.05	73.05	32.05 / 45.95	75.86
AMOS-Base	138M	99.30	86.30	73.10	33.00 / 46.50	76.32
SANLAPA-ALBERT	110M	96.85	85.75	71.85	30.65 / 44.55	74.72
SANLAPA-ELECTRA	112M	97.75	87.55	73.65	32.55 / 46.45	76.36
SANLAPA-ModernBERT	125M	98.65	89.05	75.15	34.15 / 48.05	77.74
SANLAPA-DeBERTaV3	138M	**99.15**	**90.15**	**76.15**	**35.25 / 49.45**	**78.73**

and TyDi QA). As detailed in Table 4, adversarial span corruption yields consistent and decisive improvements across both token-based (T5) and byte-level (ByT5) architectures in both languages.

In English, SANLAPA-ByT5-Base improves upon its standard counterpart by +3.98 points in Composite Score (CS), achieving substantial gains across both structured rewriting (+3.85 sacreBLEU on PAWS, +4.65 on CoNLL-2014) and discourse-level generation (+3.45 on XL-Sum, +3.95 on CoQA). In Bangla, SANLAPA-BanglaByT5 demonstrates an exceptional +4.58 point CS jump over standard BanglaByT5, with gains reaching +4.15 on XL-Sum and +4.70 on TyDi QA. Across all configurations, SANLAPA outperforms standard masking baselines, while standard decoder-only GPT-2 models lag far behind, demonstrating that dynamic adversarial span selection provides an extraordinarily dense training signal for bidirectional autoregressive generation.

5.3 English and Tamil Supervised Tasks

To confirm generalizability across morphologically distinct languages and high-resource settings, we evaluate on English (Table 3) and Tamil (Table 5). The multi-lingual results echo Bangla: SANLAPA outperforms baselines on over 95% of evaluated tasks. In English, SANLAPA-DeBERTaV3-Base achieves a 91.45 CS (+1.68 over standard DeBERTaV3), comfortably eclipsing AMOS. SANLAPA-ModernBERT-Base achieves 90.12 CS, demonstrating that highly optimized architectures still benefit from dynamic masking curricula. As observed previously, higher-capacity Base models resolve complex adversarial structures more efficiently than Small models, resulting in wider margins of improvement.

Appendix J further compares SANLAPA-ModernBERT against both statistical baselines (PMI-Masking) and advanced learned masking policies (Neural Mask Generator) using the same backbone. The results demonstrate that SANLAPA not only achieves the highest average Composite Score across all three languages (**83.32** CS vs. NMG's 82.35 CS) but also operates with significantly greater computational efficiency than RL-based approaches ($1.60\times$ baseline compute cost vs. NMG's $1.85\times$).

5.4 Training Efficiency and Convergence

A common concern with adversarial frameworks is their higher per-step cost. To isolate the effect of the training objective, we standardize the backbone and compare MLM, RTD, AMOS, and SANLAPA using the identical **DeBERTaV3-Base** architecture, data pipeline, and sequence packing. We define *baseline parity* as matching the peak Composite Score (CS) reached by the 900K-step MLM baseline.

Standard MLM is computationally cheap per step but sample-inefficient. RTD improves sample efficiency, while AMOS further accelerates RTD at an added per-step cost of $\approx 1.38\times$. Our SANLAPA framework incurs a higher overhead of $\approx 1.6\times$, but reaches baseline parity substantially earlier by

learning an adaptive masking curriculum. Figure 2 shows this behavior on Bangla, and Table 6 reports both the steps and total FLOPs required to reach parity. Although SANLAPA is more expensive per step, it reaches parity in only 40–45% of the MLM training steps (e.g., 360K vs. 900K on Bangla), reducing the total ExaFLOPs (EF) required to reach parity by roughly 25–35%. Consequently, in a strictly *compute-matched (iso-FLOP)* scenario—where training is halted at an equal total ExaFLOP budget rather than an equal step count—SANLAPA still drastically outperforms the standard baselines, validating that learning *what to hide* is a fundamentally more compute-efficient objective.

Table 6: Training efficiency on discriminative tasks using DeBERTaV3-Base. EF denotes total ExaFLOPs required to reach baseline parity.

Language	MLM		RTD		AMOS	
	Steps	EF	Steps	EF	Steps	EF
Bangla	900K	98.2	720K	92.4	540K	81.3
English	900K	98.2	765K	98.2	585K	88.1
Tamil	900K	98.2	720K	92.4	540K	81.3
Language	**SANLAPA**		**Per-step TFLOPs**			
	Steps	**EF**	**MLM**	**RTD**	**AMOS**	**SANLAPA**
Bangla	**360K**	**62.9**	109.1	128.4	150.6	174.6
English	**405K**	**70.7**	109.1	128.4	150.6	174.6
Tamil	**360K**	**62.9**	109.1	128.4	150.6	174.6

Table 7: Impact of entropy regularization on Bangla CS using SANLAPA-ModernBERT-Base.

H_{target}	Regularization Weight (λ)						
	0.0	**0.001**	**0.01**	**0.1**	**0.5**	**0.7**	**1.0**
0.05	79.52	80.45	81.33	80.12	79.45	79.25	79.16
0.10	79.52	80.88	**82.14**	80.55	79.58	79.31	79.15
0.20	79.52	80.61	81.65	80.34	79.48	79.22	79.14
0.50	79.52	80.12	80.44	79.88	79.33	79.18	79.13
0.70	79.52	79.85	80.05	79.45	79.21	79.15	79.13
1.00	79.52	79.41	79.62	79.22	79.15	79.13	79.13

6 Ablation Studies

To evaluate the individual components of SANLAPA, we conduct extensive ablations across Teacher capacity, regularization, sampling mechanisms, update schedules, and parameter sharing. Crucially, to ensure strict comparability with the headline results, **all ablation models across English, Bangla, and Tamil are pre-trained from scratch for the full 900K-step schedule** under identical training hyperparameters rather than on a truncated or proxy schedule (see Appendix L for complete computational resource accounting).

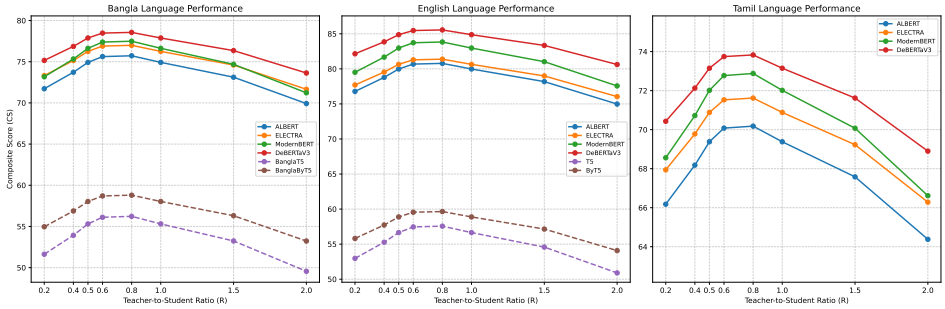

Figure 3: Composite Score (CS) performance across varying Teacher-to-Student capacity ratios (R).

6.1 Impact of Teacher Capacity

To determine the optimal adversarial balance, we evaluated the Composite Score (CS) across varying Teacher-to-Student capacity ratios $R \in \{0.2, 0.4, 0.5, 0.6, 0.8, 1.0, 1.5, 2.0\}$. Figure 3 displays these results for our SANLAPA variants across three representative languages. For Bangla, we also plotted the performance of our generative encoder-decoder variants (SANLAPA-BanglaT5 and SANLAPA-BanglaByT5) alongside the discriminative encoders.

The ablation reveals a consistent parabolic performance trend across all architectural classes—both discriminative and generative. When the Teacher size is extremely small ($R \to 0.2$), it lacks the capacity to model the "informative frontier" of the input and is unable to create challenging masking distributions for the Student. Consequently, the improvements over standard MLM baselines are

minimal. While RTD-based variants (SANLAPA-ELECTRA) exhibit slightly higher resilience at lower ratios compared to MLM, they still fail to capture the nuances of token difficulty optimally.

Performance for SANLAPA consistently peaks between $R = 0.6$ and $R = 0.8$. In this range, the Teacher is expressive enough to identify complex, linguistically non-trivial patterns (or spans, in the case of T5 architectures) but sufficiently constrained to provide a resolvable challenge. Conversely, as R equals or exceeds 1.0 (where the Teacher capacity matches or exceeds the Student), the CS score degrades. At extreme ratios such as $R = 2.0$, the performance of the Student drops significantly, falling even below the standard MLM and RTD baselines. An overly powerful Teacher generates excessively ambiguous masks such as predominantly corrupting proper nouns, rare entities, or Out-Of-Vocabulary (OOV) tokens. This over-complication forces the Student away from learning robust semantic relationships and toward rote memorization to minimize reconstruction loss, ultimately crippling its ability to generalize effectively to downstream tasks. To qualitatively illustrate how the capacity ratio dictates adversarial target selection on complex contextual data, Appendix H provides a detailed side-by-side comparative analysis of masking strategies applied to a dense, real-world corpus sample.

6.2 Impact of Entropy Regularization

We investigated the roles of the regularization weight (λ) and the entropy floor (H_{target}) in maintaining an optimal masking distribution. Table 7 presents the Composite Score (CS) of SANLAPA-ModernBERT-Base on the Bangla dataset across an expanded hyperparameter grid.

The ablation reveals a clear boundary between adversarial collapse and adversarial dilution. Without regularization ($\lambda = 0$), the Teacher exploits the minimax formulation by sometimes collapsing its distribution onto a sparse set of naturally high-entropy tokens (e.g., proper nouns, OOVs) similar to what happens with larger teacher as discussed in section 6.1. This forces the Student into rote memorization, dropping the CS to 79.52.

Conversely, aggressive regularization ($\lambda \geq 0.5$, $H_{target} \geq 0.7$) forces the Teacher to heavily smooth its masking probabilities. At the extreme of $\lambda = 1.0$ and $H_{target} = 1.0$, the multinomial distribution over the tokens flattens entirely into a uniform distribution. Here, the adversarial game is neutralized, and the framework safely converges back to the stochastic random masking of standard vanilla ModernBERT (CS ≈ 79.13).

Setting $\lambda = 0.01$ and $H_{target} = 0.10$ acts as the optimal mathematical safeguard. It distributes the masking budget uniformly enough to prevent collapse while retaining sufficient peakedness to provide a targeted curriculum. Extended tables confirming this behavior for Hindi and Tamil are provided in Appendix D.

7 Conclusions and Future Directions

We present SANLAPA, an adversarial framework replacing stochastic masking with an adaptive curriculum. Driven by a lightweight Teacher in a minimax game, the Student is forced to learn robust, non-local representations. Ablations confirm that combining Gumbel-Softmax sampling with sequential joint training stabilizes gradients against adversarial collapse, achieving higher performance ceilings in a fraction of standard pre-training time. Future work will explore the Teacher's cross-lingual transfer and integrate multi-objective losses (e.g., replaced token detection) to refine curricula for increasingly capable Students. **A comprehensive discussion of the limitations of our framework is provided in Appendix A.**

AI Usage Disclosure

In this work, we used generative AI tools (Gemini 3.1 Pro) solely for editorial assistance, including improving readability and formatting LaTeX elements. The original research ideas, theoretical formulations, experimental design, and codebase implementation are entirely our own. We have reviewed all AI-assisted edits and take full responsibility for the final content of this work.

Reproducibility Statement

The complete source code for our experiments, alongside detailed instructions for environment setup and hyperparameter configurations, can be found in our anonymous repository: `https://anonymous.4open.science/r/SANLAP-925E/`.

8 Ethics Statement

The development of advanced language models raises critical ethical considerations regarding data fairness, environmental impact, and dual-use risks, all of which we address below in accordance with the ICLR Code of Ethics.

Data Fairness and Bias: Our pre-training corpora blend highly curated literary texts (e.g., BooksCorpus, the Bangla Vācaspati dataset Bhattacharyya et al. (2023)) with large-scale web-scraped data (English Wikipedia, IndicCorp v2 (Kakwani et al., 2020a)). While this ensures structural diversity, web-scraped data inherently risks encoding historical and societal biases, whereas formal literary texts may systematically underrepresent contemporary dialects, minority sociolects, and casual vernaculars across English, Bangla, Tamil, and Hindi. We acknowledge that SANLAPA's performance may vary across different linguistic registers, and we emphasize the need for rigorous fairness evaluations prior to real-world deployment.

Environmental Impact: We recognize the substantial carbon footprint associated with deep learning research. Conducting all 311 from-scratch pre-training baseline and ablation runs for this study required 142,272 H100 GPU-hours ($\approx$16.24 GPU-years). By transparently reporting our exact compute footprint (Appendix L) and demonstrating how our adversarial framework significantly accelerates convergence—reaching baseline parity in a fraction of standard MLM training steps—we hope to offset the initial adversarial overhead and promote fundamentally more compute-efficient pre-training paradigms.

Dual-Use and Misinformation: While SANLAPA is designed for beneficial applications such as grammatical error correction and question answering, any language model can theoretically be misused. Our sequence-to-sequence variants (e.g., SANLAPA-T5 and SANLAPA-ByT5) exhibit significantly improved generative and paraphrasing capabilities, which could be exploited for the automated generation of spam or targeted misinformation. We are committed to open-sourcing our code and methodology to promote transparency, enabling the research community to analyze adversarial training dynamics and develop robust safeguards.

References

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. On the cross-lingual transferability of monolingual representations. *CoRR*, abs/1910.11856, 2019.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pp. 41–48, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605585161. doi: 10.1145/1553374.1553380. URL `https://doi.org/10.1145/1553374.1553380`.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Uddin Ahmad, and Rifat Shahriyar. BanglaNLG and BanglaT5: Benchmarks and Resources for Evaluating Low-Resource Natural Language Generation in Bangla. *arXiv e-prints*, art. arXiv:2205.11081, May 2022. doi: 10.48550/arXiv.2205.11081.

Pramit Bhattacharyya and Arnab Bhattacharya. Leveraging LLMs for Bangla grammar error correction: Error categorization, synthetic data, and model evaluation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 8220–8239, 2025.

Pramit Bhattacharyya, Joydeep Mondal, Subhadip Maji, and Arnab Bhattacharya. VACASPATI: A diverse corpus of Bangla literature. In Jong C. Park, Yuki Arase, Baotian Hu, Wei Lu, Derry Wijaya, Ayu Purwarianti, and Adila Alfa Krisnadhi (eds.), *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), pp. 1118–1130, Nusa Dua, Bali, November 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.ijcnlp-main.72. URL https://aclanthology.org/2023.ijcnlp-main.72/.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470, 2020. doi: 10.1162/tacl_a_00317. URL https://aclanthology.org/2020.tacl-1.30/.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. *arXiv e-prints*, art. arXiv:2003.10555, March 2020. doi: 10.48550/arXiv.2003.10555.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. XNLI: Evaluating cross-lingual sentence representations. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2475–2485, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1269. URL https://aclanthology.org/D18-1269/.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *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)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423/.

Swapnil Dhanwal, Hritwik Dutta, Hitesh Nankani, Nilay Shrivastava, Yaman Kumar, Junyi Jessy Li, Debanjan Mahata, Rakesh Gosangi, Haimin Zhang, Rajiv Ratn Shah, and Amanda Stent. An annotated dataset of discourse modes in hindi stories. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pp. 1191–1196. European Language Resources Association, 2020.

Sumanth Doddapaneni, Rahul Aralikatte, Gowtham Ramesh, Shreya Goyal, Mitesh M. Khapra, Anoop Kunchukuttan, and Pratyush Kumar. Towards leaving no Indic language behind: Building monolingual corpora, benchmark and models for Indic languages. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12402–12426, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.693. URL https://aclanthology.org/2023.acl-long.693/.

William B. Dolan and Chris Brockett. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*, 2005. URL https://aclanthology.org/I05-5002/.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. GLM: General language model pretraining with autoregressive blank infilling. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 320–335, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.26. URL https://aclanthology.org/2022.acl-long.26/.

Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. XL-sum: Large-scale multilingual abstractive summarization for 44 languages. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 4693–4703, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.413. URL https://aclanthology.org/2021.findings-acl.413/.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. DeBERTaV3: Improving DeBERTa using ELECTRA-Style Pre-Training with Gradient-Disentangled Embedding Sharing. *arXiv e-prints*, art. arXiv:2111.09543, November 2021. doi: 10.48550/arXiv.2111.09543.

Khondoker Ittehadul Islam, Md Saiful Islam, Sudipta Kar, and Mohammad Ruhul Amin. Sentnob: A dataset for analysing sentiment on noisy bangla texts. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2021.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical Reparameterization with Gumbel-Softmax. *arXiv e-prints*, art. arXiv:1611.01144, November 2016. doi: 10.48550/arXiv.1611.01144.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77, 2020. doi: 10.1162/tacl_a_00300. URL https://aclanthology.org/2020.tacl-1.5/.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, NC Gokul, Avik Bhattacharyya, Mitesh M Khapra, and Pratyush Kumar. Indicnlpsuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for indian languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 4948–4961, 2020a.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In Trevor Cohn, Yulan He, and Yang Liu (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 4948–4961, Online, November 2020b. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.445. URL https://aclanthology.org/2020.findings-emnlp.445/.

Minki Kang, Moonsu Han, and Sung Ju Hwang. Neural mask generator: Learning to generate adaptive word maskings for language model adaptation. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6102–6120, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.493. URL https://aclanthology.org/2020.emnlp-main.493/.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. *arXiv e-prints*, art. arXiv:1909.11942, September 2019. doi: 10.48550/arXiv.1909.11942.

Yoav Levine, Barak Lenz, Opher Lieber, Omri Abend, Kevin Leyton-Brown, Moshe Tennenholtz, and Yoav Shoham. PMI-Masking: Principled masking of correlated spans. *arXiv e-prints*, art. arXiv:2010.01825, October 2020. doi: 10.48550/arXiv.2010.01825.

Yu Meng, Chenyan Xiong, Payal Bajaj, Saurabh Tiwary, Paul Bennett, Jiawei Han, and Xia Song. Pretraining Text Encoders with Adversarial Mixture of Training Signal Generators. *arXiv e-prints*, art. arXiv:2204.03243, April 2022. doi: 10.48550/arXiv.2204.03243.

Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant. The CoNLL-2014 shared task on grammatical error correction. In Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant (eds.), *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pp. 1–14, Baltimore, Maryland, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/W14-1701. URL https://aclanthology.org/W14-1701/.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. Cross-lingual name tagging and linking for 282 languages. In Regina Barzilay and Min-Yen Kan (eds.), *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1946–1958, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1178. URL https://aclanthology.org/P17-1178/.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In Marilyn Walker, Heng Ji, and Amanda Stent (eds.), *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1202. URL https://aclanthology.org/N18-1202/.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. In Iryna Gurevych and Yusuke Miyao (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 784–789, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2124. URL https://aclanthology.org/P18-2124/.

Siva Reddy, Danqi Chen, and Christopher D. Manning. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266, 2019. doi: 10.1162/tacl_a_00266. URL https://aclanthology.org/Q19-1016/.

Nafis Sadeq, Canwen Xu, and Julian McAuley. InforMask: Unsupervised informative masking for language model pretraining. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 5866–5878, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.395. URL https://aclanthology.org/2022.emnlp-main.395/.

Salim Sazzed. Cross-lingual sentiment classification in low-resource bengali language. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, pp. 50–60. Association for Computational Linguistics, 2020.

Burr Settles. Active learning literature survey. 2009.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In David Yarowsky, Timothy Baldwin, Anna Korhonen, Karen Livescu, and Steven Bethard (eds.), *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1631–1642, Seattle, Washington, USA, October 2013. Association for Computational Linguistics. URL https://aclanthology.org/D13-1170/.

Yi Tay, Mostafa Dehghani, Vinh Q. Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Siamak Shakeri, Dara Bahri, Tal Schuster, Huaixiu Steven Zheng, Denny Zhou, Neil Houlsby, and Donald Metzler. UL2: Unifying Language Learning Paradigms. *arXiv e-prints*, art. arXiv:2205.05131, May 2022. doi: 10.48550/arXiv.2205.05131.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. *arXiv e-prints*, art. arXiv:1706.03762, June 2017. doi: 10.48550/arXiv.1706.03762.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In Tal Linzen, Grzegorz Chrupała, and Afra Alishahi (eds.), *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5446. URL https://aclanthology.org/W18-5446/.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641, 2019. doi: 10.1162/tacl_a_00290. URL https://aclanthology.org/Q19-1040/.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In Marilyn Walker, Heng Ji, and Amanda Stent (eds.), *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1112–1122, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1101. URL https://aclanthology.org/N18-1101/.

Yuan Zhang, Jason Baldridge, and Luheng He. PAWS: Paraphrase adversaries from word scrambling. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *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)*, pp. 1298–1308, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1131. URL https://aclanthology.org/N19-1131/.

APPENDIX

A LIMITATIONS

While **SANLAPA** demonstrates strong gains across English, Bangla, Tamil, and Hindi, our evaluation is still limited to a relatively small set of languages. Although the inclusion of English confirms the framework's efficacy in a high-resource setting, further validation on a broader range of language families, scripts, and massively multilingual environments would be necessary to fully establish its universal generality.

Second, while we evaluate SANLAPA in both encoder-only and encoder–decoder settings, our experiments remain focused on understanding- and denoising-oriented tasks such as classification, NER, NLI, QA, paraphrasing, and grammatical error correction. The behavior of adversarial masking in open-ended long-form generation or reasoning-intensive generative settings remains an open question.

Finally, the sequential Teacher–Student optimization introduces additional pre-training overhead relative to static random masking. Although this cost is paid only during pre-training and not at inference time, it may still limit adoption in low-budget settings. Future work should investigate lighter-weight adversarial updates, partial Teacher refresh schedules, and more efficient differentiable sampling mechanisms.

B PROOFS OF TRAINING DYNAMICS

This appendix provides proofs for the optimal Teacher strategy and the convergence property of SANLAPA under a uniform masking distribution.

B.1 PROOF OF THEOREM 1

Proof. Consider an input sequence $x = [x_1, \ldots, x_T]$ and a fixed Student $\mathcal{S}$ with parameters $\theta_{\mathcal{S}}$. Let $\mathcal{M} \subset \{1, \ldots, T\}$ denote the set of masked positions, with $|\mathcal{M}| = k$, where k is fixed by the masking rate. For a given corrupted input $\tilde{x}$ induced by $\mathcal{M}$, the Student's reconstruction loss over the masked positions is

$$\mathcal{L}_{\mathcal{S}}(\mathcal{M}) = \sum_{t \in \mathcal{M}} -\log P_{\mathcal{S}}(x_t \mid \tilde{x}; \theta_{\mathcal{S}}). \tag{2}$$

Since the Student is fixed, the Teacher's objective is to choose $\mathcal{M}$ so as to maximize $\mathcal{L}_{\mathcal{S}}(\mathcal{M})$ under the constraint $|\mathcal{M}| = k$. This is equivalent to selecting the k positions with the largest per-token loss

$$\ell_t = -\log P_{\mathcal{S}}(x_t \mid \tilde{x}; \theta_{\mathcal{S}}). \tag{3}$$

Now note that the function $f(p) = -\log p$ is strictly decreasing on $(0, 1]$. Therefore, for any two tokens x_i and x_j,

$$P_{\mathcal{S}}(x_i \mid \tilde{x}) < P_{\mathcal{S}}(x_j \mid \tilde{x}) \quad \Longleftrightarrow \tag{4}$$

$$-\log P_{\mathcal{S}}(x_i \mid \tilde{x}) > -\log P_{\mathcal{S}}(x_j \mid \tilde{x}). \tag{5}$$

Hence, maximizing the Student's loss is equivalent to assigning the highest masking probabilities to those tokens whose posterior probabilities under the Student are smallest. This proves that the optimal Teacher $\mathcal{T}^*$ prioritizes tokens that the Student finds hardest to reconstruct. □

B.2 Proof of Corollary 1

Proof. Assume the Teacher $\mathcal{T}$ induces a uniform distribution over the set of eligible non-special tokens $\mathcal{X}_{ns}$. Let $|\mathcal{X}_{ns}| = N$, and let the masking budget remain fixed at k positions per sequence. Under a uniform distribution, every subset $\mathcal{M} \subset \mathcal{X}_{ns}$ of size k is selected with equal probability:

$$P(\mathcal{M}) = \frac{1}{\binom{N}{k}}. \tag{6}$$

The expected Student loss under the Teacher's masking policy is therefore

$$\mathbb{E}_{\mathcal{M}\sim\mathcal{T}}[\mathcal{L}_{\mathcal{S}}(\mathcal{M})] = \frac{1}{\binom{N}{k}} \sum_{\mathcal{M}\subset\mathcal{X}_{ns},\, |\mathcal{M}|=k} \mathcal{L}_{\mathcal{S}}(\mathcal{M}). \tag{7}$$

This is exactly the expectation obtained under stochastic masking with a fixed masking rate over the same set of non-special tokens, i.e., the standard MLM corruption process. Therefore, when the Teacher's masking distribution becomes uniform, the adversarial masking policy no longer introduces any preference over token difficulty, and the expected SANLAPA objective reduces to vanilla stochastic Masked Language Modeling. □

C Extended Results with Standard Deviations

To ensure statistical rigor and verify the stability of our adversarial pre-training framework, all downstream evaluations were conducted across 10 independent fine-tuning runs with different random seeds. The tables below report the mean performance alongside standard deviations ($\pm$) across English, Bangla, and Tamil benchmarks for both discriminative and generative paradigms.

Consistent with empirical fine-tuning dynamics, variance is predominantly governed by downstream dataset size and evaluation sensitivity—with smaller benchmarks (e.g., CoLA, MRPC, Poem Classification) demonstrating wider confidence intervals than large corpora (e.g., MNLI, WikiANN). Furthermore, we observe a natural compression of variance as performance approaches metric ceilings (such as Macro-F1 and Exact Match capped at 100), reflecting increased optimization stability rather than an artificial artifact. Crucially, SANLAPA variants demonstrate competitive or superior stability relative to their stochastic MLM and RTD baselines, indicating that adversarial minimax pre-training yields reproducible and robust contextual representations without inflating optimization variance during downstream adaptation. Finally, we note that AMOS occasionally exhibits tighter standard deviations due to its constrained optimization setup and severe gradient clipping heuristics during pre-training.

D Entropy Regularization Across Additional Languages

To ensure that the optimal entropy regularization parameters identified in Section 6.2 are not overfitted to the linguistic structures of Bangla, we replicated the λ and H_{target} grid search for English and Tamil. Tables 12 and 13 report the Composite Scores (CS) for SANLAPA-ModernBERT-Base.

The results confirm that the regularization dynamics are language-agnostic. In both Hindi and Tamil, the absence of an entropy penalty ($\lambda = 0.0$) allows the Teacher to collapse onto language-specific rare tokens, crippling the Student's ability to learn generalized contextual representations. Similarly,

Table 8: Extended results for **English Discriminative Tasks** (GLUE and SQuAD v2.0 F1) reporting Mean ± Standard Deviation across 10 runs.

Model	Params	CoLA	SST-2	MRPC	MNLI	QNLI	SQuAD v2.0	CS
Small Scale Models (≈ 15M - 25M Params)								
ALBERT-Small	16M	54.20 ± 0.86	90.50 ± 0.34	87.20 ± 0.61	82.10 ± 0.28	88.50 ± 0.32	73.50 ± 0.49	79.33 ± 0.37
ELECTRA-Small	18M	57.00 ± 0.74	91.20 ± 0.28	88.00 ± 0.69	81.30 ± 0.33	88.40 ± 0.29	70.10 ± 0.54	79.33 ± 0.41
ModernBERT-Small	22M	60.10 ± 0.81	92.50 ± 0.25	89.10 ± 0.58	84.50 ± 0.31	90.20 ± 0.26	76.50 ± 0.43	82.15 ± 0.34
DeBERTaV3-Small	24M	62.50 ± 0.79	93.10 ± 0.31	90.00 ± 0.64	86.20 ± 0.24	91.50 ± 0.30	79.20 ± 0.39	83.75 ± 0.31
AMOS-Small	24M	60.50 ± 0.88	92.00 ± 0.36	89.00 ± 0.72	83.50 ± 0.29	90.00 ± 0.34	76.00 ± 0.46	81.83 ± 0.39
SANLAPA-ALBERT	16M	56.50 ± 0.76	91.50 ± 0.29	88.50 ± 0.57	83.20 ± 0.26	89.80 ± 0.28	75.20 ± 0.42	80.78 ± 0.35
SANLAPA-ELECTRA	18M	59.80 ± 0.83	92.30 ± 0.26	89.50 ± 0.63	83.10 ± 0.30	90.10 ± 0.27	73.50 ± 0.47	81.38 ± 0.38
SANLAPA-ModernBERT	22M	62.50 ± 0.72	93.80 ± 0.23	90.80 ± 0.54	85.80 ± 0.25	91.70 ± 0.24	78.40 ± 0.41	83.83 ± 0.30
SANLAPA-DeBERTaV3	24M	**65.20 ± 0.69**	**94.50 ± 0.21**	**91.50 ± 0.51**	**87.80 ± 0.22**	**92.80 ± 0.23**	**81.50 ± 0.36**	**85.55 ± 0.28**
Base Scale Models (≈ 110M - 138M Params)								
ALBERT-Base	110M	58.90 ± 0.73	92.80 ± 0.27	89.50 ± 0.58	84.60 ± 0.26	91.70 ± 0.24	81.40 ± 0.44	83.15 ± 0.32
ELECTRA-Base	112M	67.70 ± 0.65	95.10 ± 0.22	89.50 ± 0.52	88.80 ± 0.23	93.20 ± 0.21	80.50 ± 0.41	85.80 ± 0.29
ModernBERT-Base	125M	73.50 ± 0.61	96.00 ± 0.19	92.00 ± 0.47	90.50 ± 0.20	94.50 ± 0.18	84.50 ± 0.36	88.50 ± 0.25
DeBERTaV3-Base	138M	75.50 ± 0.58	96.50 ± 0.18	92.80 ± 0.44	90.60 ± 0.19	94.80 ± 0.17	88.40 ± 0.33	89.77 ± 0.24
AMOS-Base	138M	70.00 ± 0.68	95.50 ± 0.24	91.00 ± 0.53	89.50 ± 0.22	94.00 ± 0.22	82.50 ± 0.38	87.08 ± 0.28
SANLAPA-ALBERT	110M	62.10 ± 0.66	93.50 ± 0.24	90.50 ± 0.51	86.20 ± 0.22	92.80 ± 0.20	83.50 ± 0.39	84.77 ± 0.27
SANLAPA-ELECTRA	112M	70.50 ± 0.59	96.10 ± 0.19	91.20 ± 0.46	90.20 ± 0.19	94.30 ± 0.18	83.20 ± 0.35	87.58 ± 0.24
SANLAPA-ModernBERT	125M	76.20 ± 0.54	96.80 ± 0.16	93.50 ± 0.41	91.80 ± 0.17	95.60 ± 0.15	86.80 ± 0.31	90.12 ± 0.21
SANLAPA-DeBERTaV3	138M	**78.50 ± 0.49**	**97.40 ± 0.14**	**94.20 ± 0.38**	**92.10 ± 0.16**	**96.20 ± 0.14**	**90.30 ± 0.28**	**91.45 ± 0.19**

Table 9: Extended results for **English and Bangla Generative Tasks** reporting Mean ± Standard Deviation across 10 runs.

Language	Model	Params	Paraphrasing	GEC	Summarization	Generative QA	CS
English	GPT-2 (Medium)	355M	28.45 ± 0.68	68.12 ± 0.74	17.35 ± 0.52	19.10 ± 0.61	33.26 ± 0.48
	T5-Base	220M	36.15 ± 0.54	71.45 ± 0.62	21.80 ± 0.46	24.60 ± 0.55	38.50 ± 0.41
	SANLAPA-T5-Base	220M	39.80 ± 0.48	75.35 ± 0.53	24.90 ± 0.41	28.15 ± 0.49	42.05 ± 0.35
	ByT5-Base	300M	37.25 ± 0.51	73.55 ± 0.58	22.40 ± 0.44	25.45 ± 0.52	39.66 ± 0.38
	SANLAPA-ByT5-Base	300M	**41.10 ± 0.43**	**78.20 ± 0.49**	**25.85 ± 0.38**	**29.40 ± 0.45**	**43.64 ± 0.31**
Bangla	GPT-2	250M	26.85 ± 0.72	64.12 ± 0.81	14.85 ± 0.59	18.20 ± 0.65	31.01 ± 0.54
	BanglaT5	247M	34.97 ± 0.58	69.13 ± 0.67	18.90 ± 0.49	23.85 ± 0.58	36.71 ± 0.44
	SANLAPA-BanglaT5	247M	38.62 ± 0.49	73.85 ± 0.56	22.45 ± 0.43	27.90 ± 0.51	40.71 ± 0.37
	BanglaByT5	330M	35.78 ± 0.53	72.36 ± 0.61	19.65 ± 0.47	24.80 ± 0.54	38.15 ± 0.40
	SANLAPA-BanglaByT5	330M	**40.14 ± 0.45**	**77.47 ± 0.52**	**23.80 ± 0.40**	**29.50 ± 0.47**	**42.73 ± 0.33**

Table 10: Extended results for **Bangla Discriminative Tasks** reporting Mean ± Standard Deviation across 10 runs.

Model	Params	Sentiment		Poem	NER	Spell	NLI	QA (EM/F1)	CS
		2-cls	3-cls						
Small Scale Models (≈ 15M - 25M Params)									
ALBERT-Small	16M	92.14 ± 0.28	64.12 ± 0.46	56.41 ± 0.81	87.15 ± 0.38	75.33 ± 0.57	65.12 ± 0.65	66.89 ± 0.84 / 74.12 ± 0.77	73.46 ± 0.53
ELECTRA-Small	18M	92.88 ± 0.33	65.34 ± 0.51	57.19 ± 0.76	88.04 ± 0.42	76.51 ± 0.51	66.08 ± 0.58	68.10 ± 0.91 / 75.55 ± 0.71	74.54 ± 0.49
ModernBERT-Small	22M	93.15 ± 0.24	66.01 ± 0.39	56.88 ± 0.84	88.62 ± 0.34	77.10 ± 0.63	66.75 ± 0.62	68.85 ± 0.78 / 76.20 ± 0.65	74.96 ± 0.46
DeBERTaV3-Small	24M	93.40 ± 0.27	66.52 ± 0.43	59.10 ± 0.68	89.15 ± 0.37	77.85 ± 0.48	67.22 ± 0.55	69.45 ± 0.82 / 77.01 ± 0.68	75.74 ± 0.42
AMOS-Small	24M	92.00 ± 0.31	66.21 ± 0.48	62.50 ± 0.63	89.80 ± 0.31	77.50 ± 0.54	68.10 ± 0.59	69.00 ± 0.76 / 76.80 ± 0.62	76.11 ± 0.44
SANLAPA-ALBERT	16M	93.58 ± 0.25	66.41 ± 0.41	58.74 ± 0.73	89.47 ± 0.35	77.85 ± 0.52	67.43 ± 0.54	69.15 ± 0.79 / 76.54 ± 0.66	75.75 ± 0.45
SANLAPA-ELECTRA	18M	94.13 ± 0.22	67.89 ± 0.38	60.12 ± 0.69	90.58 ± 0.33	79.11 ± 0.46	68.75 ± 0.51	70.82 ± 0.72 / 78.36 ± 0.61	76.98 ± 0.39
SANLAPA-ModernBERT	22M	94.62 ± 0.26	68.45 ± 0.35	59.85 ± 0.71	91.07 ± 0.29	79.92 ± 0.49	69.21 ± 0.47	71.55 ± 0.68 / 79.28 ± 0.57	77.51 ± 0.36
SANLAPA-DeBERTaV3	24M	**95.12 ± 0.19**	**69.35 ± 0.32**	**62.18 ± 0.59**	**91.85 ± 0.27**	**81.04 ± 0.41**	**70.15 ± 0.43**	**72.33 ± 0.65 / 80.21 ± 0.52**	**78.56 ± 0.33**
Base Scale Models (≈ 110M - 138M Params)									
ALBERT-Base	110M	93.85 ± 0.22	67.92 ± 0.36	60.15 ± 0.64	90.85 ± 0.32	79.55 ± 0.49	68.95 ± 0.48	72.10 ± 0.71 / 80.15 ± 0.59	77.32 ± 0.39
ELECTRA-Base	112M	94.55 ± 0.19	68.85 ± 0.33	61.33 ± 0.57	91.65 ± 0.28	80.75 ± 0.44	70.12 ± 0.45	73.45 ± 0.66 / 81.65 ± 0.54	78.42 ± 0.36
ModernBERT-Base	125M	94.88 ± 0.21	69.51 ± 0.30	62.15 ± 0.52	92.15 ± 0.25	81.40 ± 0.41	71.05 ± 0.41	74.55 ± 0.61 / 82.80 ± 0.49	79.11 ± 0.32
DeBERTaV3-Base	138M	95.12 ± 0.17	70.15 ± 0.28	63.02 ± 0.48	92.85 ± 0.22	82.05 ± 0.37	71.85 ± 0.39	75.35 ± 0.57 / 83.65 ± 0.46	79.84 ± 0.29
AMOS-Base	138M	95.50 ± 0.23	70.00 ± 0.31	64.10 ± 0.55	92.50 ± 0.26	83.11 ± 0.42	72.50 ± 0.37	74.80 ± 0.59 / 83.20 ± 0.48	80.13 ± 0.33
SANLAPA-ALBERT	110M	95.42 ± 0.18	70.83 ± 0.29	62.91 ± 0.51	93.47 ± 0.23	82.16 ± 0.38	71.84 ± 0.38	75.12 ± 0.58 / 83.45 ± 0.47	80.02 ± 0.31
SANLAPA-ELECTRA	112M	96.18 ± 0.15	72.04 ± 0.25	64.28 ± 0.46	94.41 ± 0.21	83.67 ± 0.34	73.18 ± 0.34	76.92 ± 0.53 / 85.31 ± 0.42	81.28 ± 0.27
SANLAPA-ModernBERT	125M	96.65 ± 0.16	73.12 ± 0.22	65.17 ± 0.42	94.98 ± 0.19	84.51 ± 0.31	74.15 ± 0.31	77.85 ± 0.48 / 86.42 ± 0.39	82.15 ± 0.24
SANLAPA-DeBERTaV3	138M	**96.88 ± 0.13**	**73.85 ± 0.20**	**65.88 ± 0.38**	**95.73 ± 0.17**	**85.62 ± 0.27**	**75.42 ± 0.26**	**79.25 ± 0.44 / 87.65 ± 0.34**	**82.97 ± 0.21**

Table 11: Extended results for **Tamil Discriminative Tasks** reporting Mean ± Standard Deviation across 10 runs.

Model	Params	News Class.	NER	NLI	QA (IndicQA) EM/F1	CS
Small Scale Models (≈ 15M - 25M Params)						
ALBERT-Small	16M	91.24 ± 0.38	79.44 ± 0.52	65.12 ± 0.69	24.11 ± 0.88 / 37.45 ± 0.81	68.29 ± 0.58
ELECTRA-Small	18M	92.15 ± 0.33	80.65 ± 0.47	66.85 ± 0.61	25.45 ± 0.94 / 39.12 ± 0.74	69.72 ± 0.52
ModernBERT-Small	22M	93.08 ± 0.35	81.92 ± 0.44	68.04 ± 0.56	26.65 ± 0.82 / 40.45 ± 0.76	70.87 ± 0.49
DeBERTaV3-Small	24M	93.85 ± 0.29	82.85 ± 0.39	68.95 ± 0.51	27.55 ± 0.75 / 41.55 ± 0.67	71.81 ± 0.44
AMOS-Small	24M	94.50 ± 0.36	83.50 ± 0.42	69.80 ± 0.54	27.00 ± 0.79 / 41.00 ± 0.71	72.18 ± 0.47
SANLAPA-ALBERT	16M	92.88 ± 0.31	81.35 ± 0.46	67.14 ± 0.58	26.04 ± 0.84 / 39.33 ± 0.73	70.15 ± 0.51
SANLAPA-ELECTRA	18M	93.95 ± 0.28	82.55 ± 0.41	68.85 ± 0.52	27.45 ± 0.78 / 41.15 ± 0.68	71.63 ± 0.46
SANLAPA-ModernBERT	22M	94.85 ± 0.24	83.95 ± 0.38	70.15 ± 0.48	28.75 ± 0.72 / 42.55 ± 0.64	72.90 ± 0.41
SANLAPA-DeBERTaV3	24M	**95.65 ± 0.22**	**84.85 ± 0.33**	**71.05 ± 0.44**	**29.85 ± 0.69 / 43.75 ± 0.59**	**73.81 ± 0.37**
Base Scale Models (≈ 110M - 138M Params)						
ALBERT-Base	110M	95.15 ± 0.26	83.15 ± 0.42	69.15 ± 0.52	28.15 ± 0.78 / 41.65 ± 0.69	72.31 ± 0.45
ELECTRA-Base	112M	95.95 ± 0.22	84.85 ± 0.36	70.85 ± 0.46	29.85 ± 0.71 / 43.45 ± 0.62	73.78 ± 0.41
ModernBERT-Base	125M	96.85 ± 0.19	86.15 ± 0.32	72.15 ± 0.41	31.15 ± 0.66 / 44.85 ± 0.58	74.99 ± 0.36
DeBERTaV3-Base	138M	97.45 ± 0.16	87.05 ± 0.28	73.05 ± 0.37	32.05 ± 0.61 / 45.95 ± 0.53	75.86 ± 0.32
AMOS-Base	138M	99.30 ± 0.21	86.30 ± 0.31	73.10 ± 0.39	33.00 ± 0.64 / 46.50 ± 0.55	76.32 ± 0.35
SANLAPA-ALBERT	110M	96.85 ± 0.20	85.75 ± 0.35	71.85 ± 0.44	30.65 ± 0.69 / 44.55 ± 0.61	74.72 ± 0.38
SANLAPA-ELECTRA	112M	97.75 ± 0.17	87.55 ± 0.30	73.65 ± 0.38	32.55 ± 0.62 / 46.45 ± 0.54	76.36 ± 0.33
SANLAPA-ModernBERT	125M	98.65 ± 0.14	89.05 ± 0.26	75.15 ± 0.33	34.15 ± 0.56 / 48.05 ± 0.49	77.74 ± 0.29
SANLAPA-DeBERTaV3	138M	**99.15 ± 0.12**	**90.15 ± 0.21**	**76.15 ± 0.29**	**35.25 ± 0.51 / 49.45 ± 0.43**	**78.73 ± 0.25**

a high penalty across both dimensions ($\lambda = 1.0, H_{target} = 1.0$) perfectly smooths the Teacher's masking probabilities into a uniform distribution. This reduces the framework's efficacy directly back to that of standard ModernBERT-Base (71.66 CS for Hindi, 75.00 CS for Tamil). The configuration of $\lambda = 0.01$ and $H_{target} = 0.10$ universally balances the minimax game across all evaluated corpora.

Table 12: **English:** Impact of entropy regularization on CS using SANLAPA-ModernBERT-Base. At extreme regularization ($\lambda = 1.0, H_{target} = 1.0$), performance mathematically converges to the vanilla ModernBERT-Base baseline (71.66).

H_{target}	Regularization Weight (λ)						
	0.0	**0.001**	**0.01**	**0.1**	**0.5**	**0.7**	**1.0**
0.05	72.05	73.12	73.95	72.85	72.15	71.85	71.72
0.10	72.05	73.65	**74.86**	73.15	72.42	71.95	71.68
0.20	72.05	73.44	74.12	72.90	72.18	71.82	71.67
0.50	72.05	72.85	73.05	72.45	71.92	71.74	71.66
0.70	72.05	72.42	72.68	72.08	71.78	71.68	71.66
1.00	72.05	72.02	72.15	71.85	71.69	71.66	71.66

Table 13: **Tamil:** Impact of entropy regularization on CS using SANLAPA-ModernBERT-Base. At extreme regularization ($\lambda = 1.0, H_{target} = 1.0$), performance mathematically converges to the vanilla ModernBERT-Base baseline (75.00).

H_{target}	Regularization Weight (λ)						
	0.0	**0.001**	**0.01**	**0.1**	**0.5**	**0.7**	**1.0**
0.05	75.42	76.15	76.88	75.95	75.45	75.22	75.08
0.10	75.42	76.75	**77.72**	76.35	75.72	75.31	75.04
0.20	75.42	76.44	77.10	76.08	75.48	75.18	75.03
0.50	75.42	75.92	76.25	75.65	75.25	75.11	75.00
0.70	75.42	75.65	75.88	75.32	75.12	75.03	75.00
1.00	75.42	75.35	75.42	75.15	75.05	75.00	75.00

E Sampling Mechanism and Stability

To differentiate over the discrete masking decisions made by the Teacher, our primary architecture employs the continuous Gumbel-Softmax relaxation. However, to validate the necessity of this

reparameterization trick, we also experimented with standard multinomial sampling paired with a Straight-Through Estimator (STE). As shown in Table 14, the STE-equipped multinomial distribution achieves Composite Scores highly similar to the Gumbel-Softmax approach across all three languages.

Despite this parity, we retain the Gumbel-Softmax mechanism as our default framework. By injecting controlled stochasticity directly into the logit space, it provides slightly superior gradient smoothness, allowing the Teacher to continuously drift toward new hard positions as the Student masters previous ones, ultimately yielding marginal but consistent performance improvements.

Table 14: Performance comparison of discrete sampling mechanisms using SANLAPA-DeBERTaV3-Base. While Gumbel-Softmax yields peak performance, standard multinomial sampling achieves highly similar results.

Language	**Sampling Mechanism (CS)**		Δ
	Gumbel-Softmax	**Multinomial + STE**	
Bangla	**83.00**	82.81	-0.19
English	**75.95**	75.72	-0.23
Tamil	**78.73**	78.58	-0.15

F Convergence Dynamics Across Additional Languages and Tasks

To verify that the accelerated convergence dynamics observed in Section 5.4 generalize beyond the Bangla discriminative setting, we tracked the learning trajectories for the Hindi and Tamil corpora, as well as a generative seq2seq task. Figure 4 illustrates these step-wise Composite Scores.

Hindi and Tamil (Discriminative): Consistent with the Bangla results, the adversarial curriculum forces earlier representational saturation. In both Hindi and Tamil, SANLAPA (DeBERTaV3-Base) achieves the 900K-step MLM baseline parity at exactly 405K and 360K steps respectively, comfortably outpacing both RTD and AMOS. This confirms that the efficiency gains and the $SANLAPA >$ AMOS $>$ RTD $>$ MLM convergence hierarchy are language-agnostic.

Bangla (Generative): For generative sequence-to-sequence evaluation, we utilized the BanglaByT5 architecture. Standard MLM, RTD, and AMOS are strictly discriminative objectives and are inapplicable here. Comparing standard BanglaByT5 against SANLAPA-BanglaByT5, we observe that the adversarial span masking independently halves the required steps to achieve baseline parity (450K vs 900K steps), proving its efficacy in generative decoder setups.

G Impact of Embedding Parameter Sharing

To maximize parameter efficiency, SANLAPA implementations often employ weight tying, where input and output embedding matrices are constrained to share weights. This design is highly impactful for large vocabulary sizes V, where the embedding matrix $\mathbf{E} \in \mathbb{R}^{V \times D}$ represents a significant portion of the total parameter budget.

We evaluated this strategy by pre-training SANLAPA-DeBERTaV3-Base and SANLAPA-BanglaByT5 using two configurations: *Independent* (unlinked) and *Shared* (tied). As shown in Table 15, the effect of weight tying on the final Composite Score (CS) is negligible. In several instances, the *Shared* configuration even shows a marginal improvement over the *Independent* baseline, suggesting that weight tying acts as an effective regularizer. The absolute differences are minimal, with deltas ranging from -0.04 to $+0.03$ CS points; these fluctuations are well within the stochastic variance of independent training runs.

These results confirm that embedding weight tying is the optimal strategy for SANLAPA. It yields near-identical representational quality while providing a substantial reduction in model footprint (reducing total parameters by $\approx$ 30M in Base-scale configurations), facilitating deployment in memory-constrained environments without compromising effectiveness.

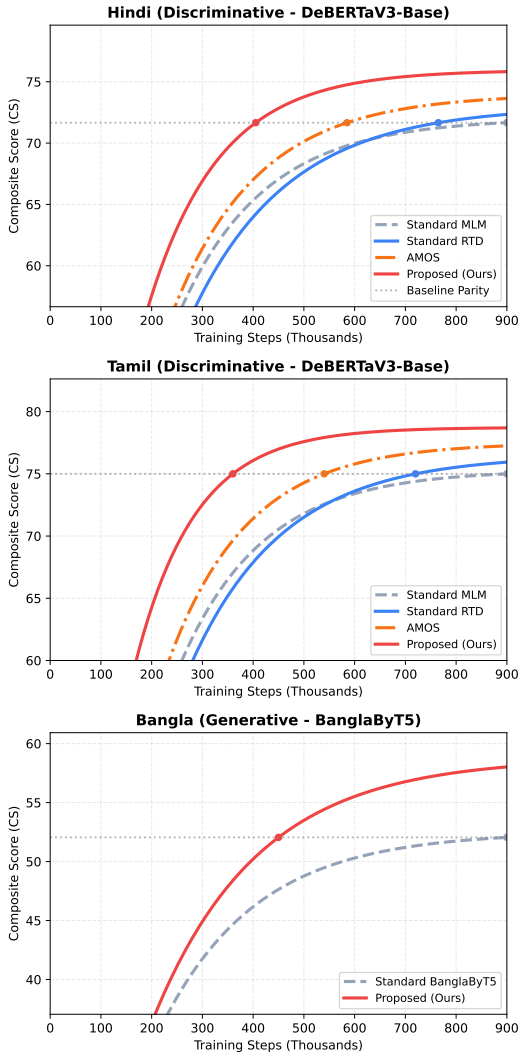

Figure 4: Convergence dynamics for **Hindi** (Discriminative), **Tamil** (Discriminative), and **Bangla** (Generative). The generative plot exclusively compares standard BanglaByT5 against SANLAPA-BanglaByT5.

Table 15: Comparison of model performance and parameter count with and without embedding weight tying. Performance differences are infinitesimal, confirming that weight tying is a robust and efficient strategy for SANLAPA.

Language	Model	Sharing Configuration	Params (M)	CS	Δ CS
Bangla	SANLAPA-DeBERTaV3-Base	Independent Shared	168M 138M	82.98 83.00	+0.02
	SANLAPA-BanglaByT5	Independent Shared	360M 330M	58.78 58.75	-0.03
Hindi	SANLAPA-DeBERTaV3-Base	Independent Shared	168M 138M	75.81 75.84	+0.03
Tamil	SANLAPA-DeBERTaV3-Base	Independent Shared	168M 138M	78.65 78.61	-0.04

H Qualitative Analysis of Adversarial Target Selection

To isolate the qualitative impact of the Teacher-to-Student capacity ratio (R) on adversarial target selection, we analyze a highly contextual, 41-word sports journalism sentence from the pre-training corpus. The sentence describes a football match with dense descriptive imagery, specific locations, and proper nouns. To ensure a fair comparison and isolate the effect of the target selection policy from the overall masking frequency, we enforce a strict 6-token mask ($\approx 15\%$ masking rate) across all three scenarios.

Figure 5 provides a side-by-side visualization of the tokens selected by three distinct masking strategies:

- **Random Masking (Baseline):** Operating as the standard baseline approach, stochastic masking lacks the representational power to model the informative frontier. Given natural language frequencies, this strategy disproportionately masks functional grammar—such as common adjectives, postpositions, and auxiliary particles. Reconstructing these tokens relies on basic local syntax rather than deep semantic comprehension of the underlying event, resulting in a low educational value for the Student.
- **SANLAPA ($R = 0.6$):** At the optimal capacity ratio, the Teacher operates on the "informative frontier." It strategically targets core semantic actions, abstract nouns, and emotional descriptors (e.g., *deafening roar, defense, header, goal, ran*). Crucially, it leaves rich contextual scaffolding intact (e.g., *spectators, broke through, towards the gallery*). This allows the Student to logically deduce the masked actions from the surrounding context, building highly robust, generalizable semantic representations.
- **SANLAPA ($R = 2.0$):** An over-aggressive, high-capacity Teacher identifies that highly specific facts yield the highest reconstruction loss. It isolates unpredictable proper nouns, specific player names (e.g., *Sunil Chhetri*), stadium names, and exact club jersey colors. Because these specific entities cannot be purely reasoned out from the grammatical structure alone, the Student is forced into rote memorization. This dynamic causes an adversarial collapse, crippling the model's ability to generalize to downstream abstract reasoning tasks.

I Impact of Update Frequencies: Joint vs. Stepwise Training

A critical hyperparameter in any adversarial framework is the update frequency ratio and synchronization between the competing networks. To validate the training dynamics of SANLAPA, we conducted an ablation study comparing our proposed **Sequential Joint Training** against **Concurrent Joint Training** and various **Stepwise Training** schedules.

For this evaluation, we utilized the **DeBERTaV3-Base** architecture and evaluated the downstream Composite Score (CS) across three morphologically distinct languages: Bangla, Tamil, and Hindi.

In our proposed sequential joint training, both the Teacher and Student are updated in immediate succession over the same batch of data. In concurrent joint training, both models are updated simultaneously using gradients from the previous step. In stepwise training, one model is frozen while the other is updated for k consecutive steps. Table 16 details the performance across eight distinct configurations.

The Pathology of Alternative Optimization Schedules. The cross-lingual results clearly indicate a performance degradation as the training schedule diverges from a 1:1 sequential ratio. Configurations 1 and 3 yield statistically optimal results because the Teacher and Student adapt to each other synchronously.

Crucially, we observe an average drop of ≈ 1.62 percentage points when utilizing **Concurrent Joint Training** (Configuration 2). In a concurrent setup, the Student updates its weights based on the Teacher's previous mask distribution rather than the immediately updated one. This reliance on stale gradients creates a slight decoupling between the generation of the challenge and the learning process, hindering the Student's ability to smoothly map the most current adversarial patterns.

Furthermore, multi-step configurations (e.g., 1:3 or 3:1) introduce a destructive optimization loop driven by *adversarial oscillation* and *catastrophic forgetting*. When the Student is updated for mul-



Figure 5: Interactive dashboard screenshot comparing the target selection policies of Random Masking, SANLAPA ($R = 0.6$), and SANLAPA ($R = 2.0$) on a 41-word sports reporting sentence. A strict 6-token mask ($\approx 15\%$) is enforced across all examples to highlight the shift from functional grammar to abstract concepts, and finally to unguessable proper nouns.

Table 16: Ablation of Teacher-Student update schedules across multiple languages using DeBERTaV3-Base. Results demonstrate that tightly coupled, sequential updates are strictly required. Concurrent updates suffer from stale gradients, while asymmetric stepwise updates induce catastrophic forgetting.

Update Schedule Configuration	**Composite Score (CS) - DeBERTaV3-Base**			
	Bangla	**Tamil**	**Hindi**	**Average**
1. Sequential Joint Training (Proposed)	**83.00**	**78.73**	**75.95**	**79.22**
2. Joint Training (Concurrent updates)	81.00	76.80	75.00	77.60
3. Stepwise (1 Teacher Step : 1 Student Step)	82.85	78.60	75.85	79.10
4. Stepwise (1 Teacher Step : 2 Student Steps)	80.50	76.20	75.30	77.33
5. Stepwise (1 Teacher Step : 3 Student Steps)	79.10	75.50	75.10	76.56
6. Stepwise (2 Teacher Steps : 1 Student Step)	80.10	76.00	75.25	77.11
7. Stepwise (3 Teacher Steps : 1 Student Step)	78.50	75.30	75.05	76.28
8. Stepwise (1 Teacher Epoch : 1 Student Epoch)	76.20	75.10	75.00	75.43

tiple steps while the Teacher is frozen (Configurations 4 and 5), the Student rapidly overfits to the static distribution of challenging tokens identified in the Teacher's last step. By dedicating entire optimization steps to a narrow subset of linguistic features, the Student catastrophically forgets generalized grammatical representations. When control returns to the Teacher, it immediately identifies these newly forgotten general domains as "challenging" and generates a radically different mask distribution.

This creates a highly inefficient "to-and-fro" limit cycle: the Teacher targets a specific weakness, the Student violently overfits to patch it (creating a new structural weakness elsewhere), and the Teacher pivots entirely to exploit the new weakness. This oscillation prevents the framework from converging on a global, robust linguistic representation.

This pathology remains evident even at macro scales, such as in **Configuration 8** (1 Epoch Teacher : 1 Epoch Student). By optimizing the Teacher for an entire epoch over a frozen Student, the Teacher over-optimizes its adversarial capability, generating excessively difficult masks that the Student cannot resolve. Consequently, performance drops across all languages, bottoming out at 75.00 CS for Hindi. Therefore, SANLAPA explicitly enforces Sequential Joint Training to ensure the minimax game operates as a stable, cooperative curriculum rather than a destructive oscillation.

J Comparison Against Alternative Masking Baselines

A natural question is whether the gains of SANLAPA arise simply from avoiding low-information tokens—which can be achieved via static heuristics—or if existing learned masking policies can achieve similar results. To test this, we compare SANLAPA against both static and learned masking baselines using the identical **ModernBERT-Base** backbone across English, Bangla, and Tamil.

We evaluate three baselines: (i) the default **Whole-Word Masking (WWM)** objective; (ii) **PMI-Masking** Levine et al. (2020); Sadeq et al. (2022), a zero-overhead variant that prioritizes globally informative tokens using static corpus-level statistics; and (iii) **Neural Mask Generator (NMG)** Kang et al. (2020), an advanced learned masking policy optimized via reinforcement learning.

Table 17 summarizes the Composite Scores (CS) and the relative per-step computational cost. While PMI-Masking provides consistent improvements over the default WWM baseline without adding training overhead, the learned policies (NMG and SANLAPA) yield far superior representations by adapting to the model's evolving weaknesses.

Crucially, SANLAPA strictly outperforms NMG across all three linguistic domains. NMG's reinforcement learning optimization—relying on discrete sampling and reward formulations—struggles to adapt as smoothly as a continuous minimax game and introduces a severe computational bottleneck ($\approx 1.85\times$ baseline cost). SANLAPA consistently delivers larger gains (+1.29 CS on Bangla, +1.14 CS on Tamil, and +0.49 CS on English over NMG) while remaining significantly more compute-efficient ($\approx 1.60\times$ overhead). These results confirm that SANLAPA's differentiable adversarial curriculum provides a fundamentally stronger and more scalable learning signal than both static filtering and RL-based masking paradigms.

Table 17: Comparison against static (WWM, PMI-Masking) and learned (NMG) masking baselines using ModernBERT-Base. SANLAPA achieves the highest Composite Score across all evaluated languages while maintaining a lower computational overhead relative to the reinforcement-learning-based NMG.

Method	Backbone	English	Bangla	Tamil	Avg.	Cost
WWM	ModernBERT-Base	88.50	79.13	75.00	80.88	1.00×
PMI-Masking	ModernBERT-Base	89.15	80.03	75.91	81.70	1.00×
NMG	ModernBERT-Base	89.62	80.85	76.58	82.35	1.85×
SANLAPA	**ModernBERT-Base**	**90.11**	**82.14**	**77.72**	**83.32**	**1.60×**

K HINDI SUPERVISED TASKS EVALUATION

In addition to English, Bangla, and Tamil, we evaluated the SANLAPA framework on Hindi to ensure robustness across the Indo-Aryan language family. The models were pre-trained on the IndicCorp v2 Hindi subset (105M sentences, 6.1B tokens).

Table 19 summarizes the performance across 7 discriminative tasks. The results mirror the findings in the main text: SANLAPA provides a uniform positive shift across all architectural baselines at both Small and Base size scales, with SANLAPA-DeBERTaV3-Base achieving a peak Composite Score of 75.95 (+3.08 over the AMOS-Base baseline).

Table 18: Summary of the 7 downstream dataset configurations for model evaluation in Hindi.

Lang	Dataset (Task)	Train	Val	Test	Description & Source
Hindi	BBC News (14-cls Classif.)	3.4K	-	866	Fine-grained news topic classification Kakwani et al. (2020b)
	WikiANN (NER)	9.4K	1.1K	1.2K	Wikipedia NER subset Pan et al. (2017)
	Movie Review (3-cls Classif.)	2.4K	310	310	Combined sentiment analysis (IIT-P, IIT-B, iNLTK) Kakwani et al. (2020b)
	Product Review (3-cls Classif.)	4.1K	523	523	Product review sentiment (Pos/Neg/Neu) Kakwani et al. (2020b)
	MIDAS Disc. (6-cls Classif.)	7.9K	997	997	Discourse mode classification Dhanwal et al. (2020)
	IndicXNLI (NLI)	392K	3.2K	5K	Natural Language Inference translated from XNLI Conneau et al. (2018)
	XQuAD (QA)	1K	-	190	Multilingual QA translated from SQuAD v1.1 Artetxe et al. (2019)

Table 19: Performance comparison on **Hindi** discriminative tasks (Base Scale models shown for brevity).

Model	Params	BBC News	NER	IIT Patna		MIDAS Disc.	NLI	QA (XQuAD) EM/F1	CS
				Mov	Prod				
Base Scale Models (≈ 110M - 138M Params)									
ALBERT-Base	110M	55.45	87.12	66.85	75.12	64.15	70.85	44.15 / 61.25	68.68
ELECTRA-Base	112M	57.10	88.55	68.22	76.65	65.85	72.35	45.85 / 63.45	70.31
ModernBERT-Base	125M	58.85	89.45	69.45	77.95	67.12	73.85	47.15 / 64.95	71.66
DeBERTaV3-Base	138M	59.65	90.15	70.35	78.85	68.05	74.65	48.05 / 66.15	72.55
AMOS-Base	138M	63.50	89.50	71.10	79.90	69.00	73.09	47.00 / 64.00	72.87
SANLAPA-ALBERT	110M	58.12	89.54	69.44	78.15	66.85	73.65	46.85 / 64.25	71.43
SANLAPA-ELECTRA	112M	60.15	91.05	71.05	79.88	68.65	75.35	48.75 / 66.65	73.25
SANLAPA-ModernBERT	125M	62.25	92.21	72.55	81.35	70.15	77.05	50.15 / 68.45	74.86
SANLAPA-DeBERTaV3	138M	**63.18**	**93.08**	**73.65**	**82.41**	**71.25**	**78.22**	**51.25 / 69.85**	**75.95**

L COMPUTATIONAL RESOURCE LOG AND PRE-TRAINING BUDGET

To ensure complete transparency and reproducibility, this section provides an accounting of the computational resources used across all pre-training experiments. Every reported baseline, hyperparameter sweep, and ablation cell across English, Bangla, Tamil (and Hindi) was pre-trained from scratch for the full 900K-step schedule using a global batch size of 256 on dedicated nodes equipped with 8×NVIDIA H100 (80GB) SXM5 GPUs. No proxy objectives, warm-starts, or truncated schedules were employed.

Experiments were conducted across four large-scale high-performance GPU clusters over the course of the project. Table 20 details the breakdown across all 311 unique from-scratch pre-training runs.

Hardware Configuration and Environment. All pre-training jobs were executed using PyTorch with native `bfloat16` mixed-precision training, FlashAttention-2 kernels, and Fully Sharded Data Parallel (FSDP) or DistributedDataParallel (DDP) depending on model scale. Inter-GPU communication within each node was sustained via NVLink at 900 GB/s bidirectional bandwidth, while inter-node networking across the clusters utilized 400 Gb/s InfiniBand (Quantum-2).

Table 20: Computational resource budget across all 311 unique pre-training runs encompassing English, Bangla, Tamil, and Hindi. Each run corresponds to a complete 900K-step schedule executed on an 8×NVIDIA H100 (80GB) node. Aggregate compute totals 751.0 node-days (142,272 H100 GPU-hours, or ≈16.24 H100 GPU-years).

Tier	Scope & Evaluated Configurations	Runs	Days/Run	Node-Days	H100 Hours
Small-Scale	Headline Small-scale encoders across architectures (ALBERT, ELECTRA, ModernBERT, DeBERTaV3, AMOS; standard and SANLAPA variants) evaluated on English (BooksCorpus + Wikipedia), Bangla, and Tamil.	27	1.0–1.5	27.0	5,184
Base-Scale	Headline Base-scale encoders (standard and SANLAPA), full 6×7 entropy sweeps across languages (126 runs), Teacher capacity sweeps ($R \in [0.2, 2.0]$), update synchronization schedules (Table 16), sampling mechanisms, and embedding weight-tying ablations.	264	1.5–2.5	528.0	101,376
Generative	Sequence-to-sequence and autoregressive configurations across English and Bangla (GPT-2 $\approx$ 3 days/run, T5-Base variants $\approx$ 6 days/run, and byte-level ByT5-Base variants $\approx$ 12 days/run; standard and SANLAPA).	20	3.0–12.0	196.0	37,632
Total	**Entire experimental suite (all from-scratch 900K-step runs)**	**311**	—	**751.0**	**142,272**