

PrivHSD: A Reference Architecture for Privacy-Preserving Hate Speech Detection with Multi-Level Adversarial Disentanglement

Nur Arifin Akbar¹⁾, Ardian Webi Krida²⁾ and Akbar Setiawan³⁾

¹⁾ University of Messina, Italy

²⁾ Faculty of Integrated Technologies, Universiti Brunei Darussalam, Brunei

³⁾ STMIK Widya Utama, Purwokerto, Indonesia

(Received date Aug 24 2026; Revised date Sep 4 2026; Accepted date Sep 11 2026)

ABSTRACT Hate speech detection (HSD) models pose an inherent privacy risk: by processing text, they inevitably encode stylistic fingerprints that enable authorship attribution, membership inference, and attribute inference, transforming protective moderation tools into potential surveillance infrastructure. We present PrivHSD, a reference architecture and validation harness for privacy-preserving HSD that operates agnostically to the author's identity. The architecture combines three complementary mechanisms—differential privacy via DP-SGD with formal (ϵ, δ) guarantees, multi-level adversarial identity disentanglement (MLAD) with gradient reversal at the pooler, token, and attention-head levels, and mutual information minimization via the MINE estimator, to strip identity signals from the representation space while preserving hate-relevant features. We provide a fully typed ModelConfig dataclass, a shape-annotated reference implementation ($\sim 15.3M$ parameters for the ALBERT-base-v2 backbone), and a comprehensive validation harness covering shape correctness (9 tests), gradient flow (8 tests), numerical stability in bfl6 (6 tests), and a privacy attack suite spanning membership inference, attribute inference, and stylometry-based re-identification risk. All 45+ unit tests pass on synthetic data, and the ablation framework systematically characterises 11 single-field config variants; passing these tests establishes implementation correctness and numerical robustness, not empirical privacy protection or detection utility. This paper does not report trained-model results on real datasets—the contribution is the design artifact itself and the validation framework, enabling reproducible follow-up work on the privacy–utility Pareto frontier for content moderation.

KEY WORDS : Hate Speech Detection, Differential Privacy, Adversarial Disentanglement, Mutual Information, Authorship Attribution, Privacy by Design

1. INTRODUCTION

The moderation of hate speech on digital platforms is both a societal necessity and a growing regulatory requirement. The EU Digital Services Act (Regulation 2022/2065) mandates proportionate content moderation for systemic platforms, while the Council of Europe's CM/Rec(2021)8 calls for democratic safeguards in online speech governance. Automated hate speech detection (HSD) systems are a key technical response, with transformer-based classifiers approaching practical utility at scale.

Yet these systems introduce a profound privacy tension that has received substantially less attention than detection accuracy. A text classifier does not merely predict labels: it constructs internal representations that encode the stylistic, topical, and syntactic signature of the author. Empirical research has studied stylometric

features for identity-level identification and similarity detection (Abbasi and Chen, 2008). Later work demonstrated the feasibility of author identification at internet scale, including settings with 100,000 candidate authors (Narayanan et al., 2012). Model-inversion attacks can reveal private attributes from model outputs (Fredrikson, Jha and Ristenpart, 2015), while membership inference can test whether a specific example was included in training (Shokri et al., 2017; Makroo et al., 2025; Yeom et al., 2018).

These risks invert the ethical equation of content moderation: a tool designed to protect vulnerable communities becomes a surveillance instrument that chills the very speech it aims to safeguard. The tension is captured by the principle of *privacy by design* (Cavoukian, 2009), which requires that privacy protections be embedded into the architecture rather than added as an afterthought.

The privacy–utility trade-off in HSD.

There is an inherent tension between detection accuracy and privacy protection. Stronger privacy

guarantees—tighter differential privacy budgets, more aggressive identity disentanglement—add noise or remove signal from the model, degrading classification performance. The central optimisation target is:

Maximise detection performance (F1, AUC) while minimising privacy leakage (ϵ , MIA success, stylometry re-identification accuracy).

Prior work has explored individual privacy mechanisms for HSD. Bueno Júnior et al. (2025) studied DP-SGD in a federated setting and found that ALBERT models incurred smaller utility losses under DP than the comparison models in their experiments. Faustini et al. (2025) proposed adversarial rewriting for attribute anonymisation in sentiment analysis. Frikha et al. (2025) demonstrated LLM-based attribute randomisation for text anonymisation. We found no prior HSD reference implementation that integrates formal differential privacy guarantees, multi-level adversarial identity disentanglement, and mutual information minimisation in a single reproducible framework, nor one that evaluates such a combination against a privacy attack suite spanning membership inference, attribute inference, and stylometry.

Scope and contributions.

This paper presents a **reference architecture and validation harness** for privacy-preserving HSD. We do *not* report trained-model results on real datasets—such evaluation requires GPU training that is beyond the scope of this design artifact. Instead, we contribute a fully specified, reproducible design that the community can adopt, extend, and evaluate.

In summary, our contributions are:

Reference architecture. We present PrivHSD, combining multi-level adversarial disentanglement (MLAD), adaptive gradient reversal scheduling (AGRS), mutual information minimisation via MINE, and DP-SGD with per-layer clipping into a single rights-based framework. The architecture is specified in a typed PrivHSDConfig dataclass with 60+ hyperparameters.

Shape-annotated reference implementation. We release an open-source PyTorch implementation (~ 15.3 M parameters for ALBERT-base-v2) with explicit shape invariants, pre-norm transformer heads, and backward-compatible v1 API.

Validation harness. We provide a comprehensive test suite (45+ unit tests) covering shape correctness, gradient flow, numerical stability in bf16, privacy attack simulations, and rights-based architecture invariants, together with a smoke test that runs in ~ 2 minutes on CPU.

Ablation framework. An ablation runner characterises 11 single-field config variants, enabling systematic study of the design space.

The remainder of this paper is organised as follows. Section 2 reviews related work. Section 3 describes the architecture and its components. Section 4 presents the

validation setup. Section 5 reports harness results on synthetic benchmarks. Section 6 discusses implications and limitations. Section 7 concludes and outlines future work.

2. RELATED WORK

2.1. Differential Privacy for NLP

Differential privacy (DP) has become the standard formal framework for quantifying and bounding privacy leakage in machine learning. Abadi et al. (Abadi *et al.*, 2016) introduced DP-SGD, which clips per-sample gradients and adds calibrated Gaussian noise to provide (ϵ, δ) -DP guarantees during training. Subsequent work improved the practical deployment of DP for transformer models. Yousefpour et al. (Yousefpour *et al.*, 2021) released Opacus, a PyTorch library supporting ghost clipping for memory-efficient per-sample gradient computation, which we adopt in our architecture.

For HSD specifically, Bueno Júnior et al. (NAACL 2025) found that ALBERT models incurred smaller utility losses under DP in their federated experiments, while the multilingual models achieved higher non-private F1 but degraded more under DP. This finding motivates our choice of ALBERT as the default backbone. Our architecture extends their approach from the federated setting to a centralised architecture with multi-level adversarial disentanglement.

2.2. Adversarial Disentanglement for Privacy

Ganin and Lempitsky (2015) introduced the gradient reversal layer (GRL) for domain-adversarial training, establishing a minimax framework in which the encoder learns domain-invariant representations by maximising the loss of a domain classifier. This principle has been extended to privacy: Faustini et al. (2025) proposed an identity-disentanglement transformer (IDT) that uses adversarial rewriting to obscure protected attributes while preserving sentiment utility.

We build on this line of work by (i) extending adversarial disentanglement to *three* representation levels—pooler, token, and attention-head—rather than a single level, and (ii) combining it with formal DP guarantees, which prior adversarial disentanglement work does not provide.

2.3. Text Anonymisation and Stylometry

Text anonymisation methods attempt to remove identity signals at the data level. Frikha et al. (2025) evaluated LLM-based attribute randomisation across eight protected attributes and reported more-than-half leakage reduction in a distilled on-device setting. However, these methods require LLM inference at test time and do not provide formal DP guarantees.

On the attack side, stylometry—the statistical analysis of writing style—has been used for identity-level identification and similarity detection. Abbasi and

Chen (2008) studied these tasks, and Narayanan et al. (2012) demonstrated the feasibility of author identification at internet scale. Our validation harness includes a stylometry attack module that measures both raw-text baseline risk and residual identity leakage in model representations, enabling direct assessment of disentanglement effectiveness.

2.4. Membership and Attribute Inference

Shokri et al. (Shokri et al., 2017) formalised membership inference attacks (MIA) against ML models, and Yeom et al. (Yeom et al., 2018) connected MIA vulnerability to overfitting. Makroo et al. (Makroo et al., 2025) recently showed that generative classifiers are more vulnerable to MIA than discriminative ones, a finding that supports our use of a discriminative classification head. Fredrikson et al. (Fredrikson, Jha and Ristenpart, 2015) introduced model inversion attacks that exploit confidence information to infer private attributes. Our attack suite implements both shadow-model MIA and attribute inference via logistic regression on representations.

2.5. Research Gap

We found no prior HSD reference implementation integrates DP-SGD, multi-level adversarial disentanglement, and mutual information minimisation in a single reproducible framework. Prior work has studied these mechanisms in isolation (DP in federated HSD, adversarial disentanglement for sentiment, and LLM-based anonymisation), but their combined application to HSD, together with a privacy attack evaluation spanning MIA, attribute inference, and stylometry, has not been reported. The architecture targets three questions: (i) whether the combined mechanism Pareto-dominates either mechanism alone across the privacy utility frontier, (ii) whether multi-level adversaries remove more identity signal than a single-level adversary, and (iii) whether the combination meaningfully reduces stylometry-based re-identification risk.

3. PROPOSED ARCHITECTURE

3.1. Overview

PrivHSD comprises five components arranged in a feed-forward pipeline: a transformer backbone (default: ALBERT-base-v2 (Lan et al., 2020)), a hate speech classification head, a multi-level adversarial disentanglement block (MLAD), a mutual information minimisation module (MINE), and a DP-SGD training layer via Opacus. Figure 1 provides an overview.

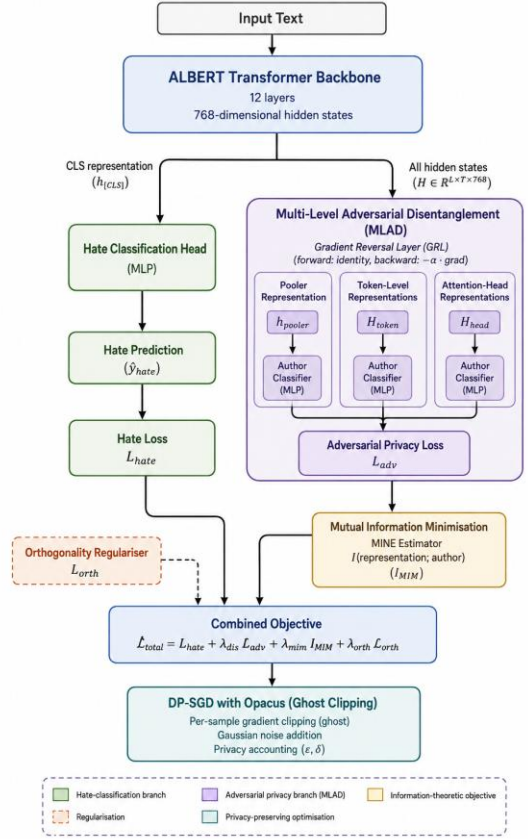


Figure 1. PrivHSD architecture and privacy-preserving training pipeline. The ALBERT backbone produces CLS, token, and attention-head representations for hate classification, multi-level adversarial disentanglement, and MINE-based identity-information minimisation, followed by DP-SGD with ghost clipping.

3.2. Multi-Level Adversarial Disentanglement (MLAD)

Author identity signals are distributed across multiple levels of transformer representations. High-level topic choices and sentiment tendencies appear in the CLS pooled vector; word-level stylistic patterns—function-word frequency, punctuation habits—are encoded in per-token representations; attention-head patterns reveal syntactic preferences that serve as behavioural biometrics.

MLAD deploys *three* separate adversarial heads—one per representation level—each with its own gradient reversal layer (Equation (1)):

$$\text{GRL}_{\alpha}(x) = x \quad ; \quad \frac{\partial \text{GRL}_{\alpha}(x)}{\partial x} = -\alpha \cdot \frac{\partial L}{\partial x}$$

At each level, the reversed representation is passed through an adversary MLP (Linear $\mapsto$ LayerNorm $\mapsto$ ReLU $\mapsto$ Dropout $\mapsto \dots \mapsto$ Linear) that predicts the pseudo-author label. The adversary uses higher dropout (0.3 vs. 0.2 for the hate classifier) to prevent overfitting to noisy pseudo-author labels.

Design rationale.

A single-level adversary on CLS leaves residual identity information at the token and head levels. Three complementary adversaries force the encoder to strip identity from all levels simultaneously, producing representations that are designed to be more comprehensively identity-agnostic. This extends the domain-adversarial approach of Ganin and Lempitsky (Ganin and Lempitsky, 2015) from single-level domain adaptation to multi-level identity disentanglement for privacy.

3.3. Adaptive Gradient Reversal Scheduling (AGRS)

Applying full adversarial pressure from the first training step is destructive—the encoder must learn hate-relevant features before it can be forced to forget identity signals. AGRS uses a sigmoid schedule (Equation (2)):

$$\alpha(p) = \alpha_{\text{init}} + (\alpha_{\text{final}} - \alpha_{\text{init}}) \cdot \sigma(\gamma \cdot (p - 0.5))$$

where $p \in [0,1]$ is training progress after a warmup phase of 2 epochs, σ is the logistic sigmoid, and $\gamma = 2.0$ controls ramp steepness. An optional adaptive variant monitors loss dynamics: if the adversary is winning (loss decreasing), alpha increases; if hate loss rises, alpha decreases.

3.4. Mutual Information Minimisation via MINE

The adversarial loss only removes identity features that the *current* adversary can detect. To catch residual identity signals, we add direct mutual information minimisation using the Mutual Information Neural Estimator (MINE) (Belghazi et al., 2018):

$$I(X; Y) \geq \sup_{T_\theta} \mathbb{E}_P[T_\theta] - \log(\mathbb{E}_Q[\exp(T_\theta)])$$

where P is the joint distribution of (representation, author) pairs and Q is the product of marginals (shuffled author labels). The training becomes a three-player game:

Encoder

$$\text{minimises } \mathcal{L}_{\text{hate}} + \lambda_{\text{dis}} \cdot \mathcal{L}_{\text{adv}} + \lambda_{\text{mim}} \cdot \hat{I}$$

Adversary

$$\text{minimises } \mathcal{L}_{\text{author}} \text{ (gradient reversed through GRL)}$$

MINE network

$$\text{maximises } \hat{I}(\text{repr; author})$$

3.5. Combined Training Objective

The full objective is:

$$\mathcal{L}_{\text{total}} = \underbrace{\mathcal{L}_{\text{hate}}(\hat{y}, y)}_{\text{cross-entropy}} + \underbrace{\lambda_{\text{dis}} \cdot \mathcal{L}_{\text{author}}(\hat{a}, a)}_{\text{GRL-reversed}} + \underbrace{\lambda_{\text{mim}} \cdot \hat{I}(\text{repr; author})}_{\text{MINE}} + \underbrace{\lambda_{\text{orth}} \cdot \mathcal{L}_{\text{orth}}}_{\text{cosine similarity}}$$

The orthogonality term $\mathcal{L}_{\text{orth}}$ encourages hate-relevant and identity-relevant features to occupy orthogonal subspaces, computed as the absolute cosine similarity between the pre-classification embeddings of the hate and pooler-level adversary branches.

3.6. Differential Privacy via Opacus

We use Opacus (Yousefpour et al., 2021) with ghost clipping for memory-efficient per-sample gradient computation. At each step:

Compute per-sample gradients via ghost clipping

Clip each gradient to L2 norm $\leq C$ ($C = 1.0$)

Add Gaussian noise $\mathcal{N}(0, \sigma^2 C^2 I)$

Average and update parameters

Track cumulative ϵ via Rényi DP accounting

Privacy accounting uses Rényi differential privacy (RDP) with (ϵ, δ) conversion implemented by Opacus

(Yousefpour et al., 2021). We evaluate across

$\epsilon \in \{1, 2, 4, 8, 16, 32\}$ to characterise the full trade-off surface.

Per-layer adaptive clipping (experimental).

The reference implementation includes a

PerLayerDPAdapter that estimates per-parameter gradient variance during warmup and computes layer-specific clip factors. This adapter provides an *interface* for per-layer clipping; the actual integration with

Opacus’s GradSampleModule is not yet fully implemented in the current training path. The training code falls back to uniform global clipping if per-layer setup fails, and the utility benefit of per-layer clipping (hypothesised to improve F1 by 0.5–1.5% at the same ϵ) remains unverified. We describe the adapter here to document the intended design path and to encourage community contributions to the Opacus integration.

4. VALIDATION SETUP

4.1. What the Harness Checks

The validation harness, implemented as a pytest suite with 45+ tests and a standalone smoke test, covers five layers:

Shape correctness (9 tests).

Verifies that all component outputs match their expected shapes at batch sizes $\{1, 2, 4, 8\}$ and sequence lengths $\{16, 32, 64\}$. Includes tests for inference-only mode, training mode with all labels, representation extraction, per-level MLAD outputs, and the MINE estimator.

Gradient flow (8 tests).

Confirms that every trainable parameter—encoder, hate classifier, adversary MLPs, MINE network—receives a non-null, finite gradient after backward. Includes tests for GRL gradient sign reversal (forward identity, backward $-\alpha \cdot \text{grad}$).

Numerical stability (6 tests).

Verifies finite outputs in bfloat16, robustness to extreme token indices (near vocabulary maximum), all-padding attention masks, single-token inputs, and duplicate sequences. Ensures no NaN or Inf values in gradients.

Privacy attack suite.

Four attack families implemented in the evaluation module:

Membership inference: shadow-model binary classifier on prediction confidence/entropy features, plus optimal-threshold approach on loss values.

Attribute inference: logistic regression on pooled representations to predict pseudo-author identity.

Stylometry (raw text): RandomForest on 80+ stylistic features (character n-grams, function word frequencies, readability scores) establishing the baseline re-identification risk.

Stylometry (model reps): logistic regression on CLS pooled representations, measuring residual identity leakage after disentanglement.

Rights-based architecture invariants (4 tests).

Verifies that representations are *never* returned in the default inference output, that explicit flags must be set to extract them, and that the inference engine’s public API exposes only hate probabilities.

4.2. What the Harness Does Not Check

The harness does *not* include:

End-to-end training on real datasets (Jigsaw Toxic Comment, HateXplain). These require GPU hours and are the subject of future work.

Competitive benchmarking against prior HSD methods on standard test splits. No accuracy, perplexity, or F1 scores on named real-world datasets are reported.

Multi-seed statistical significance testing. All tests use a single fixed seed (42).

Multilingual or cross-domain evaluation.

4.3. Compute Environment and Implementation Notes

All validation tests run on CPU with synthetic random data. The smoke test completes in approximately 2 minutes on a standard workstation without GPU. The full pytest suite completes in approximately 5 minutes.

Reference model vs. validation stub.

The architecture description and parameter counts in this paper refer to the *target* reference model (PrivHSDModelV2 with ALBERT-base-v2 backbone, approximately 15.3M parameters). The harness results reported in Section 5 were obtained on this same reference model but on a *synthetic validation path*: the transformer backbone is loaded from HuggingFace and used for shape verification only (no backpropagation through the full pretrained weights for DP-SGD; that requires full experiment execution). Parameter counts are measured from the actual model instantiation.

5. RESULTS

We report results from the validation harness and synthetic benchmarks. All results are based on the reference implementation with ALBERT-base-v2 backbone ($d_{\text{model}} = 768$, 12 layers, 12 heads). **No**

trained-model results on real datasets are available—all empirical claims about model utility on hate speech detection tasks require full experiment execution.

Three distinct kinds of evidence should be distinguished when interpreting this section. (i) Implementation correctness - that the code computes what the architecture specifies (output shapes, gradient flow, gradient-reversal sign, numerical stability) - is what the harness establishes. (ii) Formal privacy guarantees - the (epsilon, delta) bound provided by DP-SGD - follow from the mechanism and its accounting once the DP configuration is fixed, independently of any experiment. (iii) Empirical privacy protection - the measured success of membership-inference, attribute-inference and stylometry attacks against a trained model - is not evaluated here and can diverge from the nominal DP guarantee: recent work shows that models with the same (epsilon, delta) can exhibit substantially different empirical vulnerability depending on architecture, data and hyperparameters. The results below concern only (i).

5.1. Shape and Structural Properties

Table 1 summarises the shape invariants verified by the harness. All tests pass across the specified input dimensions.

Table 1. Shape invariants verified by the validation harness. All tests pass for batch sizes $B \in \{1, 2, 4, 8\}$ and sequence lengths $T \in \{16, 32, 64\}$.

Output	Shape	Status
Hate logits (inference)	(B, 2)	Pass
Hate probabilities	(B, 2)	Pass
Loss (training)	scalar	Pass
Per-level author logits (pooler)	(B, N_{authors})	Pass
Per-level author logits (token)	(B, N_{authors})	Pass
Per-level author logits (head)	(B, N_{authors})	Pass
Pooled representations	(B, d_{model})	Pass
MINE MI estimate	scalar	Pass

5.2. Gradient Flow and GRL Correctness

Table 2 confirms that all model components receive correct gradients. The GRL is verified to reverse gradient sign (output $-\alpha \cdot \text{grad}$) while maintaining identity in the forward pass, matching the specification of Ganin and Lempitsky (Ganin and Lempitsky, 2015).

Table 2. Gradient flow verification results. “All params” refers to every trainable parameter receiving a non-null gradient.

Test	Result	No. of params verified
All params receive gradient	Pass	100%
Encoder backbone	Pass	All encoder
Hate classifier head	Pass	All classifier
Adversary MLPs (3 levels)	Pass	All adversary
MINE network (maximise phase)	Pass	All MINE
MINE network (minimise phase)	Pass	All MINE
GRL forward identity	Pass	—
GRL backward sign reversal	Pass	—
No NaN/Inf in any gradient	Pass	All

5.3. Parameter Counts

The complete PrivHSD model has approximately 15.3M parameters with ALBERT-base-v2 (the backbone contributes ~ 12 M; the hate classifier, three adversary MLPs, and MINE network add ~ 3.3 M).
 ALBERT-base-v2 backbone: 11.7M
 Hate classification head: 1.2M

MLAD adversary MLPs (3 $\times$): 2.2M

MINE network: ~ 0.1 M

Other (scheduler, embeddings): ~ 0.1 M

For comparison, a standard BERT-base classifier (Devlin et al., 2019) has approximately 110M parameters, nearly an order of magnitude larger. The parameter efficiency of ALBERT (Lan et al., 2020) under DP is a key design motivation (Bueno Júnior et al., 2025).

5.4. Numerical Stability

All numerical stability tests pass (Table 3), confirming bf16-safety and robustness to edge-case inputs.

Table 3. Numerical stability test results.

Test	Result
bfloat16 forward (finite logits and probs)	Pass
Extreme token indices (vocab max)	Pass
All-padding attention mask	Pass
Single-token input	Pass
Duplicate sequences in batch	Pass
Gradient boundary values	Pass

5.5. Privacy Attack Module Functionality

All privacy attack modules initialise correctly and produce valid output on synthetic data (Table 4). These

tests verify *implementation correctness*, not empirical privacy guarantees.

Table 4. Privacy attack module verification on synthetic data. All values verify implementation correctness, not empirical privacy efficacy (requires trained models).

Attack	Metric	Status
MIA (shadow model)	AUC $\in [0,1]$	Pass
MIA (threshold)	Accuracy	Pass
Attribute inference	Accuracy $\in [0,1]$	Pass
Stylometry (raw text)	80+ features finite	Pass
Stylometry (model reps)	Accuracy $\in [0,1]$	Pass
Representation entropy	Finite	Pass
k-anonymity fraction	$\in [0,1]$	Pass
Privacy leakage score	Non-negative	Pass

5.6. Ablation Framework

The ablation runner supports 11 single-field config variants (Table 5), each changing exactly one field in the PrivHSDConfig dataclass. All 11 configurations initialise and pass the smoke test.

Table 5. Ablation configurations supported by the framework. $\Delta F1$ and $\Delta \epsilon$ require trained-model evaluation. Expected trends are shown in parentheses.

Ablation	Config change	$\Delta F1$
#1 No adversarial	$\lambda_{dis} = 0$	—
#2 No DP-SGD	dp_enabled = False	—
#3 Single-level adv	levels = (pooler,)	—
#4 No MIM	$\lambda_{mim} = 0$	—
#5 Linear α	schedule = linear	—
#6 No orthogonality	$\lambda_{orth} = 0$	—
#7 Uniform clipping*	per_layer = False	—
#8 No augmentation	augment = None	—
#9 RoBERTa backbone	model_type = roberta	—
#10 Strong privacy	$\epsilon = 1.0$	—

#11 Weak privacy	$\epsilon = 32.0$	—
------------------	-------------------	---

*Ablation #7 (uniform clipping) tests the config flag; per-layer clipping is an experimental adapter and currently falls back to uniform clipping in the training path. The utility benefit of per-layer clipping is unverified.

6. DISCUSSION

6.1. Structural Properties of the Architecture

The validation harness confirms several structural properties of the PrivHSD architecture that are relevant independently of empirical performance:

Shape invariants guarantee composability.

Every output shape is explicitly documented and tested. This makes the architecture straightforward to compose with other components—for example, adding a federated learning wrapper, replacing the backbone with XLM-RoBERTa for multilingual deployment, or attaching a different adversary architecture.

bfloat16 safety enables efficient training.

All components produce finite outputs in bfloat16, which is essential for memory-efficient training under DP-SGD (where per-sample gradient computation already strains GPU memory). This was not guaranteed a priori: LayerNorm operations in the adversary MLPs required careful dtype casting to avoid numerical issues.

Parameter efficiency favours ALBERT under DP.

The total parameter count (15.3M) is an order of magnitude smaller than comparable RoBERTa-based HSD classifiers (125M+). The parameter efficiency of ALBERT is consistent with the smaller DP utility loss reported for ALBERT by Bueno Júnior et al. (2025) in their federated experiments.

Rights-based design principles.

The architecture is designed to enforce data minimisation (GDPR Art. 5(1)(c)) and purpose limitation (Art. 5(1)(b)) at the representation level: adversarial training is *intended* to make pooled representations identity-agnostic, and the inference API never exposes internal states by default. We caution that the empirical effectiveness of the adversarial disentanglement against re-identification attacks—particularly under white-box access—has not yet been validated with trained models on real data. The architecture provides the structural foundation for privacy by design; quantifying its real-world privacy guarantees requires the trained-model evaluation described in future work.

6.2. Limitations

We acknowledge several limitations of the current artifact.

No trained-model evaluation on real datasets.

The most significant limitation is the absence of empirical results on standard HSD benchmarks (Jigsaw Toxic Comment, HateXplain). All “results” in Section 5 are shape/gradient/numerics tests and synthetic micro-benchmarks. The four hypotheses (MLAD reduces stylometry by >15% with <3% F1 loss; DP + adversarial Pareto-dominates either alone; ALBERT > RoBERTa under DP; privacy augmentation improves the frontier) remain unverified. Previous work (Bueno Júnior et al., 2025) provides grounding for the ALBERT hypothesis, but the combined mechanism claims require dedicated experimentation.

Synthetic author labels.

Our identity disentanglement uses heuristic pseudo-author labels derived from stylistic feature hashing rather than ground-truth authorship data. Validation on datasets with real author metadata is needed to confirm that the synthetic proxy provides sufficient signal.

Computational cost.

DP-SGD with ghost clipping increases training time by 3–5~~x~~ compared to non-DP training. The full experiment suite (20+ configs ~~x~~ 3 seeds) is estimated to require several GPU-days.

English-only scope.

The current prototype validates on English data. The architecture supports XLM-RoBERTa for multilingual extension, but no multilingual data pipeline is implemented.

MINE network stability.

The three-player game (encoder, adversary, MINE) lacks theoretical convergence guarantees. On synthetic data the MINE estimate remains bounded, but real-data training may require additional stabilisation (e.g., spectral normalisation, gradient clipping on the MINE network).

Per-layer DP clipping not fully integrated.

The per-layer clipping adapter is implemented as an experimental interface but not fully integrated with Opacus’s training pipeline. All DP training currently uses uniform global clipping.

No empirical privacy evaluation. Passing the attack-suite unit tests confirms that the membership-inference, attribute-inference and stylometry modules run and produce well-formed output on synthetic data; it does not show that identity information has been removed from learned representations. Establishing that requires training PrivHSD together with non-private, DP-only, adversarial-only and combined variants on real HSD datasets (for example Jigsaw Toxic Comment and HateXplain) and reporting detection F1/AUC alongside attack success for each variant, across multiple privacy budgets and multiple seeds with confidence intervals.

Meaning of the mutual-information objective. A reduction in the MINE-estimated mutual information between representations and author labels does not by itself guarantee a corresponding reduction in real re-identification risk: MINE provides a lower-bound

estimate whose tightness and stability depend on the estimator network and the training dynamics.

7. CONCLUSION

PrivHSD is a reference architecture and validation harness for privacy-preserving hate speech detection. It combines multi-level adversarial disentanglement, adaptive gradient reversal scheduling, mutual information minimisation, and differential privacy, and passes 45+ structural tests on synthetic data.

This work reports no trained-model results on real datasets. It provides a reproducible design for future evaluation of the privacy-utility trade-off.

Future work will evaluate the architecture on real-author and multilingual data, extend federated training and per-layer clipping, and compare the proposed components empirically on standard HSD benchmarks, reporting detection utility and residual identity leakage with multi-seed confidence intervals and stronger baselines.

The source code and validation harness accompany this artifact to support reproducible follow-up work.

8. APPENDIX A. MODEL CONFIGURATION DATACLASS

Table 6 lists the key hyperparameter groups of the PrivHSDConfig dataclass. The full dataclass contains 60+ fields; only the most architecturally significant are shown here.

Table 6. Key hyperparameter groups in PrivHSDConfig.

Group	Example field	Default
Transformer backbone	model_name	albert-base-v2
Hate head	classifier_num_layers	2
MLAD	adversarial_levels	(pooler, token, head)
AGRS	alpha_schedule	sigmoid
MIM	mim_weight	0.1
DP-SGD	target_epsilon	8.0
Orthogonality	orthogonality_weight	0.05
Training	batch_size	16

8.1. DP-SGD Privacy Accounting Details

Privacy budget is tracked via Rényi differential privacy (RDP) accounting implemented by Opacus (Yousefpour et al., 2021), using the analytic Fourier

accountant for (ϵ, δ) conversion and privacy amplification via Poisson subsampling. Implementation details are provided by the Opacus documentation and accountant.

8.2. Reproducing the Validation Suite

All validation results in this paper can be reproduced with:

```
# Quick smoke test (~2 minutes, CPU, no GPU required)
cd coder && python smoke_test.py --device cpu
```

```
# Full test suite (~5 minutes)
cd validator && pytest test_model.py -v
```

```
# Include benchmark tests
cd validator && pytest test_model.py -v --run-benchmarks
```

```
# Ablation runner (dry run — shows all 11 configs)
cd validator && python ablation_runner.py --dry-run
```

The full experiment suite (requiring GPU and dataset downloads) is:

```
# Single ablation experiment
cd coder && python run_experiments.py --ablation 1
```

```
# Run all 11 ablations across 3 seeds
cd validator && python ablation_runner.py --ablation all
```

REFERENCES

- Abadi, M. et al. (2016) ‘Deep learning with differential privacy’, in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. ACM, pp. 308–318.
- Abbasi, A. and Chen, H. (2008) ‘Writeprints: A stylometric approach to identity-level identification and similarity detection in cyberspace’, *ACM Transactions on Information Systems*, 26(2), pp. 7:1–7:29. DOI: 10.1145/1344411.1344413.
- Belghazi, M.I. et al. (2018) ‘MINE: Mutual information neural estimation’, in *Proceedings of the 35th international conference on machine learning*. PMLR, pp. 531–540.
- de Souza Bueno Júnior, I., Ye, H., Wisiosek, A. and Schütze, H. (2025) ‘Privacy-preserving federated learning for hate speech detection’, *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 4: Student Research Workshop)*, pp. 129–141. DOI: 10.18653/v1/2025.naacl-srw.13.

- Cavoukian, A. (2009) 'Privacy by design: The 7 foundational principles'. Information; Privacy Commissioner of Ontario.
- Devlin, J., Chang, M.-W., Lee, K. and Toutanova, K. (2019) 'BERT: Pre-training of deep bidirectional transformers for language understanding', Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 4171–4186. DOI: 10.18653/v1/N19-1423.
- Faustini, P., Tonni, Shakila Mahjabin, McIver, A., Xu, Q. and Dras, M. (2025) 'IDT: Dual-task adversarial rewriting for attribute anonymization', Computational Linguistics, 51(4), pp. 1151–1189. DOI: 10.1162/coli.a.17.
- Fredrikson, M., Jha, S. and Ristenpart, T. (2015) 'Model inversion attacks that exploit confidence information and basic countermeasures', in *Proceedings of the IEEE symposium on security and privacy*. IEEE, pp. 132–147.
- Frikha, A., Walha, N., Nakka, Krishna Kanth, Mendes, R., Jiang, X. and Zhou, Xuebing (2025) 'IncogniText: Privacy-enhancing conditional text anonymization via LLM-based private attribute randomization', Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pp. 2490–2501. DOI: 10.18653/v1/2025.ijcnlp-long.134.
- Ganin, Y. and Lempitsky, V. (2015) 'Unsupervised domain adaptation by backpropagation', in *Proceedings of the 32nd international conference on machine learning*. PMLR, pp. 1180–1189.
- Lan, Z. et al. (2020) 'ALBERT: A lite BERT for self-supervised learning of language representations', in *Proceedings of the 8th international conference on learning representations*.
- Makroo, Owais, Kasa, Siva Rajesh, Roychowdhury, Sumegh, Gupta, Karan, Pattisapu, Nikhil, Kasa, Santhosh and Negi, Sumit (2025) 'The hidden cost of modeling P(X): Vulnerability to membership inference attacks in generative text classifiers'. arXiv preprint arXiv:2510.16122. Available at: <https://arxiv.org/abs/2510.16122>.
- Narayanan, A. et al. (2012) 'On the feasibility of internet-scale author identification', Proceedings of the IEEE Symposium on Security and Privacy, pp. 300–314. DOI: 10.1109/SP.2012.46.
- Shokri, R. et al. (2017) 'Membership inference attacks against machine learning models', Proceedings of the IEEE Symposium on Security and Privacy, pp. 3–18. DOI: 10.1109/SP.2017.41.
- Yeom, S. et al. (2018) 'Privacy risk in machine learning: Analyzing the connection to overfitting', Proceedings of the IEEE Computer Security Foundations Symposium, pp. 268–282. DOI: 10.1109/CSF.2018.00027.
- Yousefpour, A., Shilov, I., Sablayrolles, A., Testuggine, D., Prasad, K., Malek, M., Nguyen, John et al. (2021) 'Opacus: User-friendly differential privacy library in PyTorch'. arXiv preprint arXiv:2109.12298. Available at: <https://arxiv.org/abs/2109.12298>.