Aggregation-Aware Synthetic Text Generation Against Authorship Re-Identification

Qian Ma, Anna Squicciarini, Sarah Rajtmajer
Information Sciences and Technology
The Pennsylvania State University
{qfm5033, acs20, smr48}@psu.edu

Abstract

Online users often release multiple texts under the same identity, giving attackers an author profile that can reveal more than any single text. Existing authorship obfuscation methods optimize privacy independently for each document, leaving them blind to cross-document correlations that make aggregation dangerous. We propose Aggregation-Aware Synthetic Text Generation (AAST), a framework that addresses this gap by jointly selecting synthetic texts at the bundle level rather than optimizing each text in isolation. AAST targets attribution and verification attacks, including cross-genre settings where attacker references come from a genre not observed during generation or selection. Experiments across same-genre, cross-genre, neural, and independent non-neural stylometric attacks show that AAST lowers account-level linkability as bundle size grows, while preserving semantic quality, linguistic acceptability, and sentiment alignment.

1 Introduction

Pseudonymous writing is central to online communities, blogs, and social media platforms (Leavitt, 2015). It lets users participate publicly while separating posts from legal identity, which matters for sensitive communities and for users managing identity boundaries online (Triggs et al., 2021; Bao and Carpuat, 2024; Xie et al., 2024). Yet pseudonyms hide legal names, but don't protect against linkability. Authorship analysis has long shown that texts carry stable signals through word choice, syntax, punctuation, function words, and other habits of expression (Stamatatos, 2009; Neal et al., 2017). Neural authorship representations make this risk more practical by learning embeddings for attribution and verification (Rivera-Soto et al., 2021). Recent work further shows that large language models (LLMs) can expose identity related cues from writing style (Nguyen et al., 2025), infer authors in open world settings (Tan et al., 2025), and deanonymize pseudonymous online profiles from unstructured text at scale (Lermen et al., 2026).

A growing body of work studies authorship obfuscation and privacy-preserving rewriting. Recent approaches use LLMs, decoding algorithms, privacy-aware abstraction, and anonymization objectives (Fisher et al., 2024a; Huang et al., 2024b; Dou et al., 2024; Huang et al., 2025). A central challenge is balancing privacy with utility; minor edits may still reveal authorial style, while aggressive rewriting can degrade meaning, fluency, or downstream usefulness (Yang et al., 2025).

Notably, nearly all existing authorship obfuscation methods optimize privacy independently for each document, despite the fact that real adversaries aggregate multiple posts into account-level author profiles. Even individually well-obfuscated texts can remain linkable when released together, because consistent stylistic features across a released collection leave account-level signals that no per-document method is designed to suppress. This mismatch undermines the performance of existing methods. In a Reddit authorship attribution setting, for example, top-eight author identification rises from 2% with one unmodified comment to 93% with 16 comments (Bao and Carpuat, 2024). Recent large scale deanonymization work further shows that LLMs can link pseudonymous profiles by extracting and matching signals across user histories (Lermen et al., 2026). Cross-genre authorship analysis adds another concern, since one person may write in different genres, such as social media posts and news articles (Rivera-Soto et al., 2021; Ma et al., 2025). Direct identifier masking is useful, but masked documents can remain vulnerable when attackers use retrieval and infilling (Pilán et al., 2022; Charpentier and Lison, 2025). These findings motivate defenses that consider the released collection at the author level, rather than each text in isolation.

We propose Aggregation-Aware Synthetic Text

Generation (AAST), a framework for reducing account-level authorship re-identification by weakening linkability across released *bundles*, where each bundle is a collection of texts attributed to the same account. Unlike prior approaches that rewrite texts independently, AAST jointly selects synthetic texts at the bundle level, targeting the aggregated author signals used in attribution and verification attacks. This shifts the method from optimizing isolated rewrites to reducing linkability across a released account history.

We evaluate AAST on same-genre and cross-genre datasets under authorship attribution and verification attacks. Same-genre evaluation uses the benchmark Blog Authorship Corpus (Schler et al., 2006) and our constructed Reddit authorship corpus. The complete dataset and protocol details are provided in §4.1. Cross-genre evaluation uses the CROSSNEWS authorship benchmark linking news articles and Twitter/X posts by the same authors (Ma et al., 2025). Compared with authorship obfuscation baselines (Bao and Carpuat, 2024; Fisher et al., 2024b), AAST reduces attribution and verification risk as bundle size grows while maintaining semantic and linguistic quality, and sentiment alignment.

Our contributions are:

- We formulate and study account-level authorship privacy under aggregation, where multiple released texts from the same account form an author profile. This shifts the privacy target from isolated text rewriting to reducing linkability across a released account history.
- We propose AAST, an aggregation-aware synthetic text generation framework that jointly selects released texts at the bundle level to reduce linkability within a user's own account history.
- We evaluate AAST across same-genre, cross-genre, neural, and independent non-neural stylometric attacks. Results show that AAST reduces account-level linkability as bundle size grows while maintaining semantic, linguistic, and sentiment utility.

All code is available at `https://github.com/masonmq/aast`.

2 Related Work

Authorship analysis and author re-identification. Authorship analysis is commonly studied through attribution and verification (Mosteller and Wallace, 1963; Holmes, 1998; Stamatatos, 2009). Two common tasks are *authorship attribution*, which identifies the author of a text from a candidate set, and *authorship verification*, which decides whether two texts were written by the same author (Keswani et al., 2016; Stamatatos et al., 2016; Karadzhov et al., 2017; Shetty et al., 2018). Recent work argues that the distinction between these tasks matters for realistic evaluation, especially when the candidate set, domain, or genre changes (Bevendorff et al., 2025; Ma et al., 2025).

Classical authorship models rely on stylometric features such as character n-grams, function words, punctuation, and compression based scores (Teahan and Harper, 2003; Koppel and Schler, 2004; Seidman, 2013). Neural models later learned author representations directly from text, with LUAR using contrastive learning to produce author embeddings for verification and attribution (Rivera-Soto et al., 2021). PAN shared tasks provide widely used verification settings and metrics (Stamatatos et al., 2022; Bevendorff et al., 2025). CROSSNEWS shows that author signals do not always transfer cleanly across genres (Ma et al., 2025).

Authorship obfuscation and text privatization. Authorship obfuscation rewrites text so that the original author becomes harder to identify while the rewritten text remains useful (Potthast et al., 2016; Mahmood et al., 2019; Altakrori et al., 2022). It differs from ordinary paraphrasing, which may preserve author style, and from style transfer, which usually assumes a target style (Mireshghallah and Berg-Kirkpatrick, 2021; Fisher et al., 2024b). Recent work studies LLMs for privacy preserving rewriting, including text abstraction, text anonymization, authorship privacy, and privacy attack modeling (Dou et al., 2024; Nguyen et al., 2025; Staab et al., 2025). LLM based paraphrasing and style imitation can reduce the robustness of authorship verification systems (Alperin et al., 2025). KiP fine tunes a language model with reinforcement learning to balance privacy, semantic soundness, and fluency, and evaluates against LUAR attribution and PAN style verification attacks (Bao and Carpuat, 2024). JAMDEC uses small language models with constrained diverse decoding, over generation, and filtering (Fisher et al., 2024b). These methods are closest to AAST because they target author style obfuscation.

Linkage risk and aggregation. Privacy risk arises not only from explicit identifiers, but also

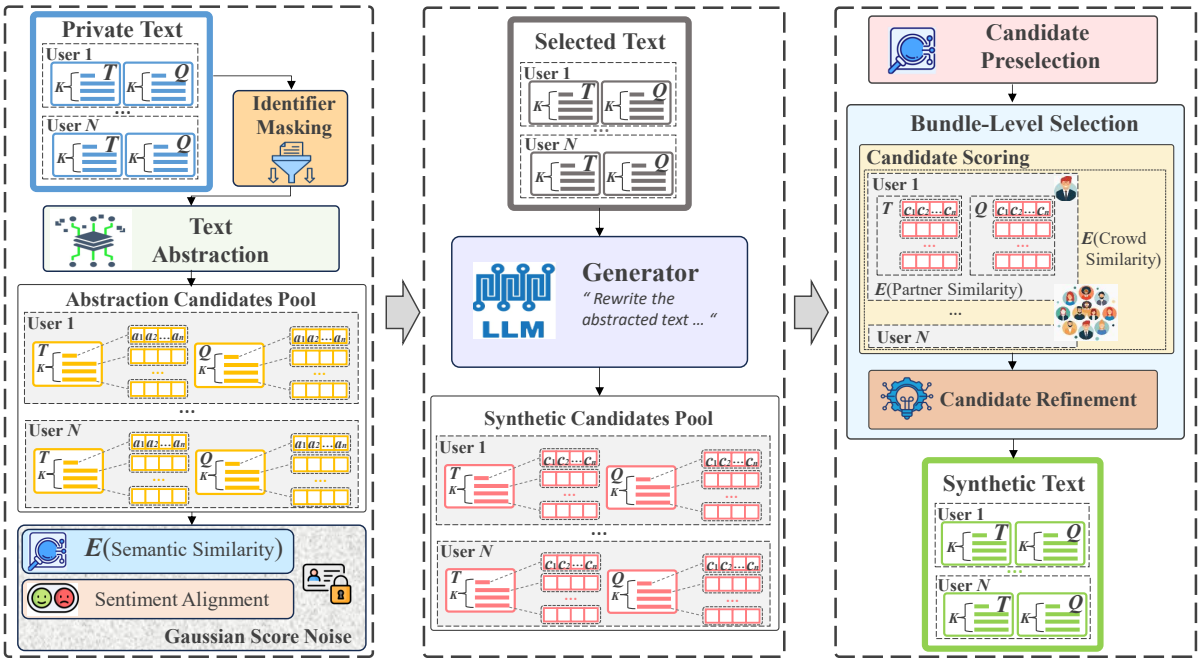


Figure 1: Overview of the AAST pipeline for aggregation-aware synthetic text generation.

from linking released data with other information. Recent work systematizes dataset privacy attacks through linkage, where re-identification and attribute inference are treated as different forms of information gain (Powar and Beresford, 2023). Text de-identification work also shows that masking direct identifiers is not enough, since adversaries can use background knowledge, retrieval, and infilling to recover masked information (Pilán et al., 2022; Charpentier and Lison, 2025). Differential privacy has also been studied for text generation, including synthetic data from foundation model APIs (Yue et al., 2023; Xie et al., 2024). AAST includes a limited noisy abstraction selection analysis, but addresses aggregation by treating released account histories as the privacy target.

3 AAST Methodology

3.1 Threat Model

Each private text x is associated with a pseudonymous account $u \in \mathcal{U}$, and $\mathcal{S}$ contains synthetic texts $\tilde{x}$ generated for the same accounts. Even after privatization, synthetic texts may preserve writing patterns that can re-identify the account author when multiple texts are observed together.

We consider an adversary that links released synthetic texts at the account level. Our primary setting is same-genre aggregation, where the released synthetic corpus contains multiple texts generated from each pseudonymous account. The adversary compares synthetic bundles and performs either an *attribution attack*, ranking candidate accounts for a synthetic bundle, or a *verification attack*, deciding whether two synthetic bundles come from the same account. Larger K represents a stronger aggregation threat because the adversary observes a larger released account history.

We also evaluate a cross-genre reference setting, where the adversary has access to external reference texts from another genre written by candidate authors and attempts to match them against the released synthetic corpus. AAST generates and selects synthetic texts from one genre only, and does not observe cross-genre reference texts during generation or selection. This setting tests whether a released synthetic text remains linkable to other writings by the same author.

Our goal is to lower attribution and verification success while preserving text utility.

3.2 System Overview

AAST reduces account-level authorship re-identification by weakening linkability at the bundle level (Figure 1). Following, we describe preprocessing and four primary steps: *private text abstraction*; *synthetic text generation*; *candidate preselection*; and, *bundle-level selection*.

Algorithm 1 AAST

Input: private bundles $\{X_u^Q, X_u^T\}_{u\in\mathcal{U}}$, abstraction model Ψ, abstraction selector S, generator G, episode embedder E, bundle size K
Parameters: candidates per text C, weight α, margin τ
Output: synthetic bundles $\{\tilde{S}_u^Q, \tilde{S}_u^T\}_{u\in\mathcal{U}}$

for all $u \in \mathcal{U}$, $b \in \{Q, T\}$, and $k \in \{1, \ldots, K\}$ **do**
  $\mathcal{A}(x_{u,k}^b) \leftarrow \Psi(x_{u,k}^b)$
  $a_{u,k}^b \leftarrow S\big(x_{u,k}^b, \mathcal{A}(x_{u,k}^b)\big)$
  $\mathcal{C}(a_{u,k}^b) \leftarrow G(a_{u,k}^b)$
  $\mathcal{C}'(a_{u,k}^b) \leftarrow \text{Can\_Preselection}\big(x_{u,k}^b, \mathcal{C}(a_{u,k}^b)\big)$
  Initialize $\tilde{x}_{u,k}^b$ with one candidate from $\mathcal{C}'(a_{u,k}^b)$
end for
for all $u \in \mathcal{U}$ **do**
  Compute current bundle embeddings $v_u^Q \leftarrow E(\tilde{S}_u^Q)$ and $v_u^T \leftarrow E(\tilde{S}_u^T)$
end for
for all $u \in \mathcal{U}$ **do**
  Build $\mathcal{V}_{-u}$ from the current bundle embeddings of other users
  for all $b \in \{Q, T\}$ and $k \in \{1, \ldots, K\}$ **do**
    Let $\bar{b}$ denote the partner side of the same user
    For each $c \in \mathcal{C}'(a_{u,k}^b)$, form the candidate bundle $\mathcal{S}_u^b(c)$, compute its episode embedding $v_u^b(c)$, and evaluate the bundle objective $\mathcal{L}(c)$
    $\mathcal{R} \leftarrow \{c \in \mathcal{C}'(a_{u,k}^b) : \mathcal{L}(c) \le \min_{c'\in\mathcal{C}'(a_{u,k}^b)} \mathcal{L}(c') + \tau\}$
    Compute $s_{\text{text}}(c)$ for each $c \in \mathcal{R}$
    $c^* \leftarrow \arg\min_{c\in\mathcal{R}} s_{\text{text}}(c)$
    Update $\tilde{x}_{u,k}^b \leftarrow c^*$ and $v_u^b \leftarrow v_u^b(c^*)$
  end for
end for
return $\{\tilde{S}_u^Q, \tilde{S}_u^T\}_{u\in\mathcal{U}}$

3.3 Preprocessing

Let the private corpus $\mathcal{X}$ be grouped by pseudonymous account $u \in \mathcal{U}$. AAST includes an optional IDENTIFIER_MASKING module before abstraction, which replaces structured identifiers and recognizable surface cues with placeholders such as `[USER]` and `[EMAIL]`. We enable it for cross-genre settings, where tweets often contain handles, URLs, hashtags, and other surface identifiers, but disable it for same-genre Reddit and Blog settings because masking can remove useful semantic content.

The attacker aggregates K texts at a time, where K denotes the number of texts from the same user grouped into each bundle. We organize each user's private texts into a *query* bundle $\mathcal{X}_u^Q = (x_{u,1}^Q, \ldots, x_{u,K}^Q)$ and a *target* bundle $\mathcal{X}_u^T = (x_{u,1}^T, \ldots, x_{u,K}^T)$, and the synthetic corpus keeps the same structure. We use an episode embedding function $E(\cdot)$ to map each bundle to a single vector, matching the attacker's aggregation behavior by scoring bundles instead of isolated texts. The AAST algorithm is provided in Algorithm 1.

3.4 Private Text Abstraction

3.4.1 Abstraction candidate generation

Prior work shows that rephrasing sensitive disclosures into less specific terms can reduce privacy risk while preserving utility (Dou et al., 2024). However, using private text for synthetic data generation can still preserve private and authorship related signals. To address this risk, we employ an abstraction model Ψ to transform each private sample $x \in \{x^{(1)}, \ldots, x^{(N)}\}$ into a set of h abstracted candidates, $\mathcal{A}(x) = \{a_1, \ldots, a_h\}$, where N is the number of private samples (Ma and Rajtmajer, 2026). The abstraction candidates preserve the core meaning and sentiment of the private text while reducing direct correspondence with its surface form. Further implementation details are provided in Appendix A.1.

3.4.2 Noisy abstraction selection

We use noisy abstraction selection as a randomized scoring step before candidate generation. The formal privacy statement is conditional and limited to the noisy score selection step; it does not cover candidate generation, the selected abstraction text, or the final synthetic corpus. The algorithm and proof are provided in Appendix A.2 and A.3.

3.5 Synthetic Text Generation

For each selected abstraction $a(x)$, we prompt an LLM to generate a pool of C synthetic candidates: $\mathcal{C}(a(x)) = \{c^{(1)}, \ldots, c^{(C)}\}$. Candidate generation is performed for each slot $x_{u,k}^b$, where $k \in \{1, \ldots, K\}$ and $b \in \{Q, T\}$. The prompt asks the LLM to preserve meaning while varying writing style; details are provided in Appendix B.1.

3.6 Candidate Preselection

We then apply Candidate Preselection to each slot's generated pool by using the private source text for that slot as a reference. AAST encodes the source text and its generated candidates in a shared stylistic embedding space, ranks candidates by similarity to the source, and retains the half with the lowest similarity. The remaining candidates enter *Bundle-Level Selection*.

3.7 Bundle-Level Selection

3.7.1 Bundle-level candidate scoring

We select exactly one candidate per slot, but score candidates at the bundle level. Fix a user u, a bundle side $b \in \{Q, T\}$, and a position k. After Candidate Preselection, let $\mathcal{C}'(a(x_{u,k}^b)) \subseteq \mathcal{C}(a(x_{u,k}^b))$

denote the reduced candidate pool for slot k. During selection, all slots have temporary synthetic choices, and candidates for slot k are scored by replacing the current choice at that slot. Let $\tilde{S}^b_u = (\tilde{s}^b_{u,1}, \ldots, \tilde{s}^b_{u,K})$ denote the current temporary synthetic bundle for user u on side b.

For any candidate $c \in \mathcal{C}'(a(x^b_{u,k}))$, we form a candidate bundle by replacing only slot k: $\tilde{S}^b_u(c) = (\tilde{s}^b_{u,1}, \ldots, \tilde{s}^b_{u,k-1}, c, \tilde{s}^b_{u,k+1}, \ldots, \tilde{s}^b_{u,K})$, and compute its episode embedding $v^b_u(c) = E(\tilde{S}^b_u(c))$.

We score c using two bundle-level terms.

(1) Partner similarity. Let $\bar{b}$ denote the other side of the same user, with $\bar{Q} = T$ and $\bar{T} = Q$. We measure similarity between the candidate bundle and the partner bundle from the same user:

$$s_{\text{partner}}(c) = \cos\big(v^b_u(c),\, v^{\bar{b}}_u\big). \tag{1}$$

(2) Crowd similarity. Let $\mathcal{V}_{-u}$ be a pool of current bundle embeddings from other users, and let m be the number of nearest other user bundles:

$$s_{\text{crowd}}(c) = \text{mean}\Big(\text{Top-}m\big\{\cos(v^b_u(c), v) : v \in \mathcal{V}_{-u}\big\}\Big). \tag{2}$$

These two terms define the bundle-level objective

$$\mathcal{L}(c) = \alpha\, s_{\text{partner}}(c) - (1-\alpha)\, s_{\text{crowd}}(c), \tag{3}$$

where lower is better. This objective favors candidates that reduce similarity between the two bundles of the same user while keeping the selected bundle close to nearby bundles from other users.

3.7.2 Candidate refinement

We then refine among candidates that remain competitive under the bundle embedding objective. Let

$$\mathcal{R} = \Big\{c \in \mathcal{C}'\big(a(x^b_{u,k})\big) : \mathcal{L}(c) \le \min_{c' \in \mathcal{C}'(a(x^b_{u,k}))} \mathcal{L}(c') + \tau\Big\}, \tag{4}$$

where τ is a small margin that defines a near optimal candidate set under the objective in §3.7.1.

Within $\mathcal{R}$, we use a character n-gram bundle similarity surrogate $g(\cdot,\cdot)$ to compare each candidate bundle $\tilde{S}^b_u(c)$ with the partner bundle $\tilde{S}^{\bar{b}}_u$. We choose the candidate with the lowest g score, update slot k, and refresh the corresponding bundle embedding. The dominant episode embedding cost of AAST is $\mathcal{O}\big(|\mathcal{U}| \cdot 2K \cdot C\big)$. Further complexity analysis is provided in Appendix A.4.

4 Experimental Methodology

4.1 Datasets

4.1.1 Same-genre

Reddit Authorship Corpus. We build a Reddit authorship corpus from self-disclosure posts about financial hardship and poverty. We select subreddits related to economic difficulty and use keyword filtering to identify posts about financial struggle. The subreddit and keyword lists are provided in Appendix Table 21. The collected posts span January 1, 2011 to March 31, 2026 and contain 103,587 posts. This corpus supports aggregation study with up to 268 users, where each user has two bundles of up to K=16 posts.

Blog Authorship Corpus. The Blog Authorship Corpus (Schler et al., 2006) is a standard authorship benchmark. It contains 681,288 posts from 19,320 bloggers collected from Blogger.com in August 2004, with about 35 posts per user on average (Tatman, 2015). We exclude posts shorter than 60 words and set the largest bundle size to K=16, so each user can provide 16 posts for the query bundle and 16 posts for the target bundle.

For comparability across conditions and bundle sizes, we use 250 users in same-genre Reddit and Blog experiments, ensuring each user can form two bundles up to K=16, or $2K$=32 texts per user.

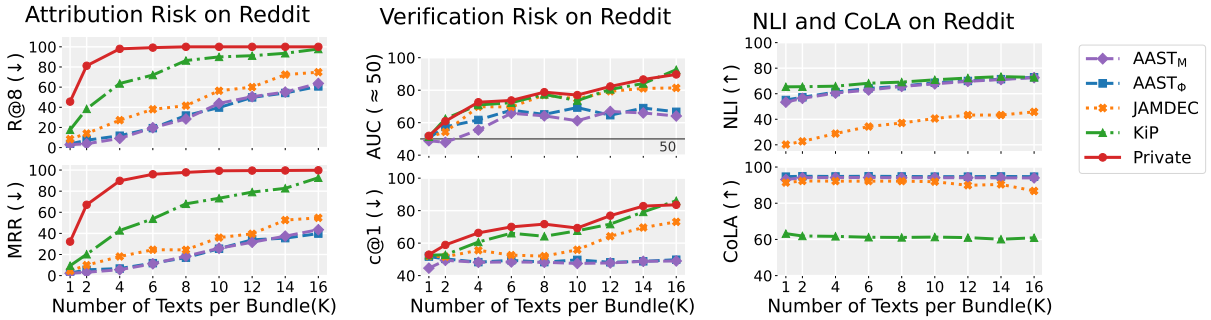
4.1.2 Cross-genre

CROSSNEWS. CROSSNEWS (Ma et al., 2025) is a cross-genre authorship benchmark linking news articles and Twitter/X posts by the same authors. We use its manually verified gold set, which contains 500 journalists from the New York Times, the Guardian, and the Times of India, with 100 articles and 100 tweets per author.

We evaluate two cross-genre settings, Article–Tweet and Tweet–Article. In Article–Tweet, AAST generates synthetic tweets from tweet inputs, while articles are used only as attacker references and are never observed by AAST during generation or selection. Tweet–Article is constructed analogously. This tests whether synthetic text in one genre remains linkable to reference text from another genre. Article–Tweet is expected to be harder because tweets are often short and metadata heavy.

4.2 Baselines

We compare AAST with KiP (Bao and Carpuat, 2024) and JAMDEC (Fisher et al., 2024b), two recent strong authorship obfuscation baselines. KiP fine tunes an LLM with reinforcement learning to hide authorial style while preserving text quality. JAMDEC uses keyword extraction, constrained generation, over generation, and filtering to produce obfuscated rewrites with smaller language



(a) Attribution risk (b) Verification risk (c) NLI and CoLA

Figure 2: Attribution risk, verification risk, NLI, and CoLA across bundle sizes on Reddit. In (c), each synthetic output is evaluated against its corresponding private text, so the private text serves as the reference and is not plotted.

models. We also include RUPTA (Yang et al., 2025) as an additional LLM-based rewriting baseline to test whether LLM rewriting alone is sufficient for aggregation-level authorship privacy.

Baseline selection and settings are provided in Appendix B.2. The RUPTA comparison in Appendix H tests whether strong LLM-based rewriting alone is sufficient for aggregation-level authorship privacy.

4.3 Metrics

Privacy Metrics. We evaluate same-genre aggregation risk with authorship attribution and verification attacks. For the main same-genre attribution evaluation, we use LUAR (Rivera-Soto et al., 2021) as a strong neural attribution attacker and report Mean Reciprocal Rank (MRR) and Recall at 8 (R@8). For verification, we use PAN22 authorship verification (Stamatatos et al., 2022) and report Area Under the Curve (AUC) and Correctness at One (c@1). To test whether the privacy gains transfer beyond the neural models used in the main evaluation, we also include independent classical stylometric attribution and verification attacks. These stylometric attacks provide non-neural transfer evaluations for both same-genre attribution and verification.

For cross-genre evaluation, we follow CROSSNEWS and use SELMA, its authorship embedding model, to test whether synthetic text remains linkable across Article and Tweet genres. We use the strongest SELMA prompt variants for each task. For attribution, we use SELMA+TaskOnly and SELMA+LIP; for verification, we use SELMA+TaskOnly and SELMA+PromptAV. We report the CROSSNEWS metrics: top-one Accuracy, R@8, and Average Rank for attribution, and Accuracy and F1 for verification. Privacy metric details are provided in Appendix C.1.

Utility Metrics. We evaluate synthetic text quality with semantic quality, linguistic acceptability, and sentiment alignment. Semantic quality is measured by NLI, linguistic acceptability by CoLA, and sentiment alignment by comparing sentiment labels between private and synthetic texts. Utility metric details are provided in Appendix C.2.

4.4 Models and Implementation Settings

We use bart-large-cnn for abstraction, sentiment-roberta-large-en for sentiment alignment, sentence-t5-base for semantic scoring, Mistral-Small and Phi-4 for generation. AAST$_M$ denotes AAST with Mistral-Small as the generator, and AAST$_\Phi$ denotes AAST with Phi-4. Models and implementation settings are provided in Appendix B.3.

5 Results

5.1 Privacy Results

5.1.1 Same-genre authorship attribution

Reddit authorship attribution. Using the main LUAR attribution attacker, Figure 2a shows that aggregation sharply increases attribution risk on Reddit (details reported in Appendix Table 3). The private setting rises from moderate linkability at K=1 to near perfect linkability at larger K. By K=8, private text already reaches 100 R@8, showing that only a small number of texts from the same account can make attribution highly reliable. In result tables, the best value is shown in **bold**, and the second best value is underlined.

AAST consistently reduces this attribution risk across all bundle sizes. At small K, AAST$_M$ gives the lowest risk (e.g., 2.2 MRR and 2.9 R@8 at K=1). As K grows, AAST$_\Phi$ becomes stronger

on most larger bundle sizes. At K=16, AAST$_\Phi$ obtains the lowest risk among all methods, with 39.8 MRR compared with 54.7 for JAMDEC and 92.7 for KiP. AAST$_\mathrm{M}$ also remains below both rewriting baselines. These results show that AAST reduces authorship attribution under aggregation on Reddit, although the attack still becomes harder to suppress as K increases.

Blog authorship attribution. AAST shows a clear advantage on Blog. As shown in Appendix Figure 4a and Table 3, AAST gives the lowest attribution risk across all bundle sizes. AAST$_\Phi$ is the strongest setting throughout, at K=16, AAST$_\Phi$ keeps R@8 at 61.4, compared with 83.6 for JAMDEC and 94.0 for KiP. AAST$_\mathrm{M}$ is consistently the next strongest method. The gap is most visible at larger K, where AAST remains much harder to attribute under stronger aggregation pressure.

Stylometric attribution. The LUAR attribution results above provide an attacker-aware neural evaluation because AAST uses an authorship episode encoder during bundle-level selection. To test whether the gains transfer beyond LUAR, Appendix Table 5 evaluates the same-genre setting with an independent classical n-gram stylometric attribution attacker. AAST remains effective under this unseen attacker. At K=16, the two AAST variants obtain the lowest and second-lowest MRR and R@8 on both Reddit and Blog. These results provide direct transfer evidence that AAST's attribution gains are not confined to the LUAR-based neural evaluator, but persist under an independent non-neural stylometric attacker. Full results are in Appendix E.

5.1.2 Same-genre authorship verification

Reddit authorship verification. In Figure 2b, the private curve shows that aggregation also makes authorship verification much easier. AAST keeps verification risk much lower across bundle sizes. AAST$_\mathrm{M}$ is usually the strongest setting, especially at the largest bundle size, where it keeps AUC at 64.1, while JAMDEC and KiP rise to 81.4 and 92.6. AAST$_\Phi$ also improves over both rewriting baselines for most K. These results show that AAST reduces verification risk under aggregation, with the clearest gains on AUC, even when larger bundles make the attack stronger. The corresponding values are reported in Appendix Table 4.

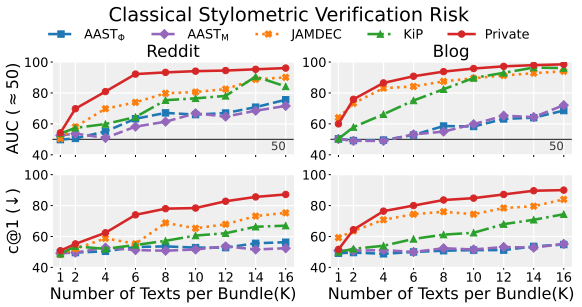


Figure 3: Verification results under a stylometric transfer attack on Reddit and Blog.

Blog authorship verification. Appendix Figure 4b and Table 4 show that AAST remains stable against verification risk on Blog. AAST$_\mathrm{M}$ gives the strongest protection at larger bundle sizes. At K=16, it keeps AUC at 68.6 and c@1 at 50.3, while JAMDEC reaches 73.3 AUC and 61.4 c@1, and KiP reaches 85.3 AUC and 68.1 c@1. The Blog verification results show that AAST weakens verification of whether two texts come from the same author under aggregation.

Stylometric verification. We also evaluate AAST with the classical stylometric transfer attack. Figure 3 and Appendix Table 6 show the same trend. At K=16, AAST$_\mathrm{M}$ gives lower AUC than JAMDEC and KiP on Reddit, 71.6 versus 90.1 and 84.2, respectively. On Blog, AAST$_\Phi$ gives the lowest AUC 68.6 and lowest c@1 54.8. Together with the stylometric attribution results in Appendix E, these results provide transfer evidence that AAST's privacy gains are not limited to neural authorship models, but persist under independent classical stylometric attacks. Full verification results are in Appendix F.

AAST$_\Phi$ and AAST$_\mathrm{M}$ show complementary strengths. AAST$_\Phi$ is often stronger on attribution at larger K, while AAST$_\mathrm{M}$ is usually more stable for verification and cross-genre settings. The stylometric attribution and verification transfer evaluations follow the same pattern. Since the selection objective is the same, these differences likely come from the candidate distributions produced by the two generators, including differences in semantic preservation, style shift, and candidate diversity. We leave a controlled analysis of generator behavior for future work.

5.1.3 Cross-genre authorship attribution

Table 1 shows that AAST reduces attribution risk when the attacker links across genres. Across

Genre Pair	Attack	Method	Acc.(↓)	R@8(↓)	Avg.Rank(↑)
Tweet–Article	SELMA + TaskOnly	AAST_{M}	**1.2**	**7.5**	173
		AAST_{Φ}	1.9	7.6	**175**
		KiP	2.7	9.8	157
		JAMDEC	2.2	11.3	121
	SELMA + LIP	AAST_{M}	**0.9**	**5.0**	**192**
		AAST_{Φ}	1.6	6.6	185
		KiP	2.2	8.4	165
		JAMDEC	2.2	10.8	125
Article–Tweet	SELMA + TaskOnly	AAST_{M}	**1.3**	**5.7**	**222**
		AAST_{Φ}	2.2	7.5	210
		KiP	5.1	14.7	160
		JAMDEC	1.6	8.1	167
	SELMA + LIP	AAST_{M}	**1.2**	**5.4**	**224**
		AAST_{Φ}	1.9	6.9	213
		KiP	4.0	12.6	166
		JAMDEC	1.5	7.3	169

Table 1: Cross-genre authorship attribution risk results.

both genre pairs and both SELMA prompt settings, AAST_{M} gives the lowest top-one Accuracy and R@8. In Tweet–Article, AAST_{M} gives the lowest Accuracy and R@8 under both SELMA+TaskOnly and SELMA+LIP. Under SELMA+LIP, it reduces R@8 to 5.0 and pushes the true author to an average rank of 192. AAST_{Φ} is usually the second strongest method.

The same pattern appears in Article–Tweet. AAST_{M} achieves the lowest Accuracy and R@8 under both attacks, and gives the highest average rank of the true author. These results show that AAST's protection is not limited to same-genre aggregation, it also weakens author signals that transfer across genres. This comes from identifier masking removing explicit markers, candidate preselection reducing stylistic cues, and bundle-level selection reducing consistent account-level signals.

5.1.4 Cross-genre authorship verification

Appendix Table 10 shows that cross-genre verification follows the attribution results. AAST lowers the verifier's ability to link synthetic and reference texts across both genre datasets, and AAST_{M} usually gives the lowest verification risk. Complete results are provided in Appendix I.

Note on Article–Tweet results. Article–Tweet is the harder cross-genre setting because tweet inputs often contain little text beyond handles, URLs, hashtags, and brief reactions. After these surface markers are normalized, the remaining text can be short or generic, which gives bundle-level selection less semantic and stylistic variation to use. Even in this harder setting, AAST_{M} gives the strongest cross-genre privacy results. Further discussion is provided in Appendix K.

5.2 Utility Results

Reddit. On Reddit, AAST maintains strong semantic quality (NLI) and linguistic quality (CoLA) while reducing account-level authorship linkability (See Figure 2c and Appendix Table 7). AAST_{Φ} gives the best CoLA score at every K, and its NLI improves as K grows, reaching 72.9 at K=16. AAST_{M} follows closely, with 72.6 NLI at K=16. KiP has strong NLI, but its privacy risks remain high. JAMDEC gives lower attribution and verification risk than KiP in some settings, but its NLI is much lower than AAST.

AAST gives a stronger privacy and utility balance, preserving both content and linguistic quality while reducing linkability under aggregation. We note that CoLA scores for AAST are consistently high across bundle sizes, so CoLA mainly confirms linguistic acceptability for our method. In contrast, lower CoLA for KiP reflects the token fragmentation artifacts observed in some outputs (Appendix Table 20). We still retain CoLA for comparability with baselines.

AAST also conditions generation on sentiment, so we measure whether each synthetic text preserves the sentiment of its corresponding private text. We report this metric only for AAST, since KiP and JAMDEC do not include sentiment control. Appendix Figure 5 and Table 8 show that AAST maintains stable sentiment alignment on Reddit, staying between 83% and 85.5%. This suggests that AAST largely preserves the sentiment of the original posts while reducing authorship linkability.

Blog and cross-genre utility. The Blog utility results follow the same pattern, with AAST preserving strong linguistic quality and stable sentiment alignment while maintaining lower authorship linkability (see Figure 4c). Blog utility results are provided in Appendix G.2.

Cross-genre utility results are provided in Appendix J, with detailed values in Table 11. In both Tweet–Article and Article–Tweet, AAST_{M} achieves the highest CoLA score and AAST_{Φ} is consistently second best, indicating stronger linguistic quality than the baselines.

5.3 LLM Rewriting Baseline Comparison

Appendix Table 9 compares AAST with RUPTA, a recent LLM-based text anonymization baseline, on Reddit and Blog. The results show that strong LLM-based rewriting alone is not sufficient for aggregation-level authorship privacy. Although

some RUPTA variants obtain higher NLI, they also show much higher attribution and verification risk as K grows. This suggests that staying closer to the private text can preserve semantic overlap, but can also preserve authorial cues. In contrast, AAST uses bundle-level selection to reduce account-level linkability while maintaining reasonable NLI and strong CoLA, giving a stronger privacy–utility balance under aggregation. Full results are in Appendix H.

5.4 Ablation Analysis

Ablation design. We use ablations to clarify which modules drive AAST's privacy and utility tradeoff. The component ablation studies Identifier Masking (§3.3), Candidate Preselection (§3.6), Candidate Refinement (§3.7.2), and Bundle-Level Selection (§3.7). We further add NoBundleSel, a controlled baseline that keeps the same settings as AAST but removes Bundle-Level Candidate Scoring and Candidate Refinement. This comparison isolates the role of bundle-level selection under our same-genre setting.

Component effects. Appendix Table 15 shows that different modules address different risks. Candidate Preselection, Refinement, and Bundle-Level Selection are always enabled in all experiments, while Identifier Masking is enabled only for cross-genre settings. Identifier Masking improves privacy but lowers NLI, so we use it only when surface markers are frequent. Candidate Preselection targets direct private–synthetic carryover, disabling it gives Bundle-Level Selection a larger candidate pool and can lower synthetic bundle linkability. Bundle-Level Selection is the main module for bundle-level privacy at larger K. Appendix L.1 provides more details on component effects.

Effect of removing bundle-level selection. Appendix Figure 7 compares AAST_{Φ} with $\text{NoBundleSel}_{\Phi}$ on Reddit and Blog. $\text{NoBundleSel}_{\Phi}$ keeps the same generator, prompt, abstraction model, and candidate pool as AAST_{Φ}, but removes Bundle-Level Selection. $\text{NoBundleSel}_{\Phi}$ shows higher attribution and verification risk as K grows, while NLI and CoLA remain close to AAST_{Φ}. This shows that bundle-level selection is the main source of the privacy gain, and that this gain does not come from a meaningful loss in semantic or linguistic utility. Full results are provided in Appendix L.2.

5.5 Effect of Noisy Selection

We also study a noisy abstraction selection variant, where Gaussian noise is added to abstraction selection scores before choosing the abstraction used for generation. As shown in Appendix M and Table 17, stronger noise generally reduces attribution and verification risk at larger bundle sizes, while introducing a small utility tradeoff.

5.6 Qualitative Analysis

We qualitatively inspect Reddit K=2 examples, focusing on whether each method preserves main events and core meaning. AAST keeps the central events and core meaning, such as the disease name, need for medical resources, loss of faith, and mental health concern, while changing the surface form. KiP preserves many local details, but often stays close to private wording and structure, and sometimes produces fragmented words or control artifacts. JAMDEC moves farther from the private text, but often loses core information through repetitive outputs and semantic drift. Appendix N provides complete examples and summary of observed strengths and weaknesses.

5.7 Generation Cost

Computational efficiency analysis. AAST is the most efficient method across all bundle sizes (Appendix Figure 6). KiP requires about $1.9\times$ longer runtime on average, while JAMDEC requires about $18.5\times$ longer runtime. Appendix O.1 reports experimental settings, relative runtime, and output length analysis for interpreting the comparison.

Token cost analysis. Token usage grows almost linearly with bundle size because larger K requires generating more synthetic texts. At K=16, 250 users $\times$ 2 bundles per user $\times$ 16 texts per bundle requires generation for 8,000 texts. Token usage results are provided in Appendix O.2.

6 Conclusion

We have introduced AAST, an aggregation-aware synthetic text generation framework for reducing account-level authorship linkability. AAST shifts the privacy target from isolated text rewriting to bundle-level selection over a released account history. Across same-genre, cross-genre, neural, and independent non-neural stylometric attacks, AAST reduces account-level linkability as bundle size grows while maintaining semantic quality, linguistic acceptability, and sentiment alignment.

Limitations

AAST reduces account-level authorship linkability, but it is not a complete anonymization mechanism. Because AAST preserves meaning, sentiment, and linguistic quality, some private topics, events, and preferences can remain in the synthetic text. Our utility evaluation also relies on automatic metrics and qualitative examples. While NLI, CoLA and sentiment alignment capture important dimensions of text quality, they may not fully reflect human judgments of fluency or downstream utility, which remain important directions for future work.

AAST is less effective when released texts are very short and metadata heavy, as in some Article–Tweet cases. AAST still gives the strongest Article–Tweet privacy results, but tweet-like inputs often contain limited free text beyond handles, URLs, hashtags, and brief reactions (see Appendix Table 12). This gives bundle-level selection less semantic and stylistic material to use. Future work should design generation and selection methods for tweet-like text.

AAST uses a LUAR-based episode encoder during bundle-level selection, so same-genre LUAR attribution is not fully independent of the selector. To test transfer beyond this selector, we also evaluate independent stylometric attribution and verification, PAN22 verification, and SELMA cross-genre attacks, none of which are used during selection. These results suggest that AAST's gains transfer beyond the LUAR-based evaluator. Future work should evaluate stronger adaptive attackers under the same aggregation protocol.

Ethical Considerations

Our work uses Reddit posts that were publicly accessible at the time of collection and gathered through Reddit's official API. Because the Reddit corpus concerns financial hardship and poverty, we treat it as sensitive even though it is public. We follow a data minimization principle: we collect only the text and metadata needed for authorship privacy evaluation, do not collect real names or external profile information, and do not combine Reddit content with outside sources to identify individuals.

We do not contact users, infer real world identities, or report results about individual accounts. Examples in this paper are used only when needed to illustrate method behavior. We avoid usernames, direct links, and unnecessary verbatim quotation in examples, and we mask identifying surface details when possible.

Access to the collected dataset will be limited to verified researchers upon request and under conditions that restrict use to privacy and synthetic text research. Researchers must agree not to attempt re-identification, contact users, redistribute the data, or link the posts with external sources. We will also honor reasonable removal requests when a post or author can be identified in the released research data.

AAST's bundle-level objective uses crowd similarity to reduce account-level linkability. While this improves privacy against attribution and verification attacks, it may also increase the chance that a synthetic bundle is mistakenly associated with nearby users. We do not intend AAST to imitate or impersonate specific individuals. Future deployments should monitor false attribution concentration, avoid outputs that closely mimic identifiable users, and use additional safeguards when the surrounding user population is small or sensitive.

AAST is intended as a defensive risk reduction tool, not as a guarantee of anonymity. As discussed in the Limitations section, synthetic outputs may still support some forms of linkage. Deployments should therefore avoid presenting AAST as complete anonymization and should use it only with appropriate privacy review and safeguards.

With respect to dual use, AAST is designed as a defensive tool to help users protect their privacy online. However, methods that reduce authorship linkability could also be misused to obscure responsibility for harmful or policy violating content, or to create misleading impressions about authorship. We recommend that AAST be used only for legitimate privacy preservation, privacy research, and controlled synthetic text release, with safeguards against impersonation, deceptive use, and attempts to evade accountability.

Acknowledgments

This work was partially supported by the National Science Foundation under Award No. 2247723.

References

Marah I Abdin, Jyoti Aneja, Harkirat S. Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, James R. Lee, Yin Tat Lee, Yuanzhi Li, Weishung Liu, Caio C. T. Mendes, Anh Nguyen, Eric Price, Gustavo de Rosa, Olli Saarikivi, and 8 others. 2024. Phi-4 technical report. *CoRR*, abs/2412.08905.

Kenneth Alperin, Rohan Leekha, Adaku Uchendu, Trang Nguyen, Srilakshmi Medarametla, Carlos Levya Capote, Seth Aycock, and Charlie Dagli. 2025. Masks and mimicry: Strategic obfuscation and impersonation attacks on authorship verification. In *Proceedings of the 5th International Conference on Natural Language Processing for Digital Humanities*, pages 102–116.

Malik Altakrori, Thomas Scialom, Benjamin CM Fung, and Jackie Chi Kit Cheung. 2022. A multifaceted framework to evaluate evasion, content preservation, and misattribution in authorship obfuscation techniques. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2391–2406.

Calvin Bao and Marine Carpuat. 2024. Keep it private: Unsupervised privatization of online text. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8678–8693.

Angelo Basile, Gareth Dwyer, Maria Medvedeva, Josine Rawee, Hessel Haagsma, and Malvina Nissim. 2017. N-gram: New groningen author-profiling model. *arXiv preprint arXiv:1707.03764*.

Janek Bevendorff, Matti Wiegmann, Emmelie Richter, Martin Potthast, and Benno Stein. 2025. The two paradigms of llm detection: Authorship attribution vs authorship verification. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 3762–3787.

Lucas Georges Gabriel Charpentier and Pierre Lison. 2025. Re-identification of de-identified documents with autoregressive infilling. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1192–1209.

Yao Dou, Isadora Krsek, Tarek Naous, Anubha Kabra, Sauvik Das, Alan Ritter, and Wei Xu. 2024. Reducing privacy risks in online self-disclosures with language models. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 13732–13754.

Cynthia Dwork and Aaron Roth. 2014. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3-4):211–407.

Jillian Fisher, Skyler Hallinan, Ximing Lu, Mitchell L Gordon, Zaid Harchaoui, and Yejin Choi. 2024a. Styleremix: Interpretable authorship obfuscation via distillation and perturbation of style elements. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4172–4206.

Jillian Fisher, Ximing Lu, Jaehun Jung, Liwei Jiang, Zaid Harchaoui, and Yejin Choi. 2024b. Jamdec: Unsupervised authorship obfuscation using constrained decoding over small language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1552–1581.

Jochen Hartmann, Mark Heitmann, Christian Siebert, and Christina Schamp. 2023. More than a feeling: Accuracy and application of sentiment analysis. *International Journal of Research in Marketing*, 40(1):75–87.

David I Holmes. 1998. The evolution of stylometry in humanities scholarship. *Literary and linguistic computing*, 13(3):111–117.

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

Shuo Huang, William MacLean, Xiaoxi Kang, Anqi Wu, Lizhen Qu, Qiongkai Xu, Zhuang Li, Xingliang Yuan, and Gholamreza Haffari. 2024b. Nap^ 2: A benchmark for naturalness and privacy-preserving text rewriting by learning from human. *arXiv preprint arXiv:2406.03749*.

Shuo Huang, Xingliang Yuan, Gholamreza Haffari, and Lizhen Qu. 2025. Zero-shot privacy-aware text rewriting via iterative tree search. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 9175–9190.

Chia-Yu Hung, Zhiqiang Hu, Yujia Hu, and Roy Lee. 2023. Who wrote it and why? prompting large-language models for authorship verification. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14078–14084.

Georgi Karadzhov, Tsvetomila Mihaylova, Yasen Kiprov, Georgi Georgiev, Ivan Koychev, and Preslav Nakov. 2017. The case for being average: A mediocrity approach to style masking and author obfuscation: (best of the labs track at clef-2017). In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 173–185. Springer.

Yashwant Keswani, Harsh Trivedi, Parth Mehta, and Prasenjit Majumder. 2016. Author masking through translation. *CLEF (Working Notes)*, 1609:890–894.

Moshe Koppel and Jonathan Schler. 2004. Authorship verification as a one-class classification problem. In

Proceedings of the twenty-first international conference on Machine learning, page 62.

Alex Leavitt. 2015. " this is a throwaway account" temporary technical identities and perceptions of anonymity in a massive online community. In *Proceedings of the 18th ACM conference on computer supported cooperative work & social computing*, pages 317–327.

Simon Lermen, Daniel Paleka, Joshua Swanson, Michael Aerni, Nicholas Carlini, and Florian Tramèr. 2026. Large-scale online deanonymization with llms. *arXiv preprint arXiv:2602.16800*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.

Alisa Liu, Swabha Swayamdipta, Noah A. Smith, and Yejin Choi. 2022. WANLI: Worker and AI collaboration for natural language inference dataset creation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6826–6847, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Marcus Ma, Duong Minh Le, Junmo Kang, Yao Dou, John Cadigan, Dayne Freitag, Alan Ritter, and Wei Xu. 2025. Crossnews: A cross-genre authorship verification and attribution benchmark. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 24777–24785.

Qian Ma and Sarah Rajtmajer. 2026. Private seeds, public LLMs: Realistic and privacy-preserving synthetic data generation. In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 189–210, San Diego, California, United States. Association for Computational Linguistics.

Asad Mahmood, Faizan Ahmad, Zubair Shafiq, Padmini Srinivasan, and Fareed Zaffar. 2019. A girl has no name: Automated authorship obfuscation using mutant-x. *Proceedings on Privacy Enhancing Technologies*.

Fatemehsadat Mireshghallah and Taylor Berg-Kirkpatrick. 2021. Style pooling: Automatic text style obfuscation for improved classification fairness. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2009–2022.

Mistral. 2025. Mistral-small-3.2-24b-instruct-2506. `https://huggingface.co/mistralai/Mistral-Small-3.2-24B-Instruct-2506`.

Frederick Mosteller and David L Wallace. 1963. Inference in an authorship problem: A comparative study of discrimination methods applied to the authorship of the disputed federalist papers. *Journal of the American Statistical Association*, 58(302):275–309.

Tempestt Neal, Kalaivani Sundararajan, Aneez Fatima, Yiming Yan, Yingfei Xiang, and Damon Woodard. 2017. Surveying stylometry techniques and applications. *ACM Computing Surveys (CSuR)*, 50(6):1–36.

Tuc Nguyen, Yifan Hu, and Thai Le. 2025. Unraveling interwoven roles of large language models in authorship privacy: Obfuscation, mimicking, and verification. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 14914–14930.

Jianmo Ni, Gustavo Hernández Ábrego, Noah Constant, Ji Ma, Keith B. Hall, Daniel Cer, and Yinfei Yang. 2021. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. *Preprint*, arXiv:2108.08877.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, and 1 others. 2011. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830.

Anselmo Peñas and Alvaro Rodrigo. 2011. A simple measure to assess non-response. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1415–1424, Portland, Oregon, USA. Association for Computational Linguistics.

Ildikó Pilán, Pierre Lison, Lilja Øvrelid, Anthi Papadopoulou, David Sánchez, and Montserrat Batet. 2022. The text anonymization benchmark (tab): A dedicated corpus and evaluation framework for text anonymization. *Computational Linguistics*, 48(4):1053–1101.

Martin Potthast, Matthias Hagen, and Benno Stein. 2016. Author obfuscation: Attacking the state of the art in authorship verification. *CLEF (Working Notes)*, pages 716–749.

Jovan Powar and Alastair R Beresford. 2023. Sok: Managing risks of linkage attacks on data privacy. *Proceedings on Privacy Enhancing Technologies*.

Rafael A Rivera-Soto, Olivia Elizabeth Miano, Juanita Ordonez, Barry Y Chen, Aleem Khan, Marcus Bishop, and Nicholas Andrews. 2021. Learning universal authorship representations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 913–919.

Jonathan Schler, Moshe Koppel, Shlomo Argamon, and James W Pennebaker. 2006. Effects of age and gender on blogging. In *AAAI spring symposium: Computational approaches to analyzing weblogs*, volume 6, pages 199–205. Stanford, CA, USA.

Shachar Seidman. 2013. Authorship verification using the impostors method. In *CLEF 2013 Evaluation labs and workshop–Working notes papers*, pages 23–26.

Rakshith Shetty, Bernt Schiele, and Mario Fritz. 2018. A4NT: Author attribute anonymity by adversarial training of neural machine translation. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 1633–1650, Baltimore, MD. USENIX Association.

Robin Staab, Mark Vero, Mislav Balunović, and Martin Vechev. 2025. Language models are advanced anonymizers. In *The Thirteenth International Conference on Learning Representations*.

Efstathios Stamatatos. 2009. A survey of modern authorship attribution methods. *Journal of the American Society for information Science and Technology*, 60(3):538–556.

Efstathios Stamatatos, Mike Kestemont, Krzysztof Kredens, Piotr Pezik, Annina Heini, Janek Bevendorff, Benno Stein, and Martin Potthast. 2022. Overview of the authorship verification task at pan 2022. In *CEUR workshop proceedings*, volume 3180, pages 2301–2313.

Estathios Stamatatos, Michael Tschnuggnall, Ben Verhoeven, Walter Daelemans, Gunther Specht, Benno Stein, and Michael Potthast. 2016. Clustering by authorship within and across documents. In *Working Notes Papers of the CLEF 2016 Evaluation Labs. CEUR Workshop Proceedings/Balog, Krisztian [edit.]; et al.*, pages 691–715.

Xinhao Tan, Songhua Liu, Xia Cong, Kunjun Li, and Xinchao Wang. 2025. Open-world authorship attribution. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 17744–17758.

Rachael Tatman. 2015. Blog authorship corpus. https://www.kaggle.com/datasets/rtatman/blog-authorship-corpus/data. Kaggle dataset, accessed March 19, 2026.

William J Teahan and David J Harper. 2003. Using compression-based language models for text categorization. In *Language modeling for information retrieval*, pages 141–165. Springer.

Anthony Henry Triggs, Kristian Møller, and Christina Neumayer. 2021. Context collapse and anonymity among queer reddit users. *New Media & Society*, 23(1):5–21.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Chulin Xie, Zinan Lin, Arturs Backurs, Sivakanth Gopi, Da Yu, Huseyin A Inan, Harsha Nori, Haotian Jiang, Huishuai Zhang, Yin Tat Lee, and 1 others. 2024. Differentially private synthetic data via foundation model apis 2: Text. *arXiv preprint arXiv:2403.01749*.

Tianyu Yang, Xiaodan Zhu, and Iryna Gurevych. 2025. Robust utility-preserving text anonymization based on large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 28922–28941.

Xiang Yue, Huseyin Inan, Xuechen Li, Girish Kumar, Julia McAnallen, Hoda Shajari, Huan Sun, David Levitan, and Robert Sim. 2023. Synthetic text generation with differential privacy: A simple and practical recipe. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1321–1342.

A Additional Details for AAST

A.1 Abstraction Models and Sentiment Alignment

For each private sample x and candidate $a_j \in \mathcal{A}(x)$, we compute the composite score $r(x, a_j)$, which combines semantic similarity, sentiment alignment, and mismatch penalty, and the highest scoring candidate is selected before generation. We use facebook/bart-large-cnn (Lewis et al., 2020) to generate an initial abstraction candidate pool. We compute sentence embeddings with sentence-t5-base (Ni et al., 2021) and measure semantic similarity by cosine similarity. A sentiment model, siebert/sentiment-roberta-large-en (Hartmann et al., 2023), provides polarity $y(a) \in \{0, 1\}$. Following the abstraction setting (Ma and Rajtmajer, 2026), we set a sentiment control parameter for abstraction. The top 5 candidates by $\mathrm{score}(\cdot)$ form $\mathcal{A}(x)$, which is the input to the noisy abstraction selection step in §3.4.2.

A.2 Noisy Abstraction Selection

Differential privacy (DP) provides a formal guarantee for randomized mechanisms (Dwork and Roth, 2014). A mechanism M satisfies (ϵ, δ)-DP if, for any neighboring datasets $\mathcal{X}$ and $\mathcal{X}'$ that differ in one element, and any event $S \subseteq \mathrm{Range}(M)$,

$$\Pr[M(\mathcal{X}) \in S] \leq e^{\epsilon} \Pr[M(\mathcal{X}') \in S] + \delta.$$

AAST can add Gaussian noise in the abstraction score selection step. The scope of the Gaussian noise analysis is limited: it applies only to the noisy score selection rule, conditioned on the candidate set used by the selector. Since the candidates are generated from the private input, we do not claim end-to-end DP for the selected abstraction text or the final synthetic corpus.

Because the raw composite score $r(x, a_j)$ is not necessarily bounded, we first normalize scores within the candidate set $\mathcal{A}(x)$:

$$u(x, a_j) = \frac{r(x, a_j) - \min\limits_{a_\ell \in \mathcal{A}(x)} r(x, a_\ell)}{\max\limits_{a_\ell \in \mathcal{A}(x)} r(x, a_\ell) - \min\limits_{a_\ell \in \mathcal{A}(x)} r(x, a_\ell)}, \quad (5)$$

when the denominator is nonzero, and set $u(x, a_j) = 0$ otherwise. This gives $u(x, a_j) \in [0, 1]$. We use $\Delta u = 1$ as the per candidate score sensitivity after normalization (Xie et al., 2024).

Given privacy budget ϵ, number of private samples N, and $\delta = \frac{1}{N \log N}$, we set the Gaussian noise scale as

$$\sigma = \frac{\sqrt{2 \log(1.25/\delta)}\, \Delta u}{\epsilon}. \quad (6)$$

For each private sample $x \in \mathcal{X}$ and candidate $a_j \in \mathcal{A}(x)$, we sample $\tilde{u}_j = u(x, a_j) + \eta_j, \eta_j \sim \mathcal{N}(0, \sigma^2)$. The noisy selected abstraction is

$$a_{\text{noisy}}(x) = \arg \max_{a_j \in \mathcal{A}(x)} \tilde{u}_j. \quad (7)$$

A.3 Analysis for Noisy Score Selection

Proposition 1. Analysis of noisy score selection. For a private sample x, let $\mathcal{A}(x) = \{a_1, \ldots, a_m\}$ be the candidate set used by the abstraction selector. Conditioned on this candidate set, define the normalized utility vector $\mathbf{u}(x) = \big(u(x, a_1), \ldots, u(x, a_m)\big)$, where each $u(x, a_j) \in [0, 1]$. Assume that $\mathbf{u}$ has global ℓ_2-sensitivity at most Δu with respect to neighboring private inputs. If Gaussian noise with scale σ from Eq. 6 is added to $\mathbf{u}(x)$, then the noisy score vector is (ϵ, δ)-DP. Selecting the index of the largest noisy score is also (ϵ, δ)-DP by post processing. This statement is conditional on the candidate set and does not cover candidate generation, the selected abstraction text, or the final synthetic corpus.

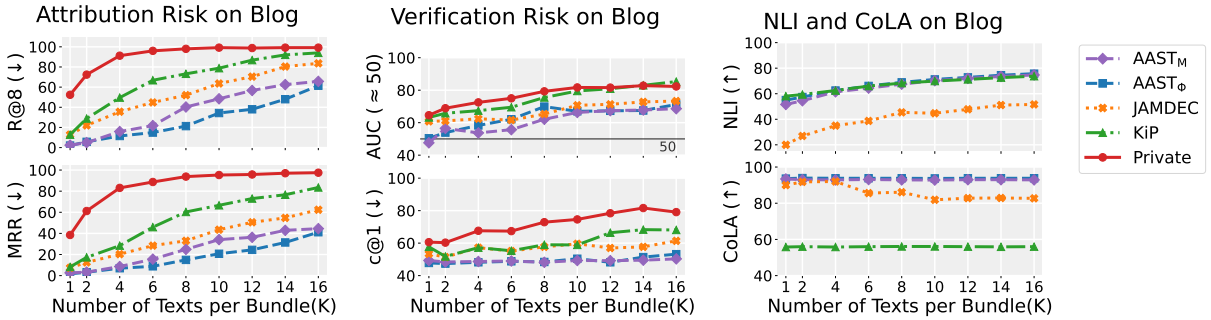
Proof. Consider the vector valued query $\mathbf{u}(x)$, conditioned on the candidate set used by the selector. The mechanism releases $\tilde{\mathbf{u}}(x) = \mathbf{u}(x) + \mathbf{z}, \mathbf{z} \sim \mathcal{N}(0, \sigma^2 I)$, where σ is calibrated to sensitivity Δu as in Eq. 6. By the Gaussian mechanism (Dwork and Roth, 2014), $\tilde{\mathbf{u}}(x)$ satisfies (ϵ, δ)-DP under this conditional score selection view. The selected index $j^* = \arg\max_j \tilde{u}_j$ is a deterministic function of the noisy scores, and also satisfies (ϵ, δ)-DP by post processing.

A.4 Complexity Analysis of AAST

A direct search over all combinations in a K bundle is exponential in K. For example, jointly searching over a Query bundle and a Target bundle requires evaluating C^{2K} combinations per user. We instead use coordinate descent, which updates one slot at a time while keeping all other slots fixed, and use one pass in our experiments.

For each slot update, AAST evaluates at most C candidate bundles through the episode embedding function. Since each user has K slots in Query and K slots in Target, one pass requires at most $2K \cdot C$ episode embeddings per user. Across all users, the dominant episode embedding cost is

$$\mathcal{O}\big(|\mathcal{U}| \cdot 2K \cdot C\big),$$



(a) Attribution risk (b) Verification risk (c) NLI and CoLA

Figure 4: Attribution risk, verification risk, NLI, and CoLA across bundle sizes on Blog. In (c), each synthetic output is evaluated against its corresponding private text, so the private text serves as the reference and is not plotted.

where $\mathcal{U}$ denotes the set of users.

The scoring objective also computes partner similarity and crowd similarity for each candidate. Partner similarity is constant cost once bundle embeddings are available. Crowd similarity compares the candidate bundle embedding with $\mathcal{V}_{-u}$, adding similarity computations against other users' current bundle embeddings. Candidate Refinement adds text similarity computations within the near optimal set $\mathcal{R}$, whose size is bounded by the candidate pool. These terms add similarity costs, but they do not change the dominant number of episode embedding evaluations. Thus, coordinate descent reduces the dominant search factor from exponential in K to linear in K and C.

B Additional Experimental Details

B.1 Prompt Design

Because the input to the generator is abstracted private text, we use the following prompt for LLM generation:

"*Rewrite the abstracted text while preserving its core meaning. Vary sentence structure and word choice to reduce stylistic similarity.*"

This prompt makes clear that the input is already an abstraction of the private text, while guiding the generator to preserve the main meaning and reduce stylistic similarity. We use the same prompt for both Mistral-Small and Phi-4 generators.

B.2 Baseline Selection and Setup

We choose KiP and JAMDEC as the main baselines because they are recent methods designed for authorship privacy and evaluated with authorship attacks. Many privacy rewriting methods focus on removing sensitive attributes, named entities, or private disclosures, which is related but not the same as reducing authorship attribution and verification risk. We focus on KiP and JAMDEC as the main baselines because they directly target author style obfuscation. In addition, we include RUPTA as an expanded LLM-based rewriting comparison, since it provides a recent text anonymization baseline without aggregation-aware bundle-level selection. This allows us to test whether strong LLM rewriting alone is sufficient under account-level aggregation.

To approximately match generation budgets, AAST and JAMDEC both use `max_new_tokens` = 128. KiP uses `token_max_length` instead. On the Reddit dataset, the average text length is 219.6 tokens, which gives 219.6 + 128 = 347.6. We therefore set `token_max_length` = 345. On the Blog dataset, the average text length is 243.4 tokens, which gives 243.4 + 128 = 371.4. Thus, we set `token_max_length` = 370. We also set both `lex_diversity` and `order_diversity` to 60 for KiP to encourage more diverse outputs.

B.3 Models and Implementation Settings

Model choices and fixed implementation settings are summarized in Table 2. These values are held constant across all experiments.

C Additional Metrics Details

C.1 Privacy Metrics Details

For same-genre evaluation, we use authorship attribution and verification attacks. LUAR attribution ranks the true user among a candidate set for each bundle (Rivera-Soto et al., 2021). We report Mean Reciprocal Rank (MRR) (Rivera-Soto et al., 2021), which averages the reciprocal rank of the true user, and Recall at 8 (R@8) (Bao and Carpuat,

Model or Parameter	Value	Purpose
Abstraction model (§3.4)	`facebook/bart-large-cnn` (Lewis et al., 2020)	Generates abstraction candidates from each private text.
Sentiment classifier model (§3.4)	`sentiment-roberta-large-en` (Hartmann et al., 2023)	Checks whether abstraction candidates preserve the private text sentiment.
Semantic scoring model (§3.4.2)	`sentence-t5-base` (Ni et al., 2021)	Computes semantic similarity between the private text and abstraction candidates.
Generator (§3.5)	`Mistral-Small` (Mistral, 2025) and `Phi-4` (Abdin et al., 2024)	Generates synthetic candidate rewrites from selected abstractions.
Style embedding model (§3.6)	`intfloat/e5-base-v2` (Wang et al., 2022)	Filters candidates that remain too close to the private input in stylistic space.
Episode encoder model (§3.7.1)	`Luar-mud` (Rivera-Soto et al., 2021)	Computes bundle embeddings for partner and crowd similarity scoring.
Text similarity surrogate n	$\in \{3, 4, 5\}$	Character term frequency-inverse document frequency. Refines among near optimal candidates using bundle-level textual similarity.
Initial abstraction candidates h	10	Number of candidates generated before abstraction candidate selection.
Abstraction Candidate set size	5	Number of abstraction candidates scored for noisy selection.
Abstraction length	Maximum 150 tokens; minimum 50 tokens	Controls the length of abstraction candidates.
Synthetic candidates per text C	4	Number of generated candidates by generator for each selected abstraction.
Generator settings	Temperature 0.5 (Mistral, 2025) for Mistral-Small; temperature 1.0 for Phi-4; repetition penalty 1.0	Fixed generator settings used across experiments.
Partner/crowd weight α	0.6	Balances partner similarity reduction and crowd similarity increase.
Crowd neighborhood size m	Top 8 other-user bundles	Computes crowd similarity using the nearest other-user bundle embeddings.
Near optimal margin τ	0.002	Defines the near optimal candidate set after bundle-level scoring.

Table 2: Models and fixed implementation settings for AAST. These values are held constant across all experiments.

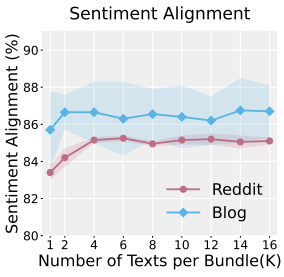


Figure 5: Sentiment alignment across bundle sizes and datasets.

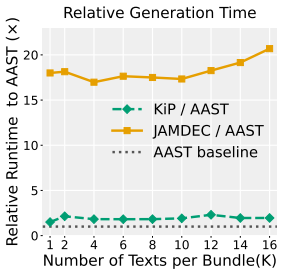


Figure 6: Relative generation time compared with AAST_{M}.

2024), which measures whether the true user appears among the top eight candidates. Lower values indicate lower attribution risk. To test transfer beyond LUAR, we also include an independent classical stylometric attribution attack (Basile et al., 2017; Pedregosa et al., 2011). This attacker uses character 2–5-grams, distance-character 1–3-grams, and word 1–3-grams, and ranks candidate users for each synthetic bundle. This evaluation tests whether AAST's attribution gains transfer from the LUAR-based attacker-aware setting to a non-neural stylometric attacker.

PAN22 verification decides whether two bundles are written by the same user (Stamatatos et al., 2022). We report Area Under the Curve (AUC) (Stamatatos et al., 2022) and Correctness at One (c@1) (Peñas and Rodrigo, 2011). For privacy, AUC closer to random chance 50% and lower c@1 indicate weaker verification. We also include a classical stylometric transfer attack to test whether the same synthetic outputs reduce verification risk under a different attack family (Basile et al., 2017; Pedregosa et al., 2011). The verifier represents each bundle with character 3–5-gram term frequency–inverse document frequency features and trains a logistic regression classifier on same author and different author bundle pairs. We split users into two disjoint halves, train on one half, and evaluate on the held-out half.

Following KiP (Bao and Carpuat, 2024), we evaluate bundle sizes $K \in$

$\{1, 2, 4, 6, 8, 10, 12, 14, 16\}$, where K is the number of texts from the same user grouped into each bundle. Each text is one post or blog entry. This sweep measures how privacy risk changes as the attacker observes more texts jointly.

For cross-genre evaluation, we use SELMA, the embedding authorship model introduced with CROSSNEWS (Ma et al., 2025). Following CROSSNEWS, we use the strongest SELMA prompt variants for each task. For attribution, we use SELMA+TaskOnly and SELMA+LIP; for verification, we use SELMA+TaskOnly and SELMA+PromptAV (Hung et al., 2023; Huang et al., 2024a; Ma et al., 2025). We report the same metrics as CROSSNEWS: top one Accuracy, R@8, and Average Rank for attribution, and Accuracy and F1 for verification.

C.2 Utility Metrics Details

We evaluate synthetic text quality along three dimensions. Semantic quality is measured by Natural Language Inference (NLI) (Liu et al., 2022). Given a private text and its synthetic counterpart, NLI measures whether the synthetic text preserves the core meaning of the private text.

Linguistic acceptability is measured by Corpus of Linguistic Acceptability (CoLA) (Warstadt et al., 2019). CoLA serves as an indicator of grammaticality and fluency.

Sentiment alignment measures whether the synthetic text preserves the sentiment polarity of the private text. We use the sentiment classifier from §4.4 to assign sentiment labels to private and synthetic texts (Hartmann et al., 2023), and report the proportion of matched labels.

D Additional Same-genre Authorship Results

Tables 3 and 4 report the full same-genre authorship attribution and verification risk results on Reddit and Blog across bundle sizes.

E Additional Same-genre Stylometric Attribution Results

Table 5 reports same-genre attribution results under a classical stylometric transfer attack, with 95% bootstrap confidence intervals computed over users. This unseen stylometric attacker evaluates whether AAST's privacy gains transfer beyond the LUAR attribution model used in bundle-level selection.

K	Method	Reddit		Blog	
		MRR(↓)	R@8(↓)	MRR(↓)	R@8(↓)
1	Private	32.2	45.6	38.5	52.4
	AAST_Φ	2.4	3.6	**1.7**	**2.0**
	AAST_M	**2.2**	**2.9**	2.6	2.4
	JAMDEC	4.5	8.4	7.4	12.8
	KiP	9.4	17.6	8.2	12.8
2	Private	67.1	81.2	61.2	72.4
	AAST_Φ	5.5	7.3	**3.2**	5.7
	AAST_M	**3.2**	**3.8**	3.3	**4.9**
	JAMDEC	9.9	14.0	12.5	22.0
	KiP	20.2	38.4	17.3	28.8
4	Private	89.8	98.0	83.1	91.2
	AAST_Φ	6.7	11.8	**7.0**	**11.5**
	AAST_M	**5.6**	**9.1**	8.6	15.9
	JAMDEC	18.1	27.2	20.3	35.6
	KiP	42.8	63.6	28.4	49.6
6	Private	96.0	99.2	88.7	96.0
	AAST_Φ	11.8	**19.3**	**8.8**	**14.9**
	AAST_M	**11.2**	19.7	15.6	21.8
	JAMDEC	24.5	38.0	28.4	44.8
	KiP	53.8	72.0	45.9	66.8
8	Private	97.8	100	93.8	98.0
	AAST_Φ	**17.0**	31.8	**14.9**	**21.3**
	AAST_M	18.4	**28.4**	24.9	40.3
	JAMDEC	24.4	41.6	33.0	52.0
	KiP	68.0	86.4	60.3	73.2
10	Private	99.3	100	95.3	99.2
	AAST_Φ	**25.5**	**39.7**	**20.7**	**34.3**
	AAST_M	25.9	43.9	34.1	48.4
	JAMDEC	36.0	56.4	43.5	63.6
	KiP	73.3	90.0	66.6	78.8
12	Private	99.5	100	95.8	98.8
	AAST_Φ	33.9	**49.8**	**24.4**	**38.0**
	AAST_M	**31.6**	50.5	36.3	56.7
	JAMDEC	39.6	60.0	50.5	70.4
	KiP	79.1	91.2	73.0	86.8
14	Private	99.6	100	97.0	99.2
	AAST_Φ	**35.5**	**54.2**	**31.4**	**48.0**
	AAST_M	37.4	55.1	42.9	62.5
	JAMDEC	52.6	72.4	54.7	80.4
	KiP	82.7	93.6	76.6	92.0
16	Private	99.8	100	97.5	99.2
	AAST_Φ	**39.8**	**60.7**	**41.1**	**61.4**
	AAST_M	43.5	63.7	44.5	65.7
	JAMDEC	54.7	74.8	62.4	83.6
	KiP	92.7	97.8	83.5	94.0

Table 3: Attribution risk results on Reddit and Blog across bundle sizes.

The results show that AAST remains effective under this independent attacker. Both AAST variants reduce attribution risk compared with KiP and JAMDEC, especially at larger bundle sizes. At K=16, AAST gives the lowest and second-lowest MRR and R@8 on both datasets, while KiP and JAMDEC remain substantially more linkable. These results suggest that AAST's protection is not specific to LUAR and transfers to a classical stylometric attribution setting.

F Additional Same-genre Stylometric Verification Results

Table 6 reports same-genre verification results under the classical stylometric transfer attack. Figure 3 shows the corresponding trends. The results are consistent with the main PAN22 verification

K	Method	Reddit		Blog	
		AUC(≈50)	c@1(↓)	AUC(≈50)	c@1(↓)
1	Private	51.9	52.9	64.6	60.6
	AAST_{Φ}	**50.6**	51.7	48.3	**48.6**
	AAST_{M}	48.9	**44.6**	**47.6**	49.3
	JAMDEC	51.2	52.0	60.8	53.4
	KiP	51.3	52.6	63.2	57.7
2	Private	60.9	58.9	68.9	60.3
	AAST_{Φ}	57.4	50.3	**53.9**	**47.4**
	AAST_{M}	**47.9**	**49.4**	56.5	48.4
	JAMDEC	54.3	51.6	61.1	51.4
	KiP	62.5	52.8	65.8	51.8
4	Private	72.6	66.3	72.5	67.6
	AAST_{Φ}	61.7	48.3	58.3	**48.2**
	AAST_{M}	**55.5**	**48.2**	**53.7**	48.7
	JAMDEC	69.8	55.6	62.4	57.2
	KiP	70.9	60.7	67.4	57.2
6	Private	73.7	70.0	75.0	67.4
	AAST_{Φ}	67.7	49.4	62.1	**48.8**
	AAST_{M}	**65.8**	**48.4**	**55.7**	49.0
	JAMDEC	69.7	52.6	61.6	55.1
	KiP	72.3	66.1	69.5	55.3
8	Private	78.8	71.7	79.3	72.9
	AAST_{Φ}	65.1	48.3	69.9	48.4
	AAST_{M}	**64.2**	**48.2**	**62.1**	**48.2**
	JAMDEC	77.0	52.0	65.3	57.7
	KiP	77.4	64.2	75.5	59.0
10	Private	77.0	69.3	81.7	74.6
	AAST_{Φ}	69.3	49.7	67.0	50.4
	AAST_{M}	**61.2**	**47.5**	**66.3**	**49.3**
	JAMDEC	74.7	55.9	70.7	59.6
	KiP	73.6	67.6	79.5	58.9
12	Private	82.3	76.9	81.6	78.4
	AAST_{Φ}	**64.6**	48.2	**67.3**	**48.2**
	AAST_{M}	66.9	**48.0**	67.4	49.2
	JAMDEC	79.3	64.2	71.3	57.0
	KiP	80.5	71.7	80.9	66.4
14	Private	86.6	82.8	82.8	81.6
	AAST_{Φ}	69.0	48.8	**67.5**	51.4
	AAST_{M}	**66.0**	**48.0**	67.8	**49.5**
	JAMDEC	81.1	69.6	72.8	57.7
	KiP	84.1	79.2	83.0	68.3
16	Private	89.7	83.6	82.3	79.1
	AAST_{Φ}	66.7	49.8	71.2	53.2
	AAST_{M}	**64.1**	**48.0**	**68.6**	**50.3**
	JAMDEC	81.4	73.1	73.3	61.4
	KiP	92.6	86.1	85.3	68.1

Table 4: Verification risk results on Reddit and Blog across bundle sizes.

results: AAST keeps verification risk lower than JAMDEC and KiP across most bundle sizes on both Reddit and Blog.

At larger K, the gap becomes clearer. On Reddit at K=16, AAST_{M} gives 71.6 AUC and 52.4 c@1, while JAMDEC gives 90.1 AUC and 75.3 c@1, and KiP gives 84.2 AUC and 67.0 c@1. On Blog at K=16, AAST_{Φ} gives the lowest AUC 68.6, while AAST_{M} gives 72.1 AUC and 55.2 c@1. In contrast, JAMDEC gives 94.0 AUC and 84.0 c@1, and KiP gives 96.1 AUC and 74.4 c@1.

These results provide additional transfer evidence that AAST's verification gains are not limited to neural authorship models, but persist under a classical stylometric attack. Together with the stylometric attribution results in Appendix E, they support our design goal: AAST reduces account-level authorship linkability while preserving useful text content instead of trying to remove all information from the synthetic text.

G Additional Same-genre Utility Results

G.1 Additional Reddit Utility Results

Table 7 reports the NLI and CoLA results on Reddit, while Table 8 provides the sentiment alignment results.

G.2 Blog Utility Results

AAST also maintains strong NLI and CoLA on Blog, as reported in Figure 4c and Table 7. AAST_{Φ} gives the best CoLA score across all bundle sizes, and its NLI becomes the strongest or tied for the strongest from K=4 onward. KiP has competitive NLI at smaller bundle sizes, but its CoLA scores are much lower and its linkability remains high. JAMDEC has lower NLI than AAST across all bundle sizes. Together with the privacy results, these utility scores show that AAST preserves content and linguistic quality while keeping authorship risk lower under aggregation.

AAST also preserves sentiment well on Blog. Figure 5 and Table 8 show that AAST_{M} and AAST_{Φ} maintain stable sentiment alignment across bundle sizes. AAST_{M} is especially stable, staying around 88%, while AAST_{Φ} stays around 85% on average. This suggests that AAST largely preserves the sentiment of the original blog entries while reducing authorship linkability.

H Additional LLM Baseline Comparison

Table 9 compares AAST with RUPTA, an additional LLM-based text anonymization baseline, on Reddit and Blog across K=1,8,16. RUPTA_{Φ} and RUPTA_{M} denote RUPTA using Phi-4 and Mistral-Small as the generator, respectively. Utility is not reported for Private.

The results show that RUPTA can preserve semantic utility, but it does not consistently reduce account-level authorship risk under aggregation. The higher NLI of some RUPTA variants should be interpreted together with their privacy scores: staying closer to the private text can preserve semantic overlap, but can also preserve authorial cues and increase linkability, similar to the pattern observed for KiP. AAST does not simply maximize NLI; it provides a stronger privacy–utility balance by keeping attribution and verification risk much lower while maintaining strong linguistic

K	Method	Reddit		Blog	
		MRR(↓)	R@8(↓)	MRR(↓)	R@8(↓)
1	Private	17.5 [14.1, 20.9]	28.0 [23.6, 32.8]	14.2 [11.2, 17.4]	25.6 [21.2, 30.4]
	AAST_{Φ}	**3.9** [2.6, 5.4]	6.4 [4.6, 8.6]	3.4 [2.2, 4.8]	**3.6** [2.4, 5.2]
	AAST_{M}	5.3 [3.8, 6.8]	8.9 [6.6, 11.6]	**2.4** [1.6, 3.4]	**3.6** [2.4, 5.0]
	JAMDEC	4.2 [2.8, 6.0]	**5.6** [3.8, 7.8]	7.9 [5.8, 10.4]	16.0 [12.6, 19.8]
	KiP	7.8 [5.6, 10.2]	11.6 [8.6, 14.8]	5.1 [3.6, 6.8]	8.8 [6.4, 11.6]
8	Private	52.0 [46.0, 58.2]	70.4 [64.0, 76.8]	45.7 [40.2, 51.6]	66.8 [60.8, 72.8]
	AAST_{Φ}	**8.0** [6.0, 10.4]	**15.6** [12.0, 19.6]	11.3 [8.6, 14.4]	19.2 [15.0, 23.8]
	AAST_{M}	10.9 [8.4, 13.6]	18.4 [14.6, 22.6]	**7.6** [5.6, 9.8]	**14.4** [11.0, 18.0]
	JAMDEC	13.8 [10.6, 17.4]	24.8 [20.0, 29.8]	25.8 [21.0, 30.8]	44.4 [38.6, 50.4]
	KiP	21.1 [16.8, 25.6]	38.0 [32.0, 44.4]	25.9 [21.0, 31.0]	42.4 [36.4, 48.6]
16	Private	59.8 [53.0, 66.6]	79.2 [73.0, 85.0]	62.6 [56.0, 69.4]	84.0 [78.0, 89.2]
	AAST_{Φ}	13.8 [10.6, 17.6]	**22.4** [17.8, 27.4]	18.1 [14.2, 22.4]	30.4 [25.0, 36.2]
	AAST_{M}	**12.6** [9.8, 15.6]	25.6 [20.6, 30.8]	**11.8** [9.0, 14.8]	**21.2** [17.0, 25.6]
	JAMDEC	21.0 [17.0, 25.4]	39.8 [34.2, 45.6]	30.1 [25.0, 35.6]	52.0 [46.0, 58.2]
	KiP	44.6 [38.0, 51.6]	64.3 [57.0, 71.6]	42.8 [36.4, 49.6]	66.0 [59.0, 72.8]

Table 5: Same-genre attribution results on Reddit and Blog under an independent classical n-gram stylometric attacker. The attributor uses character 2–5-grams, distance-character 1–3-grams, and word 1–3-grams. Values are percentages with 95% bootstrap confidence intervals computed over users.

quality and reasonable semantic preservation. As K grows, both RUPTA variants show high attribution risk, often approaching Private or KiP. In contrast, AAST maintains substantially lower R@8 and MRR across both datasets, especially at $K = 8$ and $K = 16$. A similar pattern appears in verification. AAST keeps AUC closer to 50 and c@1 lower than RUPTA in most large-bundle settings.

These results suggest that strong LLM-based rewriting alone is not sufficient for aggregation-level privacy. The bundle-level design of AAST is important for reducing linkability across a released account history.

I Additional Cross-genre Verification Results

The verification results in Table 10 follow the same cross-genre trend, with AAST lowering the verifier's ability to link synthetic and reference texts. In Tweet–Article, AAST_{M} gives the lowest F1 under both SELMA + TaskOnly and SELMA + PromptAV. Its binary verification Accuracy is also the best or second best across the two attacks. The reduction is clearer in Article–Tweet. AAST_{M} gives the lowest Accuracy and F1 under both attacks, with especially low F1 under SELMA + PromptAV. These results show that AAST reduces not only cross-genre attribution risk, but also the verifier's ability to decide whether synthetic text and reference text come from the same author.

The Article–Tweet setting is harder for AAST because tweets often contain short reactions, handles, hashtags, and URLs. We discuss this case in Appendix K.

J Cross-genre Utility Results

Table 11 shows that in both Tweet–Article and Article–Tweet, AAST_{M} achieves the highest CoLA score, indicating stronger linguistic quality than the baselines. AAST_{Φ} is consistently second best on CoLA, while KiP has much lower CoLA despite its higher NLI.

The cross-genre setting also shows a clear privacy utility tradeoff. KiP preserves more semantic content according to NLI, but its privacy results are weaker in both attribution and verification. JAMDEC has lower NLI and lower CoLA than AAST in both genre pairs. AAST_{M} gives the strongest balance in this setting. It substantially reduces cross-genre linkability while maintaining the best linguistic quality. Its lower NLI suggests that preserving exact semantic entailment across genre rewriting remains challenging, especially when converting between short tweets and longer articles.

K Discussion of the Article–Tweet Setting

Table 12 shows masked tweet inputs from one author. These examples illustrate a difficult case for bundle-level selection. Many tweet inputs contain little text beyond handles, URLs, hashtags, and other surface markers. After these markers are removed or normalized, the remaining text can be very short or generic, such as brief reactions or replies with only one word. Thus, there is less semantic and stylistic content for AAST to preserve and less variation for the bundle-level objective to exploit.

This pattern also helps explain why Article–

K	Method	Reddit		Blog	
		AUC ≈50	c@1 ↓	AUC ≈50	c@1 ↓
1	Private	54.2 [47.2, 61.0]	50.8 [44.4, 56.8]	60.0 [52.6, 66.7]	51.6 [45.6, 57.6]
	AAST_{M}	52.5 [45.5, 59.9]	49.2 [43.0, 55.4]	**50.2** [43.2, 57.3]	50.8 [44.8, 56.9]
	AAST_{Φ}	**49.8** [42.6, 57.5]	49.4 [43.0, 55.4]	50.3 [43.1, 57.9]	**49.2** [42.8, 55.2]
	JAMDEC	50.5 [43.3, 57.8]	**48.4** [42.4, 54.8]	64.0 [57.2, 70.4]	59.2 [52.8, 65.6]
	KiP	53.7 [46.2, 61.2]	48.8 [42.0, 54.8]	50.3 [42.8, 57.7]	50.0 [43.6, 56.0]
2	Private	69.8 [63.1, 76.3]	55.2 [48.4, 61.2]	75.9 [69.3, 81.5]	64.4 [58.0, 70.0]
	AAST_{M}	53.7 [46.3, 60.8]	**49.6** [43.2, 55.6]	49.0 [41.6, 56.1]	50.8 [44.0, 56.4]
	AAST_{Φ}	**50.6** [43.8, 58.3]	**49.6** [43.2, 55.6]	**49.2** [42.1, 56.5]	**49.6** [43.2, 55.6]
	JAMDEC	58.1 [50.7, 65.0]	52.0 [46.0, 57.6]	73.3 [66.7, 79.3]	63.6 [57.6, 69.2]
	KiP	57.4 [50.1, 64.4]	53.6 [47.2, 59.6]	57.9 [50.7, 65.3]	52.0 [45.2, 57.6]
4	Private	80.9 [75.4, 85.9]	62.4 [56.0, 68.4]	86.5 [81.9, 90.9]	76.4 [70.4, 81.6]
	AAST_{M}	**50.9** [43.5, 58.2]	52.8 [46.4, 58.8]	49.0 [41.7, 56.2]	51.2 [44.4, 57.2]
	AAST_{Φ}	55.1 [47.8, 62.1]	**50.4** [44.0, 56.4]	**49.5** [42.1, 56.6]	**48.8** [42.4, 54.4]
	JAMDEC	69.8 [62.8, 76.3]	58.8 [52.4, 64.8]	83.1 [78.2, 87.4]	70.8 [65.2, 76.4]
	KiP	59.9 [52.7, 66.7]	53.0 [47.0, 59.0]	66.2 [59.6, 72.5]	54.0 [47.6, 59.6]
6	Private	92.2 [89.1, 95.1]	74.0 [68.4, 79.6]	90.8 [87.5, 94.1]	80.0 [74.4, 84.8]
	AAST_{M}	**58.2** [51.1, 65.5]	**51.2** [44.8, 56.8]	53.2 [46.2, 60.0]	**50.0** [43.6, 56.0]
	AAST_{Φ}	63.3 [56.2, 69.7]	53.2 [47.2, 59.2]	**52.8** [45.2, 59.8]	**50.0** [43.6, 55.6]
	JAMDEC	73.9 [66.9, 80.4]	55.3 [48.4, 62.1]	84.2 [79.6, 88.7]	74.4 [68.8, 79.6]
	KiP	64.4 [56.2, 72.3]	54.5 [47.7, 61.2]	75.1 [68.8, 81.1]	58.4 [52.0, 64.0]
8	Private	93.3 [90.3, 96.1]	78.0 [72.8, 82.8]	93.8 [91.0, 96.2]	83.6 [78.4, 88.0]
	AAST_{M}	**61.4** [54.4, 68.5]	**50.8** [44.4, 56.8]	**55.0** [47.7, 62.3]	52.4 [46.0, 58.0]
	AAST_{Φ}	67.1 [60.4, 74.1]	53.6 [47.6, 59.6]	58.6 [51.4, 65.1]	**50.8** [44.0, 56.8]
	JAMDEC	79.8 [74.4, 85.1]	68.8 [63.2, 74.4]	87.5 [83.6, 91.5]	76.0 [70.8, 81.6]
	KiP	75.3 [69.6, 81.3]	57.2 [50.8, 63.2]	82.4 [77.3, 87.3]	61.2 [54.8, 67.2]
10	Private	94.1 [91.2, 96.6]	78.4 [73.2, 83.2]	95.8 [93.5, 97.6]	84.8 [80.4, 89.2]
	AAST_{M}	66.8 [60.4, 73.4]	**51.6** [45.2, 57.6]	59.8 [52.4, 66.9]	51.6 [45.2, 57.6]
	AAST_{Φ}	**66.0** [59.4, 73.2]	52.8 [46.8, 58.8]	**58.3** [50.6, 65.2]	**51.2** [44.8, 56.8]
	JAMDEC	80.6 [74.5, 86.7]	65.3 [58.4, 72.1]	89.5 [85.6, 93.1]	74.4 [68.4, 80.0]
	KiP	76.5 [70.0, 82.7]	60.6 [53.8, 67.5]	89.7 [85.9, 93.1]	62.4 [55.6, 68.4]
12	Private	94.5 [91.7, 97.0]	82.8 [78.4, 87.2]	97.2 [95.5, 98.6]	87.2 [82.8, 90.8]
	AAST_{M}	**64.7** [58.3, 71.5]	53.6 [47.2, 59.6]	65.3 [58.8, 72.3]	53.2 [46.4, 59.2]
	AAST_{Φ}	67.0 [60.7, 73.8]	**52.8** [47.2, 58.8]	**63.2** [55.6, 69.9]	**51.2** [44.4, 56.8]
	JAMDEC	82.5 [75.8, 88.0]	67.9 [61.0, 74.2]	91.3 [87.8, 94.6]	78.4 [73.2, 83.6]
	KiP	78.1 [71.4, 84.7]	62.1 [55.3, 68.9]	93.0 [90.0, 95.5]	68.0 [61.6, 73.6]
14	Private	95.3 [92.8, 97.6]	85.6 [81.2, 90.0]	98.0 [96.6, 99.1]	89.6 [85.6, 93.2]
	AAST_{M}	**68.5** [62.2, 75.1]	**51.6** [45.2, 57.6]	64.4 [57.2, 71.0]	**52.8** [46.4, 58.8]
	AAST_{Φ}	70.8 [64.5, 77.2]	55.6 [48.8, 61.6]	**64.0** [56.7, 70.8]	53.6 [47.2, 59.6]
	JAMDEC	88.8 [83.6, 93.3]	73.2 [66.3, 78.9]	92.8 [89.8, 95.8]	79.6 [74.4, 84.4]
	KiP	90.6 [86.3, 94.8]	66.3 [58.9, 73.2]	96.4 [94.4, 98.1]	70.8 [64.8, 76.4]
16	Private	96.1 [93.7, 98.2]	87.2 [83.2, 91.2]	98.5 [97.5, 99.4]	90.0 [86.0, 93.6]
	AAST_{M}	**71.6** [65.0, 77.6]	**52.4** [46.0, 58.4]	72.1 [65.4, 77.9]	55.2 [48.4, 61.2]
	AAST_{Φ}	75.6 [69.2, 81.1]	56.3 [50.0, 62.4]	**68.6** [61.7, 75.1]	**54.8** [48.4, 60.4]
	JAMDEC	90.1 [85.3, 94.2]	75.3 [68.7, 81.3]	94.0 [91.2, 96.6]	84.0 [79.2, 88.4]
	KiP	84.2 [72.9, 93.3]	67.0 [55.6, 78.4]	96.1 [93.8, 98.0]	74.4 [68.8, 79.6]

Table 6: Same-genre verification results under a classical stylometric transfer attack on Reddit and Blog. The verifier uses character 3–5-gram term frequency–inverse document frequency features with logistic regression. Values are percentages with 95% bootstrap confidence intervals.

Tweet differs from Tweet–Article in authorship attribution and verification risk. In Tweet–Article, AAST_{M} and AAST_{Φ} are usually the best two methods, showing that AAST has more room to preserve meaning and reduce author signals when generating longer article outputs. In Article–Tweet, AAST_{M} remains the strongest method, but AAST_{Φ} is less consistently the second best. This suggests that short tweet inputs with heavy metadata leave less room for AAST's bundle-level selection to balance privacy and utility. After handles, URLs, hashtags, and brief reactions are normalized, many tweet inputs contain limited text. Thus, the bundle-level signal can become weak or noisy, and AAST then relies more heavily on Candidate Preselection. The point is not that AAST fails in Article–Tweet. AAST_{M} still gives the strongest cross-genre privacy results in this setting.

L Ablation Analysis

Our ablation reports two analyses. First, Appendix L.1 studies the effects of Identifier Masking, Candidate Preselection, Candidate Refinement, and Bundle-Level Selection using AAST_{M} on Reddit across $K = 1, 8, 16$. Second, Appendix L.2 adds a controlled $\text{NoBundleSel}_{\Phi}$ comparison on Reddit and Blog to isolate the role of bundle-level selection under the main same-genre setting.

K	Method	Reddit		Blog	
		NLI(↑)	CoLA(↑)	NLI(↑)	CoLA(↑)
1	AAST_{Φ}	55.1	**94.7**	55.0	**93.6**
	AAST_{M}	53.4	93.1	51.6	93.4
	JAMDEC	20.2	91.4	20.0	90.0
	KiP	**65.4**	63.2	**58.1**	55.8
2	AAST_{Φ}	57.2	**94.9**	57.4	**93.9**
	AAST_{M}	56.4	94.1	54.5	93.0
	JAMDEC	22.7	92.3	26.9	91.8
	KiP	**65.5**	61.9	**59.4**	56.0
4	AAST_{Φ}	61.3	**94.8**	**62.5**	**93.8**
	AAST_{M}	60.5	94.2	61.5	92.9
	JAMDEC	28.8	92.2	35.0	92.0
	KiP	**65.9**	61.7	**62.5**	55.8
6	AAST_{Φ}	64.2	**94.9**	**66.1**	**93.8**
	AAST_{M}	62.9	94.2	64.4	93.1
	JAMDEC	34.3	92.2	38.7	85.6
	KiP	**68.2**	61.2	66.0	56.0
8	AAST_{Φ}	65.9	**94.8**	**68.9**	**93.8**
	AAST_{M}	65.7	94.1	67.5	92.9
	JAMDEC	37.1	92.2	45.4	86.1
	KiP	**69.1**	61.1	68.3	56.1
10	AAST_{Φ}	68.6	**94.7**	**71.2**	**93.7**
	AAST_{M}	67.7	94.1	70.2	92.8
	JAMDEC	40.6	91.9	44.7	81.9
	KiP	**70.9**	61.3	69.9	56.1
12	AAST_{Φ}	70.0	**94.7**	**72.8**	**93.8**
	AAST_{M}	69.7	94.1	72.0	93.0
	JAMDEC	43.3	89.9	47.8	82.8
	KiP	**72.3**	61.0	71.2	56.0
14	AAST_{Φ}	71.4	**94.8**	**74.5**	**93.8**
	AAST_{M}	71.1	94.0	73.5	93.0
	JAMDEC	43.3	90.4	51.1	82.9
	KiP	**73.4**	60.1	72.6	55.9
16	AAST_{Φ}	**72.9**	**94.7**	**75.7**	**93.8**
	AAST_{M}	72.6	94.0	74.7	92.9
	JAMDEC	45.8	86.8	51.6	82.7
	KiP	72.8	60.9	73.7	56.0

Table 7: NLI and CoLA results on Reddit and Blog across bundle sizes.

K	Method	Sentiment Alignment(%)(↑)	
		Reddit	Blog
1	AAST_{Φ}	83.0	83.6
	AAST_{M}	83.8	87.8
2	AAST_{Φ}	83.7	85.7
	AAST_{M}	84.7	87.6
4	AAST_{Φ}	85.3	85.0
	AAST_{M}	85.0	88.3
6	AAST_{Φ}	85.1	84.3
	AAST_{M}	85.4	88.3
8	AAST_{Φ}	84.8	85.2
	AAST_{M}	85.1	87.9
10	AAST_{Φ}	84.9	84.7
	AAST_{M}	85.4	88.1
12	AAST_{Φ}	84.9	84.9
	AAST_{M}	85.5	87.5
14	AAST_{Φ}	85.4	85.0
	AAST_{M}	84.7	88.5
16	AAST_{Φ}	85.3	85.3
	AAST_{M}	84.9	88.1

Table 8: Sentiment alignment results on Reddit and Blog across bundle sizes. Higher sentiment alignment indicates better preservation of private text sentiment.

L.1 Component Effects

L.1.1 Effect of Identifier Masking

Identifier Masking (§3.3) normalizes structured surface markers such as handles, URLs, hashtags, emails, numbers, emojis, and repeated punctuation. These markers do not always identify an author by themselves, but they can create direct lexical shortcuts or platform artifacts that affect generation and evaluation. In Table 15, enabling Identifier Masking lowers privacy risk on Reddit, however, it also comes with lower utility. Since AAST is designed to generate useful synthetic text, not only to remove all privacy signals, this utility drop matters for downstream use where preserving meaning is important.

Based on this tradeoff, we apply Identifier Masking selectively. In cross-genre settings involving tweet-like text, structured surface markers are frequent and often carry limited semantic content, so we enable Identifier Masking. For same-genre Reddit and Blog, these surface forms may carry useful content, and masking them can reduce semantic preservation. Therefore, we disable Identifier Masking in the reported same-genre settings and treat it as a conditional preprocessing module for cross-genre tweet settings.

L.1.2 Effect of Candidate Preselection

Candidate Preselection (§3.6) targets direct private–synthetic carryover. It compares each generated candidate with its corresponding private input and retains the candidates that are less similar to the private input before bundle-level selection.

The Reddit ablation in Table 15 measures synthetic bundle linkability, not direct private–synthetic similarity. Under this metric, disabling Candidate Preselection can lower attribution and verification risk. This happens because disabling Candidate Preselection leaves a larger candidate pool for Bundle-Level Selection, giving the bundle-level objective more freedom to choose candidates that reduce synthetic bundle linkability.

Thus, Candidate Preselection should not be interpreted as the main source of same-genre bundle-level privacy gains. Its role is different: it reduces the chance that a generated candidate remains too close to its private input. We keep Candidate Preselection in the main AAST pipeline because this protection is important for stronger direct comparison settings and for cross-genre evaluations, where private or reference texts may be compared against released synthetic texts.

K	Method	Reddit						Blog					
		Attribution		Verification		Utility		Attribution		Verification		Utility	
		R@8(↓)	MRR(↓)	AUC(≈50)	c@1(↓)	NLI(↑)	CoLA(↑)	R@8(↓)	MRR(↓)	AUC(≈50)	c@1(↓)	NLI(↑)	CoLA(↑)
1	Private	45.6	32.2	51.9	52.9	–	–	52.4	38.5	64.6	60.6	–	–
	AAST_{Φ}	3.6	2.4	**50.6**	51.7	55.1	**94.7**	**2.0**	**1.7**	**48.3**	**48.6**	55.0	**93.6**
	AAST_{M}	**2.9**	**2.2**	48.9	**44.6**	53.4	93.1	2.4	2.6	47.6	49.3	51.6	93.4
	RUPTA_{Φ}	8.4	14.5	51.0	47.9	62.1	85.8	26.4	16.8	59.6	57.6	60.5	84.5
	RUPTA_{M}	40.6	25.0	52.2	52.3	**71.1**	79.7	48.8	34.9	62.4	59.6	**76.0**	77.8
	KiP	17.6	9.4	51.3	52.6	65.4	63.2	12.8	8.2	63.2	57.7	58.1	55.8
	JAMDEC	8.4	4.5	51.2	52.0	20.2	91.4	12.8	7.4	60.8	53.4	20.0	90.0
8	Private	100	97.8	78.8	71.7	–	–	98.0	93.8	79.3	72.9	–	–
	AAST_{Φ}	31.8	**17.0**	65.1	48.3	65.9	**94.8**	**21.3**	**14.9**	69.9	48.4	68.9	**93.8**
	AAST_{M}	**28.4**	18.4	**64.2**	**48.2**	65.7	94.1	40.3	24.9	**62.1**	**48.2**	67.5	92.9
	RUPTA_{Φ}	93.6	81.1	76.2	68.9	70.1	85.6	91.6	79.2	74.9	67.3	70.5	84.8
	RUPTA_{M}	99.2	94.6	71.8	65.5	**82.3**	79.4	98.0	93.1	77.5	69.1	**83.3**	78.1
	KiP	86.4	68.0	77.4	64.2	69.1	61.1	73.2	60.3	75.5	59.0	68.3	56.1
	JAMDEC	41.6	24.4	77.0	52.0	37.1	92.2	52.0	33.0	65.3	57.7	45.4	86.1
16	Private	100	99.8	89.7	83.6	–	–	99.2	97.5	82.3	79.1	–	–
	AAST_{Φ}	**60.7**	**39.8**	66.7	49.8	72.9	**94.7**	**61.4**	**41.1**	71.2	53.2	75.7	**93.8**
	AAST_{M}	63.7	43.5	**64.1**	**48.0**	72.6	94.0	65.7	44.5	**68.6**	**50.3**	74.7	92.9
	RUPTA_{Φ}	98.0	94.3	86.3	79.5	74.9	85.7	98.0	95.4	82.4	78.1	74.8	85.0
	RUPTA_{M}	100	98.0	81.9	74.5	**86.1**	79.2	99.4	98.2	86.2	83.0	**83.1**	78.3
	KiP	97.8	92.7	92.6	86.1	72.8	60.9	94.0	83.5	85.3	68.1	73.7	56.0
	JAMDEC	74.8	54.7	81.4	73.1	45.8	86.8	83.6	62.4	73.3	61.4	51.6	82.7

Table 9: Comparison with the LLM-based RUPTA baseline on Reddit and Blog across K=1,8,16. Utility is not reported for Private.

Genre Pair	Attack	Method	Acc.(↓)	F1(↓)
Tweet–Article	SELMA + TaskOnly	AAST_{M}	**54.8**	**54.3**
		AAST_{Φ}	55.6	53.6
		KiP	58.2	56.8
		JAMDEC	55.0	62.9
	SELMA + PromptAV	AAST_{M}	55.2	**53.9**
		AAST_{Φ}	55.9	56.2
		KiP	58.4	59.2
		JAMDEC	**55.0**	62.5
Article–Tweet	SELMA + TaskOnly	AAST_{M}	**51.7**	**41.7**
		AAST_{Φ}	53.7	45.5
		KiP	56.3	47.8
		JAMDEC	53.3	50.8
	SELMA + PromptAV	AAST_{M}	**51.9**	**34.1**
		AAST_{Φ}	53.6	42.5
		KiP	56.3	47.5
		JAMDEC	53.1	50.2

Table 10: Cross-genre authorship verification risk results.

Genre Pair	Method	NLI(↑)	CoLA(↑)
Tweet – Article	AAST_{M}	27.6	**89.6**
	AAST_{Φ}	30.9	83.2
	KiP	**53.8**	53.9
	JAMDEC	18.3	71.4
Article – Tweet	AAST_{M}	14.9	**86**
	AAST_{Φ}	18.0	80.3
	KiP	**32.8**	47.7
	JAMDEC	10.8	59.5

Table 11: Cross-genre NLI and CoLA results.

ID	Masked Tweet Example
1	`@[USER1] @[USER2] @[USER3] @[USER4] @[USER5] @[USER6] Congratulations to all fellow winners`
2	`#NewProfilePic [URL]`
3	`@[USER7] No`
4	`@[USER8] haha`

Table 12: Masked examples of tweet inputs from one author. Handles and URLs are replaced with placeholders.

L.1.3 Effect of Candidate Refinement and Bundle-Level Selection

Candidate Refinement (§3.7.2) and Bundle-Level Selection (§3.7) are the main components for controlling authorship leakage. The setting without Candidate Refinement removes the refinement step after Bundle-level Candidate Scoring (§3.7.1), while the setting without Bundle-Level Selection removes both Bundle-level Candidate Scoring and Candidate Refinement. Removing Candidate Refinement sharply increases verification risk. At K=16, AUC rises from 51.3 to 78.5, and c@1 rises from 46.5 to 55.9. Removing Bundle-Level Selection causes an even larger degradation. At K=16, R@8 rises to 75.9, MRR rises to 55.9, AUC rises to 83.6, and c@1 rises to 75.7. Although these settings obtain higher utility scores, they leave strong bundle-level author signals. These results show that AAST needs Bundle-Level Selection over the generated pool, not only independent rewriting for each text.

In summary, the ablation shows that the modules address different risks. Identifier Masking

K	σ		
	ϵ = 4	ϵ = 2	ϵ = 1
1	1.02	2.03	4.07
8	1.15	2.31	4.61
16	1.19	2.39	4.78

Table 13: Gaussian noise standard deviations σ used for noisy abstraction selection.

Bundle Size K	**Total Tokens** ($\times 10^6$)
1	0.50
2	1.01
4	2.00
6	3.00
8	3.99
10	4.99
12	6.00
14	6.98
16	7.97

Table 14: Token usage of AAST_{M} on Blog across bundle sizes. Counts include both input and output tokens for Mistral generation.

normalizes structured surface markers when they are frequent and low in semantic content. Candidate Preselection reduces direct private–synthetic carryover. Bundle-Level Selection provides the main protection against bundle-level linkability as K increases.

L.2 Effect of Removing Bundle-Level Selection

Figure 7 and Table 16 compare AAST_{Φ} with $\text{NoBundleSel}_{\Phi}$ on Reddit and Blog. $\text{NoBundleSel}_{\Phi}$ keeps the same settings as AAST_{Φ}, but removes Bundle-level Candidate Scoring and Candidate Refinement. This controlled comparison tests whether the privacy gains come from bundle-level selection, not only from the LLM generator, prompt, candidate pool, or abstraction model.

Removing bundle-level selection largely increases attribution risk as K grows. On Reddit, $\text{NoBundleSel}_{\Phi}$ increases R@8 from 14.5 at K=1 to 75.1 at K=16, while AAST_{Φ} stays lower, from 3.6 to 60.7. The same pattern appears on Blog, where $\text{NoBundleSel}_{\Phi}$ reaches 73.2 R@8 at K=16, compared with 61.4 for AAST_{Φ}. Verification risk also increases after removing bundle-level selection. At K=16, $\text{NoBundleSel}_{\Phi}$ reaches 85.8 AUC and 65.6 c@1 on Reddit, while AAST_{Φ} gives 66.7 AUC and 49.8 c@1. On Blog, $\text{NoBundleSel}_{\Phi}$ reaches 82.5 AUC and 68.2 c@1, compared with 71.2 AUC and 53.2 c@1 for AAST_{Φ}.

In contrast, the utility scores remain close. Across both datasets, NLI and CoLA are similar between AAST_{Φ} and $\text{NoBundleSel}_{\Phi}$. This shows that bundle-level selection is the main source of the privacy gain, and that this gain does not come from a meaningful loss in semantic or linguistic utility.

The gap between AAST and NoBundleSel is not meant to show that bundle-level selection removes all aggregation risk, especially at the largest bundle sizes. Instead, the result should be read as a privacy–utility tradeoff. At K=16, AAST_{Φ} still preserves high utility while reducing attribution and verification risk relative to $\text{NoBundleSel}_{\Phi}$. This supports our design goal: AAST reduces account-level linkability without sacrificing semantic and linguistic quality.

M Effect of Noisy Selection

We also study AAST_{Φ} with Gaussian noise added to the abstraction score selection step under $\epsilon \in \{4, 2, 1\}$. The noise perturbs the normalized candidate scores before one abstraction candidate is selected. For each bundle size K, we set $\delta = 1/(N \log N)$, where N is the number of private texts entering the noisy selection step. In our setting, each experiment uses 250 users with two bundles per user and K texts per bundle, so $N = 250 \times 2 \times K$. This gives $N = 500, 4000$, and 8000 for $K = 1, 8, 16$, respectively. Table 13 reports the resulting Gaussian noise standard deviations σ.

Table 17 shows how noisy abstraction selection affects AAST_{Φ}. The effect becomes clearer as aggregation size increases. At K=8 and K=16, stronger noise generally lowers attribution risk compared with $\epsilon = \infty$, which denotes no noise. On Reddit, R@8 decreases from 31.8 to 22.7 at K=8 and from 60.7 to 53.4 at K=16. On Blog, R@8 decreases from 21.3 to 19.6 at K=8 and from 61.4 to 54.2 at K=16. MRR follows a similar pattern. Verification risk also tends to move closer to the desired range.

The noise introduces a utility tradeoff. The setting with no noise usually gives the highest NLI because it selects the abstraction candidate with the strongest semantic and sentiment score. As noise increases, the selected abstraction can move away from this top candidate.

These results show that noisy abstraction selection can reduce bundle-level linkability, especially at larger K, with a small drop in semantic preservation. This analysis is limited to the abstraction score selection step and does not claim end to end

Ablation configuration	K	Attribution		Verification		Utility	
		R@8(↓)	MRR(↓)	AUC(≈50)	c@1(↓)	NLI(↑)	CoLA(↑)
Identifier Masking(§3.3) enabled	1	0	1.2	55.6	52.6	18.8	92.5
	8	3.3	2.0	55.8	46.4	38.4	92.8
	16	13.5	8.2	51.3	46.5	50.6	92.7
Identifier Masking disabled	1	2.9	2.2	48.9	44.6	53.4	93.1
	8	28.4	18.4	64.2	48.2	65.7	94.1
	16	63.7	43.5	64.1	48.0	72.6	94.0
Candidate Preselection(§3.6) disabled	1	0	0.9	48.5	46.6	18.9	92.1
	8	0.8	1.3	42.5	47.1	38.6	92.5
	16	6.3	4.6	41.6	47.5	50.8	92.5
Candidate Refinement(§3.7.2) disabled	1	2.5	2.3	52.7	47.7	53.9	93.0
	8	20.7	11.6	75.2	52.5	65.5	94.2
	16	24.1	14.8	78.5	55.9	72.1	94.1
Bundle-Level Selection(§3.7) disabled	1	10.7	7.5	53.4	50.7	52.2	93.1
	8	57.9	39.1	75.8	69.4	74.6	94.2
	16	75.9	55.9	83.6	75.7	71.7	94.1

Table 15: Ablation study of AAST components on Reddit across $K = 1, 8, 16$. The first two row groups compare Identifier Masking configurations. The last three row groups disable one component relative to the Identifier Masking enabled configuration. In the reported main experiments, same-genre Reddit and Blog use Identifier Masking disabled, while cross-genre Article–Tweet and Tweet–Article use Identifier Masking enabled. Candidate Preselection, Refinement, and Bundle-Level Selection are enabled in all reported main AAST settings. Bundle-Level Selection consists of Bundle-level Candidate Scoring followed by Candidate Refinement.

DP for candidate generation or the final synthetic corpus.

N Qualitative Evaluation

N.1 Observed Strengths and Weaknesses

Table 18 summarizes the main strengths and weaknesses observed from our quantitative results and qualitative examples. AAST is designed for aggregation-aware privacy and gives the strongest overall balance. KiP often preserves local meaning but can retain private wording and author signals. JAMDEC can move farther from the private text, but this often comes with semantic drift and higher generation cost.

N.2 Example Comparison

Table 19 and Table 20 provide qualitative examples on Reddit at K=2. The private bundle contains two query texts and two target texts from the same user. We bold key semantic elements in the private texts and the corresponding preserved or rewritten content in the synthetic outputs.

The examples show that AAST preserves the main meaning of the private texts while changing the surface form. For instance, both AAST variants retain the mother's Parkinson's diagnosis, the need for medical resources, the loss of faith, the fear of death, the work and tuition constraint, and the emotional harm caused by early return from a mission. These outputs do not copy the private texts directly, but they keep the central events, sentiment, and support seeking intent.

KiP keeps many details from the private texts, which explains why its utility can appear strong in a local semantic comparison. However, it often remains close to the original wording and structure, so it does not sufficiently reduce authorship signals under aggregation. This helps explain why KiP remains much closer to the private text in the privacy metrics. Its low CoLA score is also visible in the examples, where words are frequently split into unnatural fragments such as "griev ing", "s an ity", and "in sensitive", making the text less fluent.

JAMDEC shows the opposite failure pattern. It can reduce linkability because the generated text often drifts away from the original meaning. In several examples, the core private information is lost, such as the Parkinson's diagnosis, the religious grief, the tuition repayment constraint, and the mission related mental health context. This semantic drift may help privacy scores, but it weakens the utility of the synthetic text for downstream applications. Thus, the qualitative examples support the quantitative trend that AAST better balances useful meaning preservation with reduced aggregation based authorship risk.

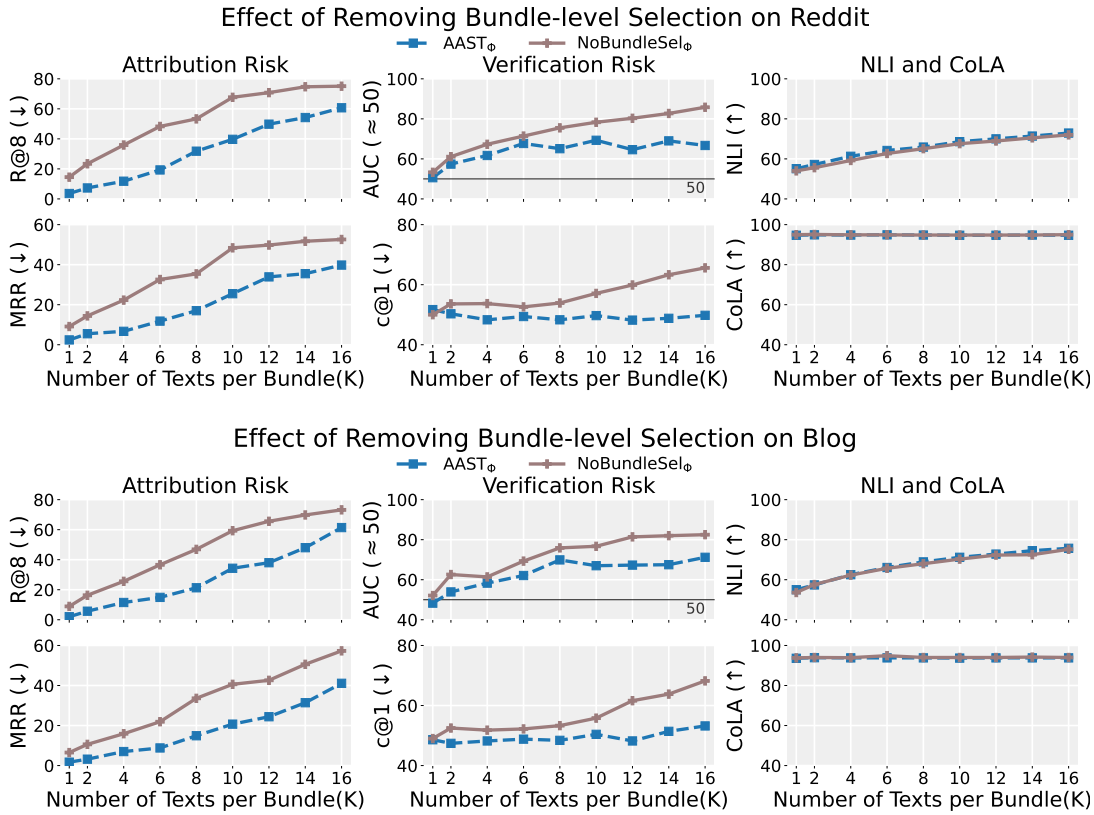


Figure 7: Effect of removing bundle-level selection on Reddit and Blog. NoBundleSel$_\Phi$ keeps the same settings as AAST$_\Phi$, but removes Bundle-level Candidate Scoring and Candidate Refinement.

O Generation Cost

O.1 Computational Efficiency Results

We evaluate computational cost by measuring the time required to generate synthetic text for 250 Blog users across K=1 to K=16. AAST uses Mistral-Small as the generator. For each method, synthetic text generation was distributed across the same three GPUs, including two NVIDIA A6000 GPUs with 48GB VRAM and one NVIDIA A100 GPU with 40GB VRAM. Because JAMDEC requires substantially longer generation time, reaching several days at larger K, we report runtime relative to AAST instead of only listing raw GPU hours.

Figure 6 shows that AAST is consistently the fastest method across all bundle sizes. KiP requires about $1.5\times$ to $2.3\times$ the runtime of AAST. JAMDEC is much more expensive, requiring about $18.5\times$ the runtime of AAST on average. At K=16, AAST completes generation in 5.8 hours, while KiP takes 11.3 hours and JAMDEC takes about 120 hours. These results show that AAST is more practical for downstream applications that require large scale synthetic text generation, as it scales efficiently with aggregation size while preserving the generation quality needed for privacy and utility evaluation.

We also report output length to make the runtime comparison easier to interpret. In the Reddit K=1 setting, each method generates 500 synthetic texts. The average private text length is 159.1 words, while the average synthetic lengths for AAST, JAMDEC, and KiP are 106.8, 118.5, and 139.2 words, respectively. KiP outputs are about $1.3\times$ longer than AAST outputs, and JAMDEC outputs are about $1.1\times$ longer. These length differences are much smaller than the runtime gaps, where KiP is about $1.9\times$ slower than AAST and JAMDEC is about $18.5\times$ slower. This suggests that AAST's efficiency comes mainly from its simpler generation pipeline. The AAST output length reflects the prompt design in Appendix B.1, which prioritizes preserving core meaning.

O.2 Token Cost Analysis

Table 14 reports the token usage of AAST$_\text{M}$ on Blog. Token usage grows almost linearly with bundle size because larger K means more private texts must be rewritten. At K=16, the setting contains

K	Method	Attribution				Verification				NLI and CoLA			
		Reddit		Blog		Reddit		Blog		Reddit		Blog	
		MRR↓	R@8↓	MRR↓	R@8↓	AUC≈50	c@1↓	AUC≈50	c@1↓	NLI↑	CoLA↑	NLI↑	CoLA↑
1	AAST_{Φ}	2.4	3.6	1.7	2.0	50.6	51.7	48.3	48.6	55.1	94.7	55.0	93.6
	$\text{NoBundleSel}_{\Phi}$	9.1	14.5	6.5	8.9	53.3	50.0	52.1	49.0	54.0	95.0	53.6	93.8
2	AAST_{Φ}	5.5	7.3	3.2	5.7	57.4	50.3	53.9	47.4	57.2	94.9	57.4	93.9
	$\text{NoBundleSel}_{\Phi}$	14.4	23.4	10.6	16.3	61.1	53.6	62.6	52.5	55.6	95.1	57.6	94.0
4	AAST_{Φ}	6.7	11.8	7.0	11.5	61.7	48.3	58.3	48.2	61.3	94.8	62.5	93.8
	$\text{NoBundleSel}_{\Phi}$	22.3	35.9	15.9	25.6	67.3	53.7	61.4	51.8	59.2	94.9	62.3	93.9
6	AAST_{Φ}	11.8	19.3	8.8	14.9	67.7	49.4	62.1	48.8	64.2	94.9	66.1	93.8
	$\text{NoBundleSel}_{\Phi}$	32.6	48.3	21.9	36.6	71.4	52.6	69.3	52.2	62.7	94.9	65.7	94.9
8	AAST_{Φ}	17.0	31.8	14.9	21.3	65.1	48.3	69.9	48.4	65.9	94.8	68.9	93.8
	$\text{NoBundleSel}_{\Phi}$	35.4	53.3	33.6	46.9	75.5	53.9	75.9	53.3	65.1	94.8	68.0	94.0
10	AAST_{Φ}	25.5	39.7	20.7	34.3	69.3	49.7	67.0	50.4	68.6	94.7	71.2	93.7
	$\text{NoBundleSel}_{\Phi}$	48.4	67.7	40.6	59.3	78.3	57.1	76.7	55.8	67.6	94.8	70.3	94.0
12	AAST_{Φ}	33.9	49.8	24.4	38.0	64.6	48.2	67.3	48.2	70.0	94.7	72.8	93.8
	$\text{NoBundleSel}_{\Phi}$	49.8	70.8	42.6	65.6	80.3	59.9	81.4	61.6	68.9	94.8	72.3	94.0
14	AAST_{Φ}	35.5	54.2	31.4	48.0	69.0	48.8	67.5	51.4	71.4	94.8	74.5	93.8
	$\text{NoBundleSel}_{\Phi}$	51.7	74.7	50.6	69.8	82.7	63.3	82.0	63.8	70.5	94.8	72.5	94.2
16	AAST_{Φ}	39.8	60.7	41.1	61.4	66.7	49.8	71.2	53.2	72.9	94.7	75.7	93.8
	$\text{NoBundleSel}_{\Phi}$	52.6	75.1	57.3	73.2	85.8	65.6	82.5	68.2	72.0	95.0	75.2	94.0

Table 16: Results for the effect of removing bundle-level selection on Reddit and Blog. $\text{NoBundleSel}_{\Phi}$ keeps the same settings as AAST_{Φ}, but removes Bundle-level Candidate Scoring and Candidate Refinement.

250 users with two bundles per user and 16 texts per bundle, resulting in $8{,}000$ private texts to generate. This produces about 7.97 million input and output tokens.

This cost should be interpreted in light of the corpus scale. The large token count at higher K comes from generating a full synthetic corpus for thousands of private texts. Thus, the token cost mainly reflects the number of texts that must be rewritten under the aggregation setting.

Dataset	K	Evaluation	Metric	$\epsilon = \infty$	$\epsilon = 4$	$\epsilon = 2$	$\epsilon = 1$
Reddit	1	Attribution	MRR(↓)	2.4	3.1	2.7	3.0
			R@8(↓)	3.6	4.0	2.8	3.2
		Verification	AUC(≈50)	50.6	56.0	58.5	54.4
			c@1(↓)	51.7	50.8	53.4	48.7
		Utility	NLI(↑)	55.1	53.4	53.7	52.3
			CoLA(↑)	94.7	94.9	94.9	94.8
			Sentiment(↑)	83.0	78.6	81.0	79.6
	8	Attribution	MRR(↓)	17.0	16.7	15.9	13.2
			R@8(↓)	31.8	30.0	25.9	22.7
		Verification	AUC(≈50)	65.1	66.7	65.6	65.0
			c@1(↓)	48.3	55.1	48.4	48.8
		Utility	NLI(↑)	65.9	64.9	65.1	64.8
			CoLA(↑)	94.8	94.8	94.9	94.8
			Sentiment(↑)	84.8	81.0	81.0	81.2
	16	Attribution	MRR(↓)	39.8	37.9	35.4	35.0
			R@8(↓)	60.7	58.3	55.7	53.4
		Verification	AUC(≈50)	66.7	67.6	67.0	65.5
			c@1(↓)	49.8	50.2	47.8	48.4
		Utility	NLI(↑)	72.9	71.9	71.6	71.6
			CoLA(↑)	94.7	94.8	94.8	94.8
			Sentiment(↑)	85.3	81.6	81.9	81.6
Blog	1	Attribution	MRR(↓)	1.7	2.2	2.5	2.2
			R@8(↓)	2.0	3.2	3.2	2.4
		Verification	AUC(≈50)	48.3	50.4	50.8	51.5
			c@1(↓)	48.6	51.6	51.8	49.5
		Utility	NLI(↑)	55.0	52.8	52.9	53.0
			CoLA(↑)	93.6	94.2	94.0	94.4
			Sentiment(↑)	83.6	84.6	86.0	87.2
	8	Attribution	MRR(↓)	14.9	13.0	13.5	12.0
			R@8(↓)	21.3	22.6	20.9	19.6
		Verification	AUC(≈50)	69.9	60.0	66.3	62.3
			c@1(↓)	48.4	49.6	50.8	48.2
		Utility	NLI(↑)	68.9	67.9	67.5	67.4
			CoLA(↑)	93.8	93.8	94.0	93.9
			Sentiment(↑)	85.2	84.2	84.1	84.4
	16	Attribution	MRR(↓)	41.1	34.3	37.4	35.2
			R@8(↓)	61.4	51.1	53.3	54.2
		Verification	AUC(≈50)	71.2	70.7	70.0	70.6
			c@1(↓)	53.2	51.0	49.6	50.3
		Utility	NLI(↑)	75.7	75.2	75.2	74.9
			CoLA(↑)	93.8	93.9	94.0	94.0
			Sentiment(↑)	85.3	84.6	85.3	84.8

Table 17: Attribution risk, verification risk, and utility of AAST_Φ on Reddit and Blog under different privacy budgets. Results are reported for $K = 1, 8, 16$.

Method	Observed Strengths	Observed Weaknesses	Main Pattern
AAST	Reduces aggregation linkability and preserves main events in qualitative examples. Does not require training generator.	May still reduce exact semantic entailment, especially in cross-genre rewriting. Some private topics and events can remain because utility is preserved.	Best privacy–utility balance under aggregation, with bundle-level selection reducing author signals across multiple released texts.
KiP	Often preserves local meaning and many details of private text.	Shows weaker privacy under aggregation. Qualitative examples suggest that outputs can stay close to private wording and structure, and may contain fragmented words or control artifacts. Requires model adaptation through fine tuning.	Strong meaning preservation, but higher authorship carryover and less stable output form.
JAMDEC	Can generate longer synthetic texts and does not require task specific generator fine tuning.	Longer outputs often contain repetitive or weakly grounded content, with substantial semantic drift in qualitative examples. It also has much higher generation time due to over generation and filtering.	Distance from the private text can help privacy, but often at the cost of meaning preservation and computational efficiency.

Table 18: Qualitative summary of observed strengths and weaknesses. The table summarizes patterns from the quantitative results and qualitative examples.

Method	Bundle	Text
Private	Query ID:1	My **mom** just got diagnosed with **Parkinson's** this week and we are pretty **devastated**. She just turned **60**, and I didn't expect to have to confront **major health problems** so soon. We don't know much about **this disease** and are **scared and devastated**, to say the least. She asked me to help her find **resources** so that she can **understand her disease better**, as she's **not internet-savvy**. Can anyone please help me with this? Where can we read more about this disease? Are there any types of **support** that might be available to her? Anything I need to know about **how to support her**?
	Query ID:2	Have any of you gone through a **grieving** process after **leaving the church**? I have been a **jack Mormon** for many years, and just in the past couple of years have I finally fully **lost my faith** and left completely. I was already living the "worldly" life, so there hasn't been much to gain since I left, but there has been a lot to lose. I am **terrified of death** now, I feel much more **pessimistic** about the future and cynical about life, and sometimes I miss that feeling of being part of a community. To make matters worse, my **mom** was just diagnosed with a **degenerative disease** that will probably **kill her**. I find myself actually wishing that I had some **belief system** and community support to sustain me through the **shock and grief** I feel from her diagnosis, but I don't know that I can ever believe in any religion again. I am **grieving for my mom**, and grieving for all I feel that I lost when I lost my faith. I feel like I will be **crushed under the weight** of all of this grief.
	Target ID:1	I'm a **student taking evening classes** and have also been **working full time** to pay the bills. My job is getting steadily more **stressful and demanding**, and I'm struggling to **mentally cope** and to perform my duties. But I took some **tuition assistance from my employer** to pay for school, and their stipulation is that I have to **stay at this job** until at least Jan of 2021 or I have to **pay back all the money** (which I can't do). I am losing my mind because of how much I **hate this job**, and I need some advice on how to **keep my sanity**. What keeps you sane when you're **stuck at a job you hate but can't afford to leave**?
	Target ID:2	I just need to **vent**. I have a friend who has to **come home early from her mission** due to **mental health issues**. I just had dinner with her the other day and was saddened to see how much she's struggling with **feelings of failure** from being home, compounded by the incredibly **insensitive comments** some members have made. It was so hard for me to keep my mouth shut while I was with her, because I was just boiling inside with **rage at the church** for making young folks feel this way. I had to go home early from my mission also, and although I returned eventually I never fully got over the guilt until I left the church. Missions are **emotionally** (and sometimes physically) **traumatic** for many people, and the church does NOT take care of those kids the way it should protect its **UNPAID VOLUNTEERS**. I could go on and on about everything that is fucked up about the way **missions are run**.
AAST_{M}	Query ID:1	The news that my **mother** was recently diagnosed with **Parkinson's** has left our family **deeply shaken**. She was looking forward to enjoying her **retirement**, and this unexpected diagnosis has been incredibly difficult to process. The reality of facing such a **serious health challenge** so soon is overwhelming. We're still learning about the disease and feel a mix of **fear and uncertainty** about what lies ahead. My mom, who **isn't comfortable navigating the internet**, has asked for my help in finding **reliable information** so she can better understand her condition. If anyone has recommendations for **trustworthy resources**, I'd be so grateful for your guidance. Where should we turn for **accurate and helpful information**?
	Query ID:2	The weight of my **grief** is twofold: I mourn the absence of my **mother**, and I mourn the dissolution of the **beliefs** that once shaped my world. For years, I identified as a **jack Mormon**, but only recently did I fully sever my ties to the **faith**. The transition hasn't been a revelation or a release; it's been a quiet, aching **loss**. I was already living outside the church expectations, so the departure didn't bring new freedoms, only a deepening sense of **emptiness**. The absence of **faith** hasn't lifted me; it's left me **adrift**.
	Target ID:1	This job is driving me absolutely crazy, and I'm desperate for ways to **maintain my mental well-being**. How do you manage to stay grounded when **trapped in a job you despise but can't walk away from**? The situation is even more frustrating because I accepted **tuition assistance from my employer**, which comes with a catch. I'm obligated to **remain here** until I've fully **repaid the money**.
	Target ID:2	I feel the need to **express my frustration**. A close friend of mine is **returning home prematurely from her mission** because of **mental health challenges**. We recently shared a meal together, and it broke my heart to witness how deeply she's grappling with a **sense of inadequacy** over her early return. The situation is made even worse by the **thoughtless remarks** she's had to endure from others, which only pile onto her distress.
AAST_{Φ}	Query ID:1	Just last week, my **mom** found out she has **Parkinson's disease**. Dad finds technology a real maze, so we're **in the dark** about what this all means. I need to know right away **how to help mom** figure out her **next steps**. Feeling **in the dark and scared** is tough. Does anyone know of **specialists or services** that can help her see what she can do? Plus, any **advice** to make this journey smoother would be amazing. Thanks a ton for any **help** you can provide.
	Query ID:2	For years, I considered myself a **jack Mormon**, but recently, my **faith has faded** significantly, prompting my final decision to **leave**. This change has instilled in me an overwhelming **fear of death**, casting a shadow over my view of the **future**, filled with life's **cynicism**. Sometimes, I miss the sense of **community and connection** I once cherished. The thought of embracing any **religion** again appears unlikely.
	Target ID:1	In the whirlwind of daily life, juggling a **full-time education with evening classes** and a **challenging job** is an immense undertaking. The weight of my **workplace duties** has been steadily climbing, causing a **mental tug-of-war** as I strive to meet my commitments. In times of such difficulty, how does one stay rooted and **preserve their mental health** in a **job they find utterly detestable but from which there is no escape**? How do you navigate these turbulent waters?
	Target ID:2	It's quite disheartening that my friend had to **cut her mission short** due to **mental health issues**. Witnessing her struggle with **guilt** over something she wasn't responsible for is distressing. The way **missions are currently managed** is seriously lacking and warrants significant criticism.

Table 19: Qualitative examples on Reddit at K=2. This part shows the private bundle and the AAST outputs. Bold text marks key semantic elements in the private texts and corresponding preserved content in the synthetic texts.

Method	Bundle	Text
KiP	Query ID:1	My **mother** was just diagnosed with p y lor i this week, and we are pretty **devastated**. She just turned **60**, and I didn't expect to have to confront **major health problems** so soon. We don't know much about **this disease** and are, to say the least, **scared and devastated**. She asked me to help her find **resources** so that she might better understand her disease, as she is **not web-savvy**. Can anyone please help me with this?. Where can we read more about this disease?. A re there any kinds of **support** that might be available to her?. Anything I need to know about **how to support her**?
	Query ID:2	H a ve any of you gone through a **griev ing process** after **leaving the church**?. I have been a **jack-mor mon** for many years, and only in the last couple of years have I completely **lost my faith** and left completely. I was already living a life of the world ly life. So there has n't been much to gain since I left, but there has been a lot to lose. I am now more **f right ened of death**; I feel much more **pe s s im istic about the future** and **c y n ical about life**; and sometimes I miss that feeling of being part of a community. To make matters worse, my **mother** was just diagnosed with a **de generative disease** that will probably **kill her**. And really, I find myself wanting that I had some **belief system** and community support to sustain me through the **shock and grief** I feel from her diagnosis, but I don't know that I can ever again believe in any **religion**. I am **griev ing for my mother**, and griev ing for all that I feel that I have lost when I have **lost my faith**. I feel as if I shall be **crushed under the weight** of all these **grief s**
	Target ID:1	I am a **student taking evening classes** and have been **working full time** to pay the bills. lexic al = 60. My job is steadi ly more **difficult and more demanding**; and I am struggling to **cope mentally** and perform my duties. But I took some **tuition assistance from my employer** to pay for school, and their s tipul ation is that I have to **stay at this job** until at least Jan of 20 21, or I must **pay back all the money**, which I cannot do. I am losing my mind because of how much I **hate this job**. I need some advice on how to **keep my s an ity**. What keeps you s an e when you are **stuck at a job you hate but can't afford to leave**?
	Target ID:2	I just need to **vent**. lexic al = 60. And order = 60. I have a friend who has to **come home early from her mission** because of **mental health problems**. I just had dinner with her the other day and was sad d ened to see how much she was struggling with **feelings of failure** from being at home, compound e d by the incredibly **in sensitive comments** some members made. I t was so hard for me to keep my mouth shut while I was with her, because I was boiling inside with **rage at the church** for making young people feel this way. ...(omitted)
JAMDEC	Query ID:1	We are trying to figure out what we are going to do with the money, but we don't know if we will be able to **take care of her** for the rest of our lives or if she will need to be in a nursing home for the remainder of her life, so we are just trying to get as much money as we can to **help her out** as much as possible while we are still able to do so! Thank you so much for taking the time to read our story and if you are able to **help us out**, we would love to hear from you so we can thank you for your kindness! ! ...(omitted)
	Query ID:2	If so, what was it like for you, and what advice would you give to someone who is going through the same thing right now? Please share your thoughts and experiences in the comments section below, and if you'd like to learn more about this topic, check out my book, Becoming a Supporter (affiliate link). If so, what was it like for you, and what advice would you give to someone who is going through the same thing right now? I was raised in the **church**, went to seminary, and served a mission, but after a few years I realized that the **church** was full of lies and that I had been lied to for so long, that I just couldn't take it anymore. ...(omitted)
	Target ID:1	I'm not sure what I'm going to do in the future, but I don't know what I want to do with my life, or if I even want to have a future in the first place, so I'm trying to figure out what to do and how to do it while I still have the ability to do so, and while I'm still able to **pay my bills** and keep up with my obligations to my family, my friends, and the people who care about me, and who have supported me in the past and will continue to support me for as long as I'm able to continue to live and work in this country I don't want to do that, so I'm looking for a job that will allow me to work from home, ...(omitted)
	Target ID:2	I don't know what I'm going to do with my life, but I know I'm not going to be able to do what I want to do for the rest of my life if I don't do something about it now." That's what I've been trying to say to myself for the past few years, but it's hard to say it out loud when you're in a relationship with someone who doesn't seem to understand what you're going through, or if they do, they don't understand it as well as you do, so you end up saying it to yourself in a way that makes you feel like She has been in and out of the hospital for the past year and a half due to her **mental illness**, and I have been trying to help her get the help she needs, but she has been unable to do so due to the fact that she is a member of the **Church** of Jesus Christ of Latter Day Saints. ...(omitted)

Table 20: Qualitative examples on Reddit at K=2, continued. This part shows KiP and JAMDEC outputs for the same private bundle in Table 19. Bold text marks retained content related to the private text.

Subreddits		
frugal	povertyfinance	poverty
poor	assistance	homeless
almosthomeless	tenanthelp	section8publichousing
unemployment	foodstamps	povertykitchen
debt	bankruptcy	borrow
studentloans	medicaid	
Filtering Keywords		
low income	financial burden	poverty
poor	unemployment	underemployment
wage stagnation	low wage	minimum wage
underpaid	low pay	financial hardship
money struggles	financially strained	financial stress
affordability issues	living in poverty	living below the poverty line
working poor	underprivileged	financially disadvantaged
impoverished	economic inequality	trapped in a cycle of poverty
living in low-income housing	experiencing financial instability	living with no savings
completely broke	barely making ends meet	struggling to make ends meet
paycheck to paycheck	living paycheck to paycheck	barely scraping by
barely surviving	cutting back on basic necessities	can't afford
cant afford	cannot afford	can't pay
cant pay	cannot pay	can't pay rent
cant pay rent	cannot pay rent	behind on rent
late on rent	rent overdue	rent arrears
eviction	eviction notice	getting evicted
facing eviction	homeless	almost homeless
about to be homeless	living in my car	sleeping in my car
couch surfing	shutoff notice	utilities shut off
power shut off	water shut off	gas shut off
food insecurity	food bank	food pantry
food stamps	snap	ebt
lost my job	laid off	hours cut
reduced hours	collections	in collections
debt collector	payday loan	title loan
bankrupt	bankruptcy	chapter 7
chapter 13	medical debt	no health insurance
section 8	housing voucher	public housing
medicaid	broke	behind on bills

Table 21: Subreddits and filtering keywords used to construct the Reddit dataset.