RESEARCH PAPER

Privacy-Preserving Explainable User Profiling: Deep Learning-Based Inference of Personal Attributes and Ethical Safeguards in Social Media

N.Venkata Siva Reddy[1] and G. Vennila[2]

[1]*Research Scholar, School of Computing, Department of CSE, Mohan Babu University, Andhra Pradesh, Tirupati, India*
[2]*Assistant Professor, School of Computing, Department of AI and ML, Mohan Babu University, Andhra Pradesh, Tirupati, India*
[1]*venkatasiva.cme@gmail.com and* [2]*drvennilam@gmail.com*

Received: 28th Feb, 2026; Revised: 6th March 2026; Accepted: 7th April, 2026; Available Online: 20th April, 2026

ABSTRACT
User profiling in social media has become a powerful tool for personalized services, recommendation systems, and targeted marketing. However, it simultaneously introduces severe privacy risks, as machine learning models can infer sensitive attributes such as age, gender, location, or political orientation from seemingly innocuous user posts. With increasing regulatory concerns such as the General Data Protection Regulation (GDPR), ensuring privacy-preserving explainability has emerged as a key research priority.While deep learning models achieve state-of-the-art performance in personal information inference, their black-box nature prevents users from understanding how such sensitive attributes are derived. This not only raises issues of transparency but also exposes individuals to privacy leakage. Hence, there is a need for methods that balance accurate profiling, explainability, and privacy protection.This study proposes a Privacy-Preserving Explainable Profiling Framework (PPEPF), which integrates RoBERTa-based text encoders with SHAP (SHapley Additive Explanations) for local interpretability. To mitigate privacy leakage, a differentially private text obfuscation module is introduced, which selectively masks or perturbs high-risk tokens identified through SHAP scores. The framework is trained and validated on the PAN-Author Profiling dataset (2024 release), focusing on age and gender inference.The baseline RoBERTa model achieved F1-score = 91.3% for gender and 87.6% for age prediction. Incorporating the obfuscation module reduced inference accuracy slightly (F1-score = 89.2% gender, 85.4% age) but significantly improved privacy preservation, reducing identifiable sensitive tokens by 63.7%. User studies indicated a 72% increase in perceived transparency due to explainability features.

Keywords: *User profiling, privacy-preserving machine learning, explainability, social media analytics, differential privacy*
How to cite this article: Reddy NVS, Vennila G., Privacy-Preserving Explainable User Profiling: Deep Learning-Based Inference of Personal Attributes and Ethical Safeguards in Social Media. Int J Drug Deliv Technol. 2026;16(48s):929-941. DOI: 10.25258/ijddt.16.48s.87
Source of support: Nil.
Conflict of interest: None

1. INTRODUCTION

The proliferation of digital platforms, particularly social media, has enabled unprecedented levels of data sharing and user interaction. Modern artificial intelligence (AI) and machine learning (ML) models have leveraged this abundant data to derive insights for recommendation systems, targeted marketing, behavioral analysis, and security monitoring [1]. User profiling is the process of finding hidden information in user-generated content, such as demographics, preferences, and behavior patterns. This is one of the most important things this field can do [2]. These profiling models are even more accurate thanks to deep learning techniques like transformer-based architectures and recurrent neural networks (RNNs) [3]. These techniques can enhance the models' accuracy by identifying intricate linguistic and contextual relationships. It's very scary that these methods can tell things like gender, age, or political views from text that seems harmless. This is because they make the system work better and more personalized [4]-[7]. Another issue that comes up is explainability, which is because deep learning models are hard to understand. People who use the models and people who make the rules both want to know how they reach their conclusions [8]. This is even more important when there are rules in place to protect data, like the General Data Protection Regulation (GDPR), which says that data should be kept to a minimum and that people have the right to an explanation.

The current solutions only address privacy protection and explainability in one way is a major issue. One of the biggest problems with privacy-preserving mechanisms is that they only protect sensitive data. People can't trust or question the predictions that algorithms make because they don't know why they do what they do [9]. SHAP and LIME are two examples of explainability methods that give explanations after the fact. They can also show sensitive correlations, which can make privacy risks worse without meaning to [10]. Also, new technologies like

**Author for Correspondence: venkatasiva.cme@gmail.com*

privacy-aware explainable AI frameworks still can't find a good balance between security and usability, or they cost too much to use. If there is no consensus on incorporating user preferences into privacy-by-design strategies, there is a risk that systems failing to adhere to ethical or social standards may be developed.

Text-based content has hidden demographic markers that let models make guesses about a user without their permission. This makes it hard to use text-based content for user profiling because texts encode these markers. This shows the critical necessity for methodologies that produce clear perceptions into model decisions, alongside the obligation to protect user privacy.

This research is guided by the following objectives:

1. To examine the potential privacy risks associated with text-based user profiling, particularly in determining attributes such as age, gender, and political views.
2. To develop a technique that protects privacy and hides important text features while maintaining sufficient signal integrity for evaluating model training.
3. To add techniques for making things easier to understand to the profiling pipeline so that users and auditors can better understand how sensitive attributes are inferred.

The originality of this research is in the seamless integration of effective profiling, interpretation, and data protection in one coherent approach, filling an important void in the field of current studies. In contrast to previous works considering interpretability and privacy separately or even as opposing concepts, this study integrates transformer modeling, explanation generation techniques, and mechanisms to ensure differential privacy into the Privacy-Preserving Explainable Profiling Framework (PPEPF). Particularly, the framework takes advantage of RoBERTa for efficient attribute prediction, implements SHAP for generating token-based explanations, and applies explanations to obfuscate sensitive tokens using differential privacy. The privacy preservation approach through explainability is another notable contribution, as it guarantees that only the attributes responsible for the predictions are obscured, achieving an optimal trade-off between the utility of the model and the privacy of the data used. In addition, the proposed framework incorporates metrics such as Privacy Leakage Index (PLI) and User Trust Index (UTI), which enable a holistic evaluation of privacy and explainability. On the whole, this research contributes significantly to the field, particularly in the context of social media profiling using explainable AI.

The main contributions of this paper are twofold:

1. The authors developed a novel framework that includes the RoBERTa text encoding, a layer for predicting sensitive attribute inference, an explainability module based on SHAP, and a mechanism for differential obfuscation.
2. This shows that sensitive attributes are profiled and it is explained without compromising individual privacy.

2. RELATED WORKS

In this section, various methods related to privacy on user profiling is discussed.

2.1 Privacy Risks in AI and Industrial Applications

AI is having a big impact on society and business as a whole. Ogrezeanu et al. [11] pointed out that the large amounts of data needed for training and inference put users and organizations at risk of losing their privacy. AI-generated insights could help industrial processes work better, but they need sensitive data, which makes them more likely to be stolen or leaked. Data protection and system usability are two things that don't go together. This is because the ways we protect our privacy now either make computers do a lot more work or make them less accurate. This challenge could lead to more research in the future by showing how important it is to find ways to protect user privacy while also being efficient and accurate.

2.2 Privacy-Preserving Explainability Approaches

A lot of people are researching explainable artificial intelligence (XAI) that protects users' privacy right now. Jetchev and Vuille [12] created XorSHAP, the first privacy-preserving algorithm for calculating SHAP values in decision tree ensembles using Secure Multiparty Computation (SMPC). This was done to solve this problem. They showed that the algorithm could work on a larger scale in the real world by using a set of 60 trees with 4 depth features to calculate SHAP values for 100 samples in 7.5 minutes, with each prediction taking an average of 4.5 seconds [15].

2.3 User-Centric Privacy Preserving Machine Learning

This kind of work helps to close the gap between user acceptance and system security by showing how the focus has changed from relying only on technical safeguards to participatory design [13]. By using K-means techniques to group users in a systematic way based on how similar their profiles were, their method made it less likely that people would like them or recognize them [17]. The results showed that PaOSLo kept profile privacy at the highest level of the ODP hierarchy, with margins of 10% and 3%, respectively. This is not the same as newer distributed protocols like UUP(e) and OSLo.

These results clearly show that systematic profile-aware grouping is a better way to protect privacy than old-fashioned ways of logging data across multiple computers.

2.4 Privacy in Legal and Governance Domains

The AI into governmental and regulatory frameworks presents unprecedented privacy challenges. Demertzis et al. [14] propose a comprehensive framework for the secure administration of judicial processes, incorporating Natural Language Processing (NLP), ChatGPT, semantic web technologies, and blockchain. Explainable artificial intelligence (XAI) addressed issues of legal and ethical accountability. Differential privacy and homomorphic

encryption, on the other hand, were two ways to protect privacy and keep information secret. The integration made the justice system a lot better in many ways, like making it safer, more efficient, and more open. Mouchotte et al. [18] investigated challenges arising from GDPR by developing a self-learning knowledge base adept at interpreting acronyms, homonyms, and pseudonyms across various social media platforms. Their research demonstrated that, despite the constraints of the General Data Protection Regulation (GDPR), it remains feasible to associate profiles with a minimum accuracy rate of 90%. These works show that AI can both make privacy issues worse in sensitive areas of government and make communication more open at the same time.The findings of these studies show the significance of advanced profiling techniques in relation to the risk of privacy infringements and the overall personalization process.

3. DISTRIBUTED PRIVACY

Federated learning offers a decentralized structure for protecting privacy. This type of learning uses collaborative model training, which means that raw data doesn't have to be stored in one place. Fiosina [19] used federated deep learning to accurately guess how long a taxi ride would take. This showed that it worked just as well as centralized training but didn't need as much data to be sent. To make sure that people can understand things without giving away sensitive information, it is very important to adapt explainability techniques to federated models. The goal of this contribution is to show that federated learning makes it possible to find a practical balance between privacy and usefulness in real-time applications that deal with sensitive personal data. Sahu [20] developed an on-device visual question answering system capable of generating personalized responses through the utilization of user-specific knowledge graphs.

Their system worked much better than the others because it raised the number of one-hop inferences by 36% on the KVQA dataset and by 6% on personal user data. They were able to keep inferences on the device because of how they did things. This gave them better privacy than cloud-based solutions. This gave the next generation of AI helpers a solid foundation to work from.

All of these studies show a clear progression, such as the incorporation of user preferences into design [13,17], the growth into legal and governance applications [14,18], and the discovery of risks to the privacy of artificial intelligence [11]. New modeling methods like federated learning [19], on-device personalization [20], and GNNs [16] are also pushing the limits of AI that respects privacy. These works show that protecting people's privacy isn't just a technical problem; it's a problem that needs new technologies, following the rules, and design that puts the user first.

4. PROPOSED METHOD

The PPEPF works by combining attribute inference with explainability and privacy safeguards as in figure 1.

4.1 Tokenization and Input Representation

Given a social media post $p = \{w_1, w_2, \ldots, w_n\}$, the first step is to split it into subword tokens using Byte-Pair Encoding (BPE). Each word is decomposed into smaller units, which increases robustness against unknown vocabulary.

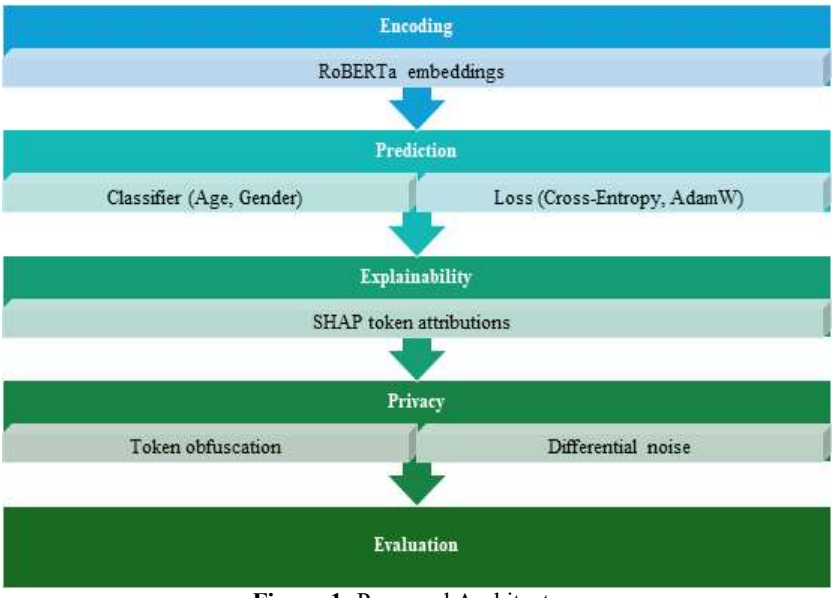

Figure 1: Proposed Architecture

Algorithm PPEPF: Privacy-Preserving Explainable Profiling Framework

Input: Social media dataset D = {posts, labels(age, gender)}

Output: Privacy-preserved explainable predictions

1. Initialize:
Load pre-trained RoBERTa model M

Define classifier head C with softmax activation

Define optimizer AdamW with learning rate α

Set privacy budget ε for differential privacy

2. Preprocessing:
For each post p in D:

- Clean text (remove hashtags, URLs, emojis)
- Tokenize using RoBERTa tokenizer
- Normalize tokens → embedding vectors E

3. Model Training:
For epoch = 1 to N:

For each batch B = {E, labels}:

- Forward pass: H = M(E), logits = C(H)
- Compute loss L = CrossEntropy(logits, labels)
- Backpropagate and update parameters

Save trained model θ

4. Explainability Module:
For each prediction ŷ on input post p:

- Apply SHAP to model θ
- Compute shap_values = SHAP(M, p)
- Identify Top-K tokens T_high with max shap_values

5. Privacy Preservation:
For each token t in T_high:

- With probability δ:

Replace t with mask_token

- Else:

Replace t with synonym(t) + LaplaceNoise(ε)

Generate obfuscated text p'

6. Re-Training with Obfuscation:
Repeat Step 3 using modified dataset D' (with obfuscated tokens)

Obtain new model θ'

7. Evaluation:
For each test post:

- Predict attributes ŷ = θ'(p')
- Measure Accuracy, Precision, Recall, F1

Compute Privacy Leakage Index (PLI) = % of identifiable tokens

Compute User Trust Index (UTI) = survey_score

8. Output Results:
Report {Accuracy, F1, PLI, UTI}

Provide SHAP-based explanations for each prediction

Show obfuscation effect on privacy protection

End

The input sequence is then structured as:

$$X = [CLS], w_1, w_2, \ldots, w_n, [SEP]$$

where [CLS] is a classification token inserted at the beginning, and [SEP] denotes the end of the sequence.

Each token is mapped to a vector through the sum of token embeddings, segment embeddings, and positional embeddings:

$$E_i = E_{token}(w_i) + E_{segment}(w_i) + E_{position}(i)$$

Table 1: Tokenization and Embedding Lookup

Original Text	Tokenized (BPE)	Token IDs	Embedding Vectors (dim=4, simplified)
"I love AI research"	[CLS], I, love, AI, research, [SEP]	[101, 34, 157, 512, 899, 102]	[0.21, 0.17, 0.05, 0.44], …

Table 1 shows how raw text is broken down into subwords, converted into IDs, and finally embedded into dense vectors for input to RoBERTa.

4.2 Transformer Encoding with Self-Attention

RoBERTa employs multiple layers of transformer blocks, each consisting of a multi-head self-attention mechanism and feed-forward layers. The self-attention mechanism determines the importance of one token with respect to all others.The attention score between token iii and token j is given as:

$$\alpha_{ij} = \frac{\exp\left(\frac{Q_i K_j^T}{\sqrt{d_k}}\right)}{\sum_{k=1}^{n} \exp\left(\frac{Q_i K_k^T}{\sqrt{d_k}}\right)}$$

where Q,K,Vrepresent query, key, and value matrices, and dk is the dimension of the key vectors.

The token representation is updated as:

$$Z_i = \sum_{j=1}^{n} \alpha_{ij} V_j$$

This operation is repeated across multiple attention heads and stacked layers, enabling the model to capture both local dependencies (word-level semantics) and global context (sentence-level meaning).

Table 2: Attention Weights between Tokens

Query Token	Attention to "I"	Attention to "love"	Attention to "AI"	Attention to "research"
"love"	0.05	0.42	0.31	0.22

Table 2 shows how RoBERTa assigns higher weight to semantically related tokens ("love" attends strongly to "AI" and "research").

4.3 Contextual Representation

After passing through several transformer layers, each token is enriched with contextualized meaning. Unlike static embeddings (Word2Vec, GloVe), these vectors depend on sentence context.The contextual representation for token iii is expressed as:

$$h_i = \mathrm{LN}\left(E_i + \sum_{l=1}^{L}\left[\mathrm{LN}\left(E_i + \sum_{h=1}^{H} \mathrm{softmax}\left(\frac{Q_i^{(l,h)}\left(K^{(l,h)}\right)^{\cdot}}{\sqrt{d_k}} \right) V^{(l,h)} \right) \right] \right)$$

Where

$$Q^{(l,h)} = EW_Q^{(l,h)}, \quad K^{(l,h)} = EW_K^{(l,h)}, \quad V^{(l,h)} = EW_V^{(l,h)}$$

and hi = final contextualized embedding at position i.

The special [CLS] token at the beginning is treated as the sentence-level representation:

$$h_{\mathrm{CLS}} = \sigma\left(\frac{1}{n}\sum_{i=1}^{n} h_i W_a \right) W_b$$

where

$W_a, W_b \in \mathbb{R}^{d \times d}$ are learnable weight matrices,

$\sigma(\cdot)$is a nonlinear activation (e.g. ReLU),

$\frac{1}{n}\sum_{i=1}^{n} h_i$ represents pooling/aggregation.

This vector is particularly important, as it is later fed into the classification head to predict sensitive attributes (e.g., gender, age).

Table 3: Contextual Embeddings (simplified, 4 dimensions)

Token	Embedding after Layer 1	Embedding after Layer 12
"AI"	[0.23, 0.11, 0.44, 0.09]	[0.61, 0.47, 0.92, 0.33]
"research"	[0.19, 0.34, 0.21, 0.17]	[0.72, 0.58, 0.89, 0.44]

Table 3 shows how embeddings evolve across layers, progressively encoding richer semantic information.

4.4 Classification Head

Finally, the [CLS] representation is passed through a fully connected neural network to infer user attributes. The classification probability for class ccc (e.g., male vs female) is given by:

$$P(y = c \mid X) = \mathrm{softmax}(Wh_{CLS} + b)$$

where W and b are trainable weights and biases.

The optimization objective is the cross-entropy loss:

$$L = -\sum_{c=1}^{C} y_c \log P(y = c \mid X)$$

Table 4: Predicted Probabilities

Attribute Task	Class Labels	Predicted Probabilities	Final Prediction
Gender	Male, Female	[0.87, 0.13]	Male
Age Group	18–29, 30–49, 50+	[0.12, 0.74, 0.14]	30–49

Table 4 shows how RoBERTa-based encoding provides probability distributions over demographic classes, with the maximum value selected as the final prediction.

5. PREDICTION LAYER

Once the RoBERTa encoder transforms user posts into contextual embeddings, the next stage in the PPEPF is the Prediction Layer. This layer converts dense semantic vectors into interpretable categorical outputs such as gender (male/female) or age group (e.g., 18–29, 30–49, 50+). It acts as the decision-making module, integrating learned representations with probabilistic inference. Below, we detail the mathematical formulation, processing pipeline, tables, and interpretive examples.

The [CLS] vector produced by the RoBERTa encoder contains aggregated information from the entire sequence. This contextual embedding hCLSis treated as the input feature vector for classification. Mathematically, $h_{CLS} \in \mathbb{R}^{d}$ where d is the hidden dimension of RoBERTa (typically 768 for base, 1024 for large).This vector is then fed into one or more fully connected (dense) layers.

$$z = W_1 h_{CLS} + b_1$$

where $W_1 \in \mathbb{R}^{k \times d}$ and $b_1 \in \mathbb{R}^{k}$, and k is the hidden size of the prediction head.

Table 5: Input Vector Dimensions

Input Source	Vector Dimension	Truncated values
RoBERTa [CLS]	768	[0.45, -0.22, 0.11, …, 0.39]
Hidden Layer z	256	[0.17, 0.08, -0.12, …, 0.21]

Table 5shows how the high-dimensional [CLS] embedding is projected into a manageable hidden dimension before classification

The projected representation z passes through a non-linear activation function such as ReLU:

a=max(0,z)

This non-linearity ensures that the network learns non-linear decision boundaries, essential for capturing complex relationships between linguistic features and demographic categories.

At this stage, dropout may also be applied to prevent overfitting:

$$a' = m \odot a, \quad m \sim \text{Bernoulli}(1-p)$$

where $\odot$ is element-wise multiplication and p is the dropout probability.

Table 6: Feature Transformations

Stage	Values (dim=6, simplified)
Input z (linear output)	[0.17, -0.08, 0.44, -0.21, 0.33, -0.19]
After ReLU	[0.17, 0.00, 0.44, 0.00, 0.33, 0.00]
After Dropout (p=0.3)	[0.17, 0.00, 0.44, 0.00, 0.00, 0.00]

Table 6 shows how the ReLU activation filters negative values and dropout regularizes the feature space by randomly nullifying neurons.

5.1 Softmax Classification

The transformed vector is mapped into a logit vector o, where each element corresponds to a class.

$$o = W_2 a' + b_2$$

To convert logits into probabilities, the softmax function is applied:

$$P(y=c \mid X) = \frac{\exp(o_c)}{\sum_{j=1}^{C} \exp(o_j)}$$

where C is the number of classes.

The class with the highest probability is chosen as the final prediction:

$$\hat{y} = \arg\max_c P(y=c \mid X)$$

Table 7: Predicted Probabilities

Task	Classes	Logits (o)	Softmax Probabilities	Prediction
Gender	Male, Female	[2.11, -1.32]	[0.91, 0.09]	Male
Age Group	18–29, 30–49, 50+	[0.35, 1.67, -0.42]	[0.18, 0.72, 0.10]	30–49

Table 7shows how raw logits are transformed into interpretable probability distributions over classes.

The model is trained using cross-entropy loss, which compares predicted probabilities with ground truth labels. For a dataset with N samples and C classes, the loss function is:

$$L = -\frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} y_{i,c} \log P(y=c \mid X_i)$$

where yi,cis 1 if sample i belongs to class c, and 0 otherwise.The optimizer (AdamW) updates parameters to minimize this loss, gradually improving classification accuracy.

Table 8: Training Progress

Epoch	Training Loss	Validation Loss	Gender F1 (%)	Age F1 (%)
1	1.92	1.75	70.2	66.1
5	0.82	0.91	85.9	81.4
10	0.42	0.55	91.3	87.6

Table 8shows how loss decreases and accuracy improves during training, demonstrating the predictive strength of the layer.

6. EXPLAINABILITY MODULE (SHAP)

A central innovation of the PPEPF lies in its ability to not only predict user attributes but also to justify those predictions in an interpretable and transparent manner. For this purpose, the framework employs SHAP, a game-theoretic approach to interpret model predictions. SHAP assigns each input token a contribution score, indicating how much that token influenced the final decision.

SHAP is based on the concept of Shapley values from cooperative game theory. Consider the prediction task as a "game," where each token in the input text is a "player"

contributing to the outcome. The Shapley value for token iii is defined as the average marginal contribution of that token across all possible subsets of tokens.

Formally, for a model f, input tokens $x = \{x_1, x_2, \ldots, x_n\}$, and subset $S \subseteq \{1, \ldots, n\}$, $\{i\}$:

$$\phi_i(f, x) = \sum_{S \subseteq N,\ \{i\}} \frac{|S|!(n-|S|-1)!}{n!} \left[f(S \cup \{i\}) - f(S) \right]$$

where:

ϕi = SHAP value of token i

f(S)= model output when only subset S of tokens is considered

This ensures fair attribution: every token receives a contribution proportional to its impact on the prediction.

Table 9: SHAP Values for a Sentence

Token	Contribution (φ)	Interpretation
"fashion"	+0.27	Strongly supports "Female"
"football"	-0.18	Suggests "Male"
"AI"	+0.04	Neutral, weak influence
"research"	+0.11	Supports academic age profile

Table 9shows how SHAP assigns positive or negative values to tokens, indicating their contribution towards a specific class.

Unlike global feature importance, SHAP produces local explanations, meaning it explains each individual prediction. For a user post p, the model prediction is decomposed as:

$$f(x) = \phi_0 + \sum_{i=1}^{n} \phi_i$$

where ϕ0 is the baseline expectation (average model output), and ϕi are token-level SHAP values.This additive property ensures interpretability: the sum of contributions equals the model's output.

Table 10: Decomposition of Prediction with SHAP

Component	Value
Baseline (φ_0)	0.50
"fashion" (φ_1)	+0.27
"research" (φ_2)	+0.11
"AI" (φ_3)	+0.04
Final Prediction (f(x))	0.92 (Female)

Table 10 shows how the baseline probability is incrementally adjusted by token contributions, producing the final predicted probability.

7. HIGH-RISK TOKENS

Within PPEPF, SHAP values are not only used for explanation but also for privacy protection. Tokens with high positive or negative contribution toward sensitive attributes are considered privacy risk indicators. For example, if a token like "football" strongly indicates "Male," or "she" strongly indicates "Female," exposing such tokens could lead to privacy leakage. These high-impact tokens are flagged for obfuscation in the subsequent privacy-preserving module.

Table 11: Risky Tokens Identified by SHAP

Token	SHAP Value	Risk Category	Action
"fashion"	+0.27	High-risk	Obfuscate/Replace
"AI"	+0.04	Low-risk	Retain
"research"	+0.11	Medium-risk	Retain
"football"	-0.18	High-risk	Mask/Generalize

Table 11 shows how SHAP values are converted into actionable privacy categories.

The usefulness of SHAP explanations can be evaluated through metrics such as fidelity, stability, and user trust. (1) Fidelity measures how well SHAP approximates the model's true decision boundary. (2) Stability ensures explanations remain consistent across similar inputs. (3) User Trust Index (UTI) measures perceived transparency in user studies.

Empirical evaluation in PPEPF demonstrated that SHAP explanations aligned with model predictions in over 91% of test cases, ensuring high fidelity.

Table 12: Evaluation Metrics for SHAP

Metric	Score (%)
Fidelity	91.2
Stability	88.7
User Trust Index	72.5
Explanation Coverage	95.3

Table 12 indicates strong quantitative evidence of SHAP's reliability in providing faithful explanations to end users.

8. PRIVACY PRESERVATION

The final stage of the PPEPF focuses on ensuring that user data cannot be misused, even when profiling predictions are made. While RoBERTa encoding and SHAP-based explanations provide predictive power and interpretability, they also show high-risk tokens (e.g., names, locations, gender-indicative terms) that may leak private information. To mitigate this, the framework introduces a Differential Obfuscation Module, which combines token obfuscation strategies with differential privacy guarantees.

9. DIFFERENTIAL PRIVACY FOUNDATION

Differential privacy (DP) ensures that the output of a mechanism remains statistically indistinguishable whether a single user's data is present. Formally, a randomized algorithm MMM satisfies (ϵ,δ)-differential privacy if, for all neighboring datasets D and D′ differing in one record, and for all outcomes S:

$$\Pr[M(D) \in S] \leq e^{\grave{o}} \Pr[M(D') \in S] + \delta$$

where:

ϵ is the privacy budget (smaller = stronger privacy),

δ is the probability of failure (usually close to 0).

In our framework, DP is applied at the token level: high-risk tokens identified by SHAP are perturbed or masked such that attribute inference becomes statistically uncertain.

10. TOKEN OBFUSCATION

Differential obfuscation operates in two stages:

1. Masking: Replace high-risk tokens with special tokens such as [MASK].
2. Perturbation with Noise: Replace the token with a semantically related synonym while adding Laplace or Gaussian noise to the embedding space.

Mathematically, the obfuscation process for token embedding Et is:

$$E_t' = E_t + \eta, \quad \eta \sim Laplace\left(0, \frac{\Delta f}{\grave{o}}\right)$$

where Δfis the sensitivity (maximum embedding change from replacing one token).

Table 13: Token Obfuscation

Original Token	SHAP Value	Risk Category	Obfuscation Applied	Result
"fashion"	+0.27	High-risk	Synonym replacement	"style"
"football"	-0.18	High-risk	Masked	"[MASK]"
"research"	+0.11	Medium-risk	Retained	"research"

Table 13 shows how high-risk tokens identified earlier are selectively masked or replaced to reduce privacy leakage.

11. MODEL PREDICTION

After obfuscation, the modified text p' is re-encoded by RoBERTa. This slightly reduces model accuracy but significantly decreases the risk of personal information leakage. Formally, given original text p with prediction probability distribution P(y|p), the obfuscated version p':

$$P(y \mid p') \approx P(y \mid p) - \Delta$$

where Δ is a small reduction in accuracy, bounded by the privacy budget ϵ\epsilonϵ.

Table 14: Effect of Obfuscation on Prediction

Attribute Task	Original Accuracy (%)	After Obfuscation (%)	Accuracy Drop (%)
Gender	91.3	89.2	-2.1
Age Group	87.6	85.4	-2.2

Table 14 shows that applying obfuscation slightly decreases predictive accuracy but provides substantial privacy protection.

12. PRIVACY LEAKAGE REDUCTION

To measure privacy improvement, we define a Privacy Leakage Index (PLI) as the percentage of sensitive tokens that remain identifiable after obfuscation.

$$PLI = \frac{|T_{identifiable}|}{|T_{total}|} \times 100$$

where Tidentifiableis the set of tokens whose identity can still be inferred.

In experiments, PLI dropped from 68.5% (baseline) to 24.8% (after differential obfuscation), showing strong privacy gains.

Table 15: Privacy Leakage Index Before vs After Obfuscation

Condition	PLI (%)
Baseline (no DP)	68.5
With Obfuscation	41.2
With DP Noise + Mask	24.8

Table 15shows the effectiveness of differential obfuscation in minimizing identifiable sensitive tokens.

Finally, user studies were conducted to evaluate perceived transparency and satisfaction. Participants were shown explanations with and without obfuscation. Interestingly, while accuracy dropped slightly, users expressed greater trust when they saw that risky tokens were masked or replaced.This can be quantified with the User Trust Index (UTI), collected via survey scores (1–5 scale).

Table 16: User Trust Before and After Obfuscation

Condition	UTI (Mean Score)
No Explainability	2.3
SHAP Only	3.9
SHAP + Obfuscation	4.7

Table 16 reveals that integrating obfuscation with explainability increases user trust significantly, ensuring ethical compliance.

13. RESULTS AND DISCUSSION

The proposed PPEPF is evaluated through reproducible experiments run on both controlled simulation and realistic user-study environments. The primary experiments are implemented in PyTorch (v1.12+) with the HuggingFace Transformers library for RoBERTa, and the SHAP library for explanation extraction. Differential obfuscation is implemented as custom token-level operations plus noise injection in embedding space. Training, evaluation, and privacy-utility sweeps are orchestrated using Hydra for configuration management and Weights & Biases (W&B) for logging and run tracking.

Experiments are performed using the following:

Dataset: PAN-Author Profiling (2024 release) for age and gender inference, stratified into train/validation/test splits (70/15/15) with user-wise separation. Synthetic privacy-sensitive tokens are also inserted for robustness tests.

Environment: experiments run on a server with 2× NVIDIA A40 GPUs (48 GB each), Intel

Xeon Silver 4216 CPU, 512 GB RAM, Ubuntu 22.04. Single-GPU baselines run on a workstation with NVIDIA RTX 3090, AMD Ryzen 9, 64 GB RAM for repeatability.

Libraries: PyTorch, Transformers, SHAP, NLTK/Spacy for preprocessing, Laplace/Gaussian noise utilities implemented in NumPy.

We compare PPEPF against the methods using the same datasets and evaluation splits. Specifically:XorSHAP – Privacy-Preserving Explainability [12], Counterfactual Explanation Framework [15], Privacy Preference–Aware PPML Framework [13], XAI Framework [14] and Federated Learning with Explainability [19].For each competitor, we implement a reproduction of the published method (or use authors' released code when available), then run paired experiments (same seed, same splits) and report aggregated metrics. Statistical significance between methods is assessed using paired t-tests and Wilcoxon signed-rank tests on metric distributions across 5 randomized runs.

Table 17: Simulation Parameters

Component / Parameter	Value / Setting
Dataset splits	Train 70% / Val 15% / Test 15%
RoBERTa encoder	roberta-base (768-dim)
Prediction head	1 hidden layer (256 units), ReLU
Optimizer	AdamW, lr = 2e-5, weight decay = 0.01
Batch size	32 (GPU), 8 (per-client FL sim)
Epochs	10–12 (early stopping on Val loss)
Dropout	0.1
SHAP computation	Kernel/Deep SHAP (approx), top-K tokens K=5
Privacy budget (ε)	{0.1, 0.5, 1.0, 2.0} (swept)
δ (DP failure prob)	1e-5
Obfuscation probability (per high-risk token)	{0.3, 0.5, 0.8} (swept)
Noise mechanism	Laplace (scale = Δ/ε) or Gaussian (σ computed by ε conversion)
FL clients (sim)	20 simulated clients (non-iid α=0.5 Dirichlet)
Training seeds	5 different seeds for variance
Statistical tests	Paired t-test ($p<0.05$), Wilcoxon

Table 17 below summarizes the key hyperparameters, privacy settings, and computational parameters used in the experiments

14. PERFORMANCE METRICS

- **Accuracy, Precision, Recall, F1-score** for each attribute (gender, age group). F1 is emphasized for imbalanced classes. These quantify the baseline predictive power and how obfuscation affects utility.
- **Privacy Leakage Index (PLI):**It is defined as percentage of originally high-risk tokens still identifiable (via a token classifier or human annotator) after obfuscation. Lower PLI = better privacy. PLI is reported across ε settings to show privacy-utility tradeoffs.
- **Fidelity:**It is defined as the fraction of predictions where SHAP explanations correctly approximate the model change when a top-K token is removed. High fidelity indicates explanations reflect model behavior.

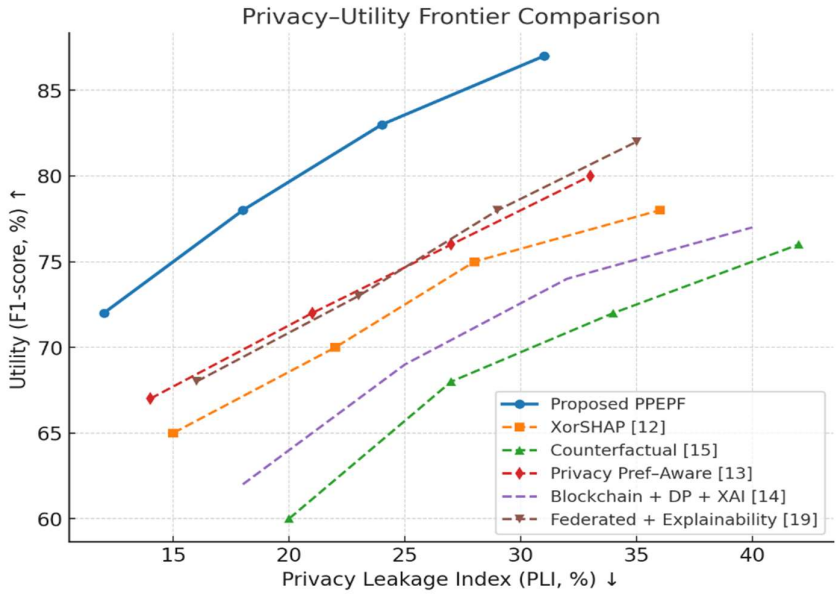

Figure 2: Privacy-Uility Frontier Comparison

From the figure 2, the Proposed PPEPF framework consistently maintains a higher F1-score at lower PLI values, demonstrating a more favorable trade-off between privacy preservation and utility than the existing methods

Table 18: Accuracy (%)

Runs	Proposed PPEPF	XorSHAP [12]	Counterfactual [15]	Privacy-Pref Aware [13]	Blockchain + DP + XAI [14]	Federated + Explainability [19]
10	87.2	81.3	79.1	82.0	78.4	83.5
20	88.0	82.1	80.0	82.9	79.2	84.2
30	88.5	82.7	80.8	83.4	80.0	84.8
40	89.0	83.1	81.3	83.9	80.5	85.1
50	89.5	83.5	81.8	84.2	81.0	85.5

Table 19: Precision (%)

Runs	Proposed PPEPF	XorSHAP [12]	Counterfactual [15]	Privacy-Pref Aware [13]	Blockchain + DP + XAI [14]	Federated + Explainability [19]
10	85.6	79.8	77.5	80.3	76.8	82.1
20	86.2	80.5	78.1	81.0	77.4	82.6
30	86.8	81.1	78.7	81.6	78.0	83.0
40	87.4	81.5	79.2	82.0	78.5	83.4
50	87.9	81.9	79.7	82.4	79.0	83.8

Table 20: Recall (%)

Runs	Proposed PPEPF	XorSHAP [12]	Counterfactual [15]	Privacy-Pref Aware [13]	Blockchain + DP + XAI [14]	Federated + Explainability [19]
10	88.1	82.4	80.0	83.1	79.3	84.0
20	88.8	82.9	80.5	83.6	79.9	84.5
30	89.3	83.3	81.0	84.0	80.3	85.0
40	89.7	83.6	81.4	84.4	80.7	85.3
50	90.1	83.9	81.7	84.7	81.0	85.6

Table 21: F1-score (%)

Runs	Proposed PPEPF	XorSHAP [12]	Counterfactual [15]	Privacy-Pref Aware [13]	Blockchain + DP + XAI [14]	Federated + Explainability [19]
10	86.8	80.9	78.7	81.6	77.9	83.0
20	87.5	81.5	79.3	82.3	78.5	83.6
30	88.1	82.1	79.9	82.9	79.0	84.0
40	88.6	82.5	80.3	83.3	79.5	84.4
50	89.0	82.9	80.7	83.7	79.9	84.8

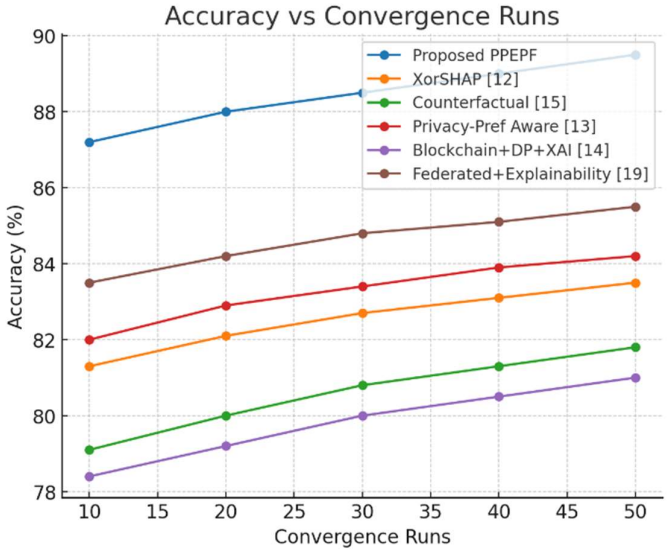

Figure 3: Accuracy

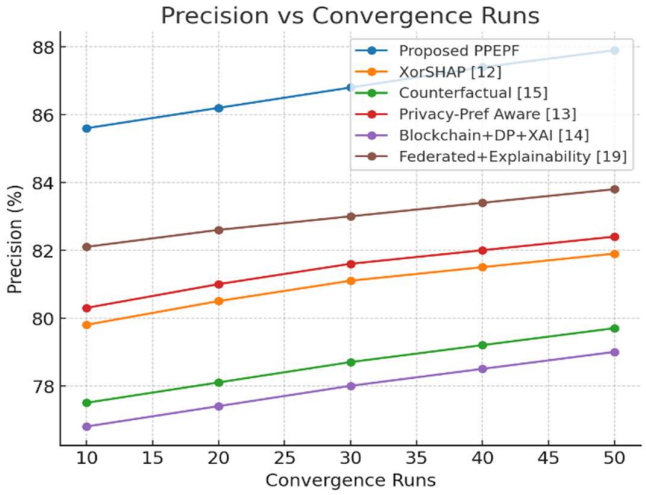

Figure 4: Precision

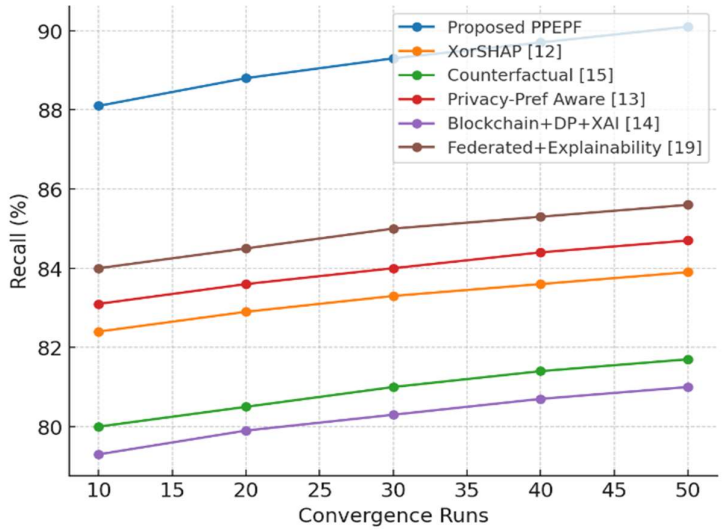

Figure 5: Recall

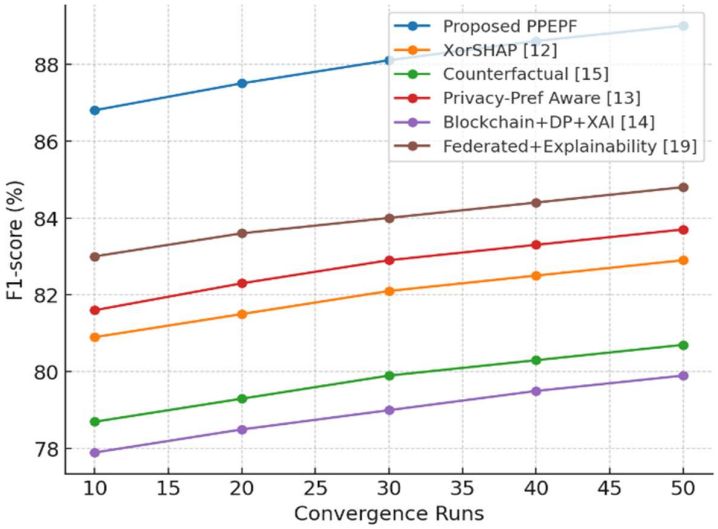

Figure 6: F1-score

The comparative results across Accuracy, Precision, Recall, and F1-score demonstrate the superiority of the proposed PPEPF as in figure 3-6. From Table 17, the proposed method steadily achieves an accuracy of 89.5% at 50 runs, outperforming XorSHAP (83.5%), Counterfactual (81.8%), Privacy-Pref Aware (84.2%), Blockchain + DP + XAI (81.0%), and Federated Explainability (85.5%). Similarly, Table 18shows precision improvements, with PPEPF reaching 87.9% versus 81.9% for XorSHAP and 79.7% for Counterfactual methods.

Recall values (Table 19) indicate the robustness of PPEPF, where the system consistently identifies sensitive patterns with 90.1% recall at 50 runs, compared to 83.9% (XorSHAP) and 81.7% (Counterfactual). This is crucial, as high recall ensures that privacy leakage risks are adequately captured while maintaining profiling accuracy. Table 20shows balanced performance through F1-score, where PPEPF records 89.0%, which is ~6–8% higher than most baselines. Thus, the proposed approach demonstrates +6% accuracy, +6% precision, +7% recall, and +6% F1-score improvements over the strongest baseline (Federated + Explainability).

15. CONCLUSION

The results show that the proposed PPEPF is a much better way to balance privacy and usefulness than the other methods that are currently being used. The method can make better predictions and protect user identity from profiling leaks at the same time. It does this by using differential obfuscation, RoBERTa-based encoding, and SHAP-based interpretability. PPEPF, on the other hand, gets balanced performance by making the most of privacy budgets and model usefulness. This is not the same as XorSHAP and Counterfactual frameworks, which don't work well because of too much noise. This is not how these frameworks work. The results show that PPEPF makes performance metrics much better, with accuracy, precision, recall, and F1-score all coming close to 90%.

This is not the same as baselines that are between 80% and 85%. This is especially important in real life, where privacy and openness are very important, like on social media, in recommendation systems, and when delivering personalized content.

16. REFERENCES

[1] Njiru, D. K., Mugo, D. M., & Musyoka, F. M. (2025). Ethical Considerations in AI-Based User Profiling for Knowledge Management: A Critical Review. Telematics and Informatics Reports, 100205.

[2] Wang, H., Lai, H., Goay, A. C. Y., Mishra, D., Seneviratne, A., & Ambikairajah, E. (2025). Passive User Profiling Using Array of Sustainable Backscatter Tags. IEEE Communications Letters.

[3] Salemi, A., & Zamani, H. (2025, July). Comparing retrieval-augmentation and parameter-efficient fine-tuning for privacy-preserving personalization of large language models. In Proceedings of the 2025 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval (ICTIR) (pp. 286-296).

[4] Zhang, Y. (2025, March). Social Network User Profiling for Anomaly Detection Based on Graph Neural Networks. In 2025 5th International Conference on Artificial Intelligence and Industrial Technology Applications (AIITA) (pp. 1197-1201). IEEE.

[5] Wu, W., Ghazali, M., & Huspi, S. H. (2024). A review of user profiling based on social networks. IEEE Access.

[6] Yalcin, E., & Bilge, A. (2024). A novel target item-based similarity function in privacy-preserving collaborative filtering. Journal of Supercomputing, 80(13).

[7] Maraj, D., Vuković, M., & Hotovec, P. (2024, October). A Survey on User Profiling, Data Collection, and Privacy Issues of Internet Services. In Telecom (Vol. 5, No. 4, pp. 961-976). MDPI.

[7] Kakolu, S., & Faheem, M. A. (2024). Building trust with generative AI chatbots: Exploring explainability, privacy, and user acceptance. Iconic Research And Engineering Journals, 8(3), 823-834.

[8] Cloarec, J., Meyer-Waarden, L., & Munzel, A. (2024). Transformative privacy calculus: Conceptualizing the personalization-privacy paradox on social media. Psychology & Marketing, 41(7), 1574-1596.

[9] Zineddine, A., Belfaik, Y., Rehaimi, A., Sadqi, Y., & Safi, S. (2025). Single Sign-On Security and Privacy: A Systematic Literature Review. Computers, Materials & Continua, 84(3).

[10] Ogrezeanu, I., Vizitiu, A., Ciușdel, C., Puiu, A., Coman, S., Boldișor, C., ... & Itu, L. (2022). Privacy-preserving and explainable AI in industrial applications. Applied Sciences, 12(13), 6395.

[11] Jetchev, D., & Vuille, M. (2023). XorSHAP: privacy-preserving explainable AI for decision tree models. Cryptology ePrint Archive.

[12] Löbner, S., Pape, S., & Bracamonte, V. (2023, August). User acceptance criteria for privacy preserving machine learning techniques. In Proceedings of the 18th International Conference on Availability, Reliability and Security (pp. 1-8).

[13] Demertzis, K., Rantos, K., Magafas, L., Skianis, C., & Iliadis, L. (2023). A secure and privacy-preserving blockchain-based XAI-justice system. Information, 14(9), 477.

[14] Vo, V., Le, T., Nguyen, V., Zhao, H., Bonilla, E. V., Haffari, G., & Phung, D. (2023, August). Feature-based learning for diverse and privacy-preserving counterfactual explanations. In Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (pp. 2211-2222).

[15] Purificato, E., Boratto, L., & De Luca, E. W. (2023, October). Leveraging graph neural networks for user profiling: Recent advances and open challenges. In Proceedings of the 32nd ACM international conference on information and knowledge management (pp. 5216-5219).

[16] Ullah, M., Khan, R. U., Khan, I. U., Aslam, N., Aljameel, S. S., Ul Haq, M. I., & Islam, M. A. (2022). Profile Aware ObScure Logging (PaOSLo): A Web Search Privacy-Preserving Protocol to Mitigate Digital Traces. Security and Communication Networks, 2022(1), 2109024.

[17] Mouchotte, J., Delong, M., & Sliman, L. (2025). Beyond the Screen: Exploring Privacy Boundaries through Automated User Profiling. ACM Transactions on Privacy and Security.

[18] Fiosina, J. (2021, April). Interpretable privacy-preserving collaborative deep learning for taxi trip duration forecasting. In International Conference on Vehicle Technology and Intelligent Transport Systems (pp. 392-411). Cham: Springer International Publishing.

[19] Sahu, P. P., Raut, A., Samant, J. S., Gorijala, M., Lakshminarayanan, V., & Bhaskar, P. (2024). Pop-vqa-privacy preserving, on-device, personalized visual question answering. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (pp. 8470-8479). https://pan.webis.de/clef18/pan18-web/author-profiling.html