LEVERAGING k-ANONYMITY FOR DATA SAFEGUARDING AND ENHANCED CONFIDENTIALITY IN GENERATIVE AI

Padmaja H[a], Dr. P Jayashree[b]

[a]*Department of Computer Technology, Anna University , Chennai,*
[b]*Professor and Head, Department of Computer Technology, Anna University, Chennai,*

Abstract

In Today's scenario, Generative AI has become a part and parcel of our daily life. Their computing power is being used extensively to analyse data from multiple perspectives and for applications requiring unconventional solutions. But these systems train on huge volumes of data collected from public domain , enterprise etc. As AI systems collect more personal data, privacy risks increase, threatening personal rights and security if data is misused. However, privacy must be balanced with utility. We plan to demonstrate how k-anonymity, l-diversity, and t-closeness allow the use of data to train AI models while protecting sensitive attributes. These techniques work by generalizing, perturbing, or restricting data to make individuals less identifiable while retaining overall patterns for Generative AI to effectively get trained and analyse them.

Keywords: k-Anonymity, l-diversity, t-closeness, Generative Adversarial Network

1. Introduction

The advent of generative artificial intelligence (AI) as a consumer product has sparked excitement in virtually every industry and brought a new level of awareness of machine learning's capabilities to the public. Now that Generative AI is no longer confined to research labs and has entered the public square, it has sprouted into dozens of new applications and at least as many products. Generative AI is a class of machine learning algorithms that use neural networks to create text, images, and other content that is substantially different from anything it was trained on and much more complex than any previous machine learning model was capable of [22] . Generative AI could change how we interface with computers forever in the coming days [9] . Generative AI presents a dramatic leap forward in machine learning's capabilities. Advanced machine learning algorithms can now generate natural sounding text and produce compelling imagery quickly and easily with minimal human involvement. Generative AI has profoundly shifted the way ideas, data, and information are created and then shared, a domain that has required human intelligence since, well, the birth of language itself. The success of Generative AI model is highly dependent on being trained on very large data sets (about terabytes to petabytes of data). Such models Foundation models (FM). The term foundation model was coined by researchers to describe ML models trained on a broad spectrum of generalized and unlabeled data and capable of performing a wide variety of general tasks such as understanding language, generating text and images, and conversing in natural language. Models are based on complex neural networks including generative adversarial networks (GANs), transformers, and variational encoders. These Foundation models work by studying a large body of text from all over the internet and from user queries and user uploaded data for analysis, report generation, summarization. The core challenge posed by generative AI right now is that unlike conventional applications, these models have no "delete" button. There's no straightforward mechanism to "unlearn" specific information [8] . The data used by the FM could be employee's sensitive information, a company's confidential reports, insurance details of consumer, due-diligence contract or banking details. These data deserve to be protected against unauthorised access and exploitation. This brings the need to develop advanced privacy preserving techniques which can strike a balance between data privacy and utility.

2. Key Terminology

PII: Personally, identifiable information: These are identifiers like Name which can directly identify an individual. Any representation of information that permits the identity of an individual to whom the information applies to be reasonably inferred by either direct or indirect means.
Quasi identifiers: These are identifiers themselves cannot identify a person directly but can identify indirectly.
k-anonymity[3]: This is a generic privacy protection concept that suggests that the maximum probability of identifying an individual in a specific set must be lower than 1/k.
L-diversity: This ensures that each k-anonymous group contains at least l different values of the sensitive attribute. Therefore, even if an adversary can identify the group of a person, he/she still would not be able to find out the value of that person's sensitive attribute with certainty[12] .
T-closeness: This makes sure that the distribution of sensitive attribute values in a given partition is similar to the distribution of the values in the overall dataset.

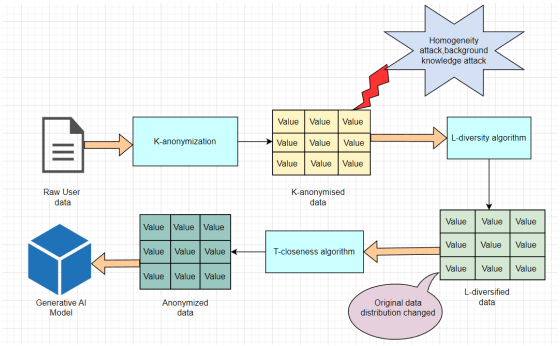

Figure 1: Proposed System model

Generalization: This technique involves replacing (or recoding) a value with a less specific but semantically consistent value.
Suppression: This technique involves not releasing a value at all. It uses asterisks or specific labels to mask some part of the value to mask the value.
Homogeneity : Similar or same values for the attributes in a table or dataset .

3. Proposed Architecture

The Figure 1 depicts the flow of the technique.The figure discusses the Privacy protection technique using k-anonymity , l-diversity and t-closeness. The dataset will be first anonymised with k-anonymisation by choosing the appropriate value for k. Even after k-anonymisation , the data might still be prone to homogeneity attacks . To mitigate against these attacks , we apply l-diversification with optimal value of l. This can perturb the data distribution making it prone to skewness attacks and change the results of analysis which is dependent on the original distribution of data. T-closeness can be applied to match the distribution of sensitive attribute in the cluster to distribution in the original dataset. This anonymised dataset would be fed to a Generative AI model such as GAN (Generative Adversarial Network). The detailed description of the algorithms would be explained in the subsequent sections.

4. Methodology

There are several techniques to implement the above mentioned anonymisation like generalisation, suppression, Median based Mondrain algorithm . In this Journal we have discussed about the Mondrain Algorithm. The Mondrian algorithm is a top-down data anonymization algorithm that uses greedy partitioning to anonymize relational datasets. It recursively partitions a dataset by finding the median of the quasi-identifier with the highest number of unique values. The algorithm continues to partition until the equivalence class size is less than 2k-1. The Mondrian algorithm starts with the least specific value of the attributes in the QID set and specializes as partitions are performed on the data. It is the fastest local recording algorithm, while also preserving good data utility.

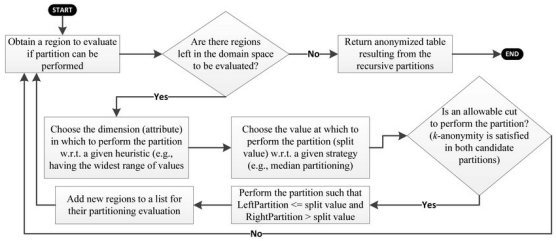

Figure 2: Flow chart of Mondrian Algorithm

It uses median as it is immune to outliers and missing data compared to mean. This is a robust statistic and hence used to partition the dataset. The partitions are then examined to check whether they satisfy the k-anonymity, l-diversity and t-closeness criterion. The flowchart of the Algorithm is depicted in Figure 2
DATASET USED : Adult Census Income Dataset with 48,842 instances and 14 features, is considered for the implementation project purpose. The data records and its tuples relation can be expressed mathematically as follows. Let B(A_1,A_2,..., A_n) be a table with a finite number of tuples. The finite set of attributes of B are {A_1,A_2,..., A_n} Given a table B(A_1,A_2,..., A_n), {A_1,..., A_j} is a subset {A_1,A_2,..., A_n}, and a tuple t belongs to B, We use t[A_1,..., A_j] to denote the sequence of the values, v_1,..., v_j, of A_1,..., A_j in t. We use B[A_1..., A_j] to denote the projection of the record.

Definition of k-Anonymity [4][24] : Let RT(A_1,A_2, ..., A_n) be a table and [QI_{RT}] be the quasi-identifier associated with it. RT is said to satisfy k-anonymity if and only if each sequence of values in RT[QI_{RT}] appears with at least k occurrences in RT[QI_{RT}].

As shown in algorithm , K-anonymity demands that we group individual rows/persons of our dataset into group of at least k rows/persons and replace the quasi-identifier attributes of these rows with aggregate quantities, such that it is no longer possible to read the individual values. This protects people by ensuring that an adversary who knows all values of a person's quasi-identifier attributes can only find out which group a person might belong to but not know if the person is really in the dataset.
Definition of L-diversity : Let T be the initial table to be published. T contains d quasi-identifier (QI) attributes$QA_1, QA_2, ..., QA_d$ and a sensitive attribute (SA). Although each QA_i (1 i d) can be either numerical or categorical. Then a table I satisfies L-diversity when there are at least l "well-represented" or l- distinct values for SAs.

As mentioned in the definition, L-diversity demands that we group individual rows/persons of our dataset into group of at least k rows/persons and such that each group or partition contains l-distinct values of the sensitive data attribute [25] . Upon doing this we replace the quasi-identifier attributes of these rows with aggregate quantities, such that it is no longer possible to locate the specific person's record.

L-diversity algorithm would have perturbed the dataset distribution. Hence any analysis to be performed or dependent on the original distribution of data would get affected. T-closeness resolves this further by reducing variance of attribute distributions compared to the overall distribution through agglomerative clustering and dimensionality reduction. A well-implemented privacy-preserving AI system upholds ethical data usage standards. Establishing privacy in AI is essential for public trust, preventing discriminatory outcomes, and mitigating financial, social, and security harms from emerging technologies.

Definition of t-closeness Principle : An table T is said to have t-closeness if the distance between the distribution of a sensitive attribute in this table and the distribution of the attribute in the whole table is no more than a threshold t.
This makes sure that the distribution of sensitive attribute values in a given partition is similar to the distribution of the values in the overall dataset. T-closeness preserve the semantic significance of the data distribution. Also, this way any inference drawn or analysis made from the distribution of original data does not get deviated due to the anonymization.This way any analysis done based on the original distribution of the data and data utility is preserved and at the same time privacy is also protected.

Algorithm 1 Anonymization with Optimal k-Value

Input :Original Dataset D; quasi-identifier attributes $QI = (q_1, \ldots, q_m)$
Output:Anonymized data frame D_a with optimal k-anonymity

$P_{\text{finished}} \leftarrow \{\}$ ▷ Finished set of partitions
$P_{\text{working}} \leftarrow \{\{1, 2, \ldots, N\}\}$ ▷ Working set of partitions initialized to entire dataset
while P_{working} is not empty **do** ▷ While there are partitions in the working set
    $p \leftarrow \text{pop}(P_{\text{working}})$ ▷ Get one partition from the working set
    Calculate relative spans of all columns in partition p.
    Sort columns by their span (in descending order).
    for each column in partition p **do**
        Split partition along the column using median as the split point.
        if resulting partitions are valid according to k-anonymity criteria **then**
            Add the two new partitions to the working set.
            break ▷ Exit the loop
        else
            Add the original partition to P_{finished}.
        end if
    end for
end while
return P_{finished} ▷ Return the finished set of partitions
▷ Plot rectangular subplots showing partitions and distribution of data
▷ Display anonymized table with quasi and sensitive identifiers for each partition

Algorithm 2 Anonymization with Optimal l-Value

Input :Original Dataset D; quasi-identifier attributes $QI = (q_1, \ldots, q_m)$; Sensitive Attributr SA
Output:Anonymized data frame D_a with optimal l-diversity

$P_{\text{finished}} \leftarrow \{\}$ ▷ Finished set of partitions
$P_{\text{working}} \leftarrow \{\{1, 2, \ldots, N\}\}$ ▷ Working set of partitions initialized to entire dataset
Determine the correct value of l by observing the sensitive data attribute and identifying the number of unique values in the sensitive attribute SA.
while P_{working} is not empty **do** ▷ While there are partitions in the working set
    $p \leftarrow \text{pop}(P_{\text{working}})$ ▷ Get one partition from the working set
    Calculate relative spans of all attributes in partition p.
    Sort attributes by their span (in descending order).
    for each column in partition p **do**
        Split partition along the column using the median of the column values as the split point.
        if resulting partitions are valid according to k-anonymity and l-diversity criteria **then**
            if each partition contains at least l distinct values of sensitive data attribute **then**
                Add the two new partitions to the working set.
                break ▷ Exit the loop
            end if
        else
            Add the original partition to P_{finished}.
        end if
    end for
end while
return P_{finished} ▷ Return the finished set of partitions
▷ Plot rectangular subplots showing partitions and distribution of data
▷ Display l-diversified table with quasi and sensitive identifiers for each partition

EVALUATION METRICS USED Discernibility Metric: Measures how many potential identifying attributes are suppressed or generalized. Higher discernibility indicates better generalization. It is a measure of the decrease in granularity of quasi-identifiers by calculating the difference in number of distinct combinations of Quasi-Identifier attributes before and after anonymization. The discernibility metric (DM) measures how indistinguishable a record is from others by assigning a penalty to each record, equal to the size of the equivalence class in which it belongs.
Formula for metric :
$discernibility = distinct\_combinations\_original - distinct\_combinations\_anonymized$

A higher value is better here, as it means more generalization was applied and there are fewer distinguishing quasi-identifier combinations in the anonymized data.

The Mondrian Algorithm was implemented in Python language with Pandas for data manipulation and matplotlib for graphical visualization of data points and cluster realization.

Algorithm 3 t-Closeness Anonymization Method

Input :Original Dataset D; quasi-identifier attributes $QI = (q_1, \ldots, q_m)$; Sensitive Attributr SA

Output:Anonymized data frame D_a with optimal t-closeness

$P_{\text{finished}} \leftarrow \{\}$ ▷ Set of finished partitions

$P_{\text{working}} \leftarrow \{\{1, 2, \ldots, N\}\}$ ▷ Initial partition of entire dataset

$total\_count \leftarrow$ total number of records in the dataset

$group\_counts \leftarrow$ group data and count occurrences of each value

for each value in $group\_counts$ **do**

    Calculate the proportion p of each value in the sensitive attribute compared to $total\_count$.

    Store proportion p in the dictionary $global\_freqs$ with the value as the key.

end for

while P_{working} is not empty **do**

    $part \leftarrow \text{pop}(P_{\text{working}})$ ▷ Take a partition from working set

    Group data in partition $part$ by the specified column and count occurrences, storing the result in $group\_counts$.

    $d_{\max} \leftarrow 0$ ▷ Initialize maximum difference

    for each $(value, count)$ in $group\_counts$ **do**

        Calculate proportion p of $count$ to $total\_count$.

        Calculate absolute difference d between p and $global\_freqs[value]$.

        Update $d_{\max}$ if d is greater than the current $d_{\max}$.

    end for

    if $d_{\max}$ is within threshold **then**

        Add partition $part$ to P_{finished}.

    else

        Add original partition $part$ to P_{finished}.

    end if

end while

return P_{finished} ▷ Return the finished partitions

▷ Plotting and displaying the data after t-closeness

▷ Display the table after t-closeness with quasi and sensitive identifiers for each partition

5. Results of Anonymization

Following cases have been tested for different set of quasi identifiers to evaluate the discernability metric.

a) Two quasi identifiers both numerical

b) Two quasi identifiers one numerical , one categorical

c) Two quasi identifiers both categorical

d) Three quasi identifiers all numerical

But we have illustrated the results of only Case (a) in detail here . The results shown here are the outputs obtained on applying k-anonymisation , l-diversification and t-closeness .

```
In [1]: runfile('C:/Users/LENOVO/OneDrive/Desktop/CIP project/
implementation/untitled0.py', wdir='C:/Users/LENOVO/OneDrive/Desktop/CIP
project/implementation')
   age         workclass  fnlwgt  ... hours-per-week  native-country income
0   39         State-gov   77516  ...             40   United-States  <=50k
1   50  Self-emp-not-inc   83311  ...             13   United-States  <=50k
2   38           Private  215646  ...             40   United-States  <=50k
3   53           Private  234721  ...             40   United-States  <=50k
4   28           Private  338409  ...             40            Cuba  <=50k

[5 rows x 15 columns]
{'age': 73, 'workclass': 9, 'fnlwgt': 1478115, 'education': 16, 'education-
num': 15, 'marital-status': 7, 'occupation': 15, 'relationship': 6, 'race':
5, 'sex': 2, 'capital-gain': 99999, 'capital-loss': 4356, 'hours-per-week':
98, 'native-country': 42, 'income': 2}
460
[((17.0, 7.0), (18.0, 9.0)), ((18.0, 7.0), (20.0, 9.0)), ((21.0, 10.0),
(22.0, 11.0)), ((25.0, 10.0), (27.0, 11.0)), ((37.0, 9.0), (39.0, 10.0)),
((37.0, 10.0), (38.0, 13.0)), ((41.0, 10.0), (43.0, 13.0)), ((39.0, 13.0),
(41.0, 16.0)), ((46.0, 10.0), (48.0, 13.0)), ((46.0, 13.0), (48.0, 16.0))]
```

Figure 3: View of original dataset and the span for all features

```
          age  education-num income  count
0   17.000000       7.200599  <=50k    334
1   18.227876       7.283186  <=50k    451
2   18.227876       7.283186   >50k      1
3   21.000000      10.000000  <=50k    568
4   21.000000      10.000000   >50k      2
..        ...            ...    ...    ...
771 82.714286      14.428571   >50k      3
772 90.000000      14.000000  <=50k      2
773 90.000000      14.000000   >50k      3
774 89.200000      15.000000  <=50k      3
775 89.200000      15.000000   >50k      2

[776 rows x 4 columns]
     age  education-num income  count
605  17.0       4.000000  <=50k      5
110  17.0       5.000000  <=50k     36
111  17.0       6.000000  <=50k    198
0    17.0       7.200599  <=50k    334
120  17.0       9.000000  <=50k     14
..    ...            ...    ...    ...
726  90.0       9.000000   >50k      4
736  90.0      10.545455  <=50k      9
737  90.0      10.545455   >50k      2
772  90.0      14.000000  <=50k      2
773  90.0      14.000000   >50k      3
```

Figure 4: : Depiction of k-Anonymised dataset view with the cluster median value for the quasi identifiers

```
[776 rows x 4 columns]
sorted
303
Finished 1 partitions...
Finished 101 partitions...
Finished 201 partitions...
Finished 301 partitions...
           age  education-num income  count
0    17.706107       7.248092  <=50k    785
1    17.706107       7.248092   >50k      1
114  18.341463       3.365854  <=50k     40
115  18.341463       3.365854   >50k      1
4    19.320276      10.000000  <=50k   1301
..         ...            ...    ...    ...
589  89.727273      13.000000   >50k      2
578  90.000000      10.545455  <=50k      9
579  90.000000      10.545455   >50k      2
602  90.000000      14.000000  <=50k      2
603  90.000000      14.000000   >50k      3

[606 rows x 4 columns]
```

Figure 5: Depiction of l-diversified dataset view with the cluster median value for the quasi identifiers

```
{'<=50k': 0.7607182343065395, '>50k': 0.23928176569346055}
114
Finished 1 partitions...
Finished 101 partitions...
          age  education-num income  count
12   24.543287      11.476788  <=50k    738
13   24.543287      11.476788   >50k     59
2    25.747108      10.000000  <=50k   5617
3    25.747108      10.000000   >50k    520
0    26.697666       8.124394  <=50k  10248
..         ...            ...    ...    ...
225  84.142857      10.142857   >50k      1
198  84.266667       9.000000  <=50k     55
199  84.266667       9.000000   >50k      5
226  90.000000      10.545455  <=50k      9
227  90.000000      10.545455   >50k      2

[228 rows x 4 columns]

In [4]:
```

Figure 6: Depiction of t-closeness dataset view with the cluster median value for the quasi identifiers and discernability metric after anonymization

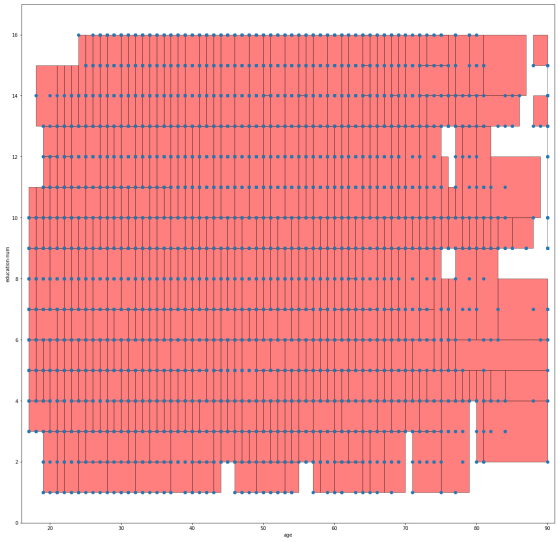

Figure 7: Graph of k-anonymity partition : 2 Quasi Identifier (Numerical)

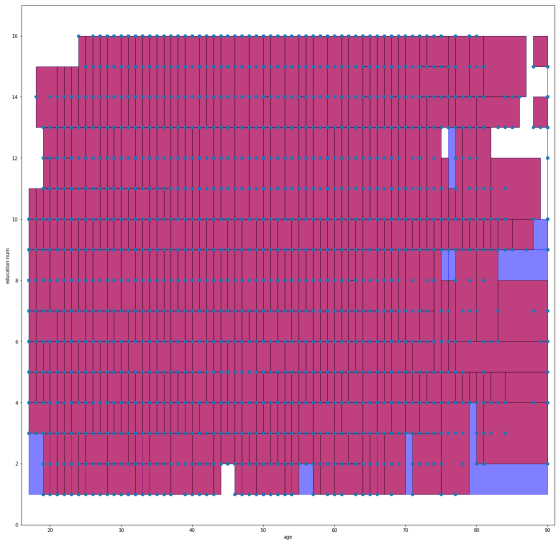

Figure 8: Graph of L-diversity partition : (2 Quasi Identifier (Numerical)

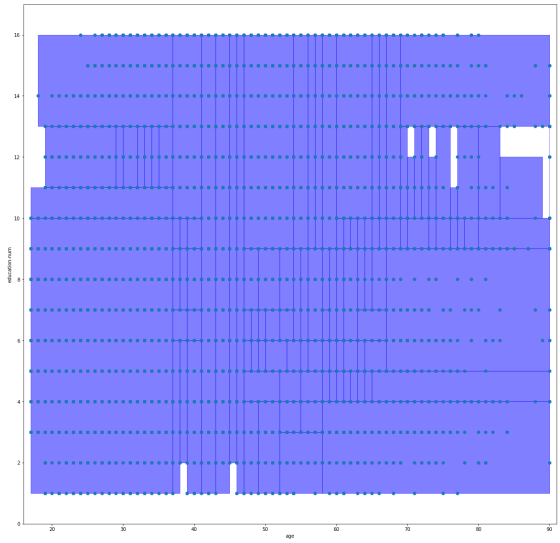

Figure 9: Graph depicting Partition after T closeness : (2 quasi Identifiers- Numerical)

Age	Race	Income	Count
31.38235	Amer-Indian-Eskimo	<=50k	31
31.38235	Amer-Indian-Eskimo	>50k	3
33	Amer-Indian-Eskimo	<=50k	12
33	Amer-Indian-Eskimo	>50k	1
..	..	..	..
51.29412	Amer-Indian-Eskimo	<=50k	15
51.29412	Amer-Indian-Eskimo	>50k	2
53.11111	Amer-Indian-Eskimo	<=50k	8
53.11111	Amer-Indian-Eskimo	>50k	1

Figure 10: Table representing the k-anonymisation for one categorical and one numerical

Age	Race	Income	Count
34	Amer-Indian-Eskimo	<=50k	18
34	Amer-Indian-Eskimo	>50k	3
35.33333	Amer-Indian-Eskimo	<=50k	30
35.33333	Amer-Indian-Eskimo	>50k	3
..	..	..	..
55.39286	Asian-Pac-Islander	<=50k	20
55.39286	Asian-Pac-Islander	>50k	8
57.27586	Asian-Pac-Islander	<=50k	16
57.27586	Asian-Pac-Islander	>50k	13

Figure 11: Table representing the l-diversity for one categorical and one numerical

Occupation	Nationality	Income	Count
Handlers-cleaners	Haiti,Ecuador,Columbia	<=50k	9
Handlers-cleaners	Haiti,Ecuador,Columbia	>50k	1
Handlers-cleaners	Japan,South,Outlying-US(Guam-USVI-etc)	<=50k	8
Exec-managerial	Vietnam,Scotland	<=50k	7
..	..	..	..
Farming-fishing	Poland,Cuba,Mexico	>50k	2
Farming-fishing	Germany,Puerto-Rico	<=50k	11
Tech-support	Mexico,Puerto-Rico	<=50k	6
Tech-support	Mexico,Puerto-Rico	>50k	3

Figure 12: Table representing the k-anonymisation for two categorical attributes

Occupation	Nationality	Income	Count
Exec-managerial	Columbia,Taiwan	<=50k	11
Exec-managerial	Columbia,Taiwan	>50k	8
Exec-managerial	Ecuador,Laos	<=50k	3
Exec-managerial	Ecuador,Laos	>50k	2
..	..	..	..
Protective-serv	Philippines	<=50k	3
Protective-serv	Philippines	>50k	2
Priv-house-serv,Tech-support	Ecuador,England	<=50k	8
Priv-house-serv,Tech-support	Ecuador,England	>50k	3

Figure 13: Table representing the l-diversity for two categorical attributes

Occupation	Nationality	Income	Count
Protective-serv	Philippines	<=50k	3
Protective-serv	Philippines	>50k	2
Priv-house-serv,Tech-support	Ecuador,England	<=50k	8
Priv-house-serv,Tech-support	Ecuador,England	>50k	3
..	..	..	..
Craft-repair	Yugoslavia,Outlying-US(Guam-USVI-etc)	<=50k	4
Craft-repair	Yugoslavia,Outlying-US(Guam-USVI-etc)	>50k	1
Other-service	Yugoslavia	<=50k	3
Other-service	Yugoslavia	>50k	2

Figure 14: Table representing the t-closeness for two categorical attributes

Age	Edu-num	Hours per week	Income	Count
39.57	10.4286	6.00	<=50k	7
39.57	10.2857	11.14	<=50k	5
39.57	10.2857	11.14	>50k	2
37.43	10.1429	15.71	<=50k	7
40.17	10.3333	15.25	<=50k	8
..	..	..	..	..
52.83	12	21.67	<=50k	4
52.83	12	21.67	>50k	2
58.00	12	21.75	<=50k	6

Figure 15: Table representing the k-anonymisation for three numerical attributes

Age	Edu-num	Hours per week	Income	Count
38.53	12	28.47	<=50k	14
38.53	12	28.47	>50k	1
37.56	12	36.22	<=50k	7
37.56	12	36.22	>50k	2
40.56	12	36.00	<=50k	5
40.56	12	36.00	>50k	4
..	..	..	..	..
57.96	9	20.04	<=50k	20
57.96	9	20.04	>50k	3

Figure 16: Table representing the l-diversity for three numerical attributes

Age	Edu-num	Hours per week	Income	Count
37.00	9	35.82	<=50k	23
37.00	9	35.82	>50k	5
38.47	9	36.12	<=50k	31
38.47	9	36.12	>50k	3
40.33	9	36.13	<=50k	26
40.33	9	36.13	>50k	4
..	..	..	..	..
57.57	9	30.43	<=50k	18
57.57	9	30.43	>50k	3

Figure 17: Table representing the t-closeness for three numerical attributes

6. Comparison of the Discernability metric

The chart 18 below depicts how discernability metric value for 2 – quasi-identifiers rises when we move from k-anonymity to t-closeness. It depicts that it is better to use numerical attributes for quasi-identifiers than using categorical values. This behavior is depicted because in numerical values we can even choose median values to depict the clusters or groups. When we use Categorical values then we must fix the values of that attribute for cluster centroid or rather clustroid. This reduces the no of values that can be used. Hence the anonymization is reduced.

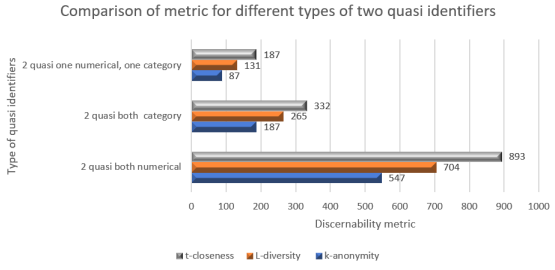

Figure 18: A bar chart showing the Comparison of metric for different types of two quasi identifiers

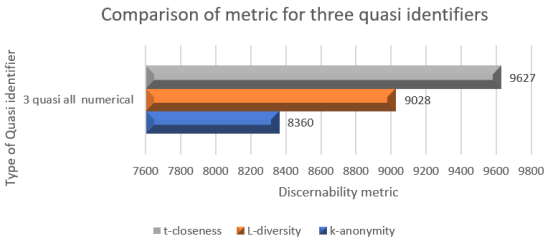

Figure 19: A bar chart showing the Comparison of metric for three quasi identifiers

7. Generative Adversarial Network

Generative Adversarial Networks are type of unsupervised machine learning method which try to generate new, synthetic instances of data that mimics the real data.[19] They are extremely popular in image, video (Deepfake) and voice generation. Generating tabular data using GANs, can produce some pretty good results. The usage of adversarial learning for text generation is promising as it provides alternatives to generate the so-called "natural" language. GANs are constructed of two neural networks: Generator and Discriminator.[22] Generator, using some random noise as input, tries to mimic the real data and Discriminator tries to classify the data into real and fake. It could be said that they are each other's enemies. Both Neural Networks (Generator and Discriminator) are trained separately through backpropagation with regards to their loss.

Conditional GAN or cGAN is a type of Generative Adversarial Network which adds the label y as an additional parameter to the generator in hope that the corresponding data will be generated. The labels are also added to the discriminator input to distinguish real data better. In a cGAN, the generator receives an additional input, the conditioning information, and the random noise used as the latent code. The generator model generates new data, similar to that of the problem domain, while the discriminator model tries to identify whether the provided example is fake or real.[11]

As a next step to feed the dataset to Generative AI models, we have developed a GAN – Generative adversarial network. We will show that the results of the Generative AI model will be similar for both Original dataset and anonymized dataset.

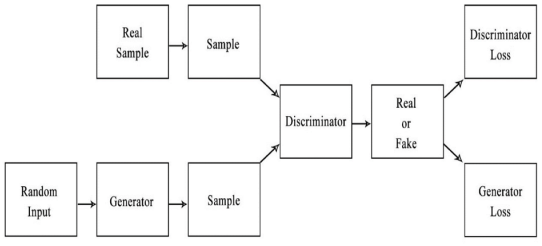

Figure 20: Architecture diagram of Generative Adversarial network

Algorithm 4 Training GAN Model

Initialization:
Define GAN architecture parameters:
    Set *latent_dim* (size of noise vector) and *out_shape* (vector size of the output).
Set up the optimizer:
    Initialize the Adam optimizer with a learning rate and a beta_1 value of 0.5.
Create the discriminator model using `self.discriminator()`.
Compile the discriminator with binary cross-entropy loss and the optimizer.
Create the generator model using `self.generator()`.
Combine generator and discriminator in a combined model, ensuring the discriminator is not trainable during generator training.
Training:
Define training parameters: epochs, batch_size, sample_interval, and the boolean *sampling*.
valid ← array of ones ▷ Ground truth for real data
fake ← array of zeroes ▷ Ground truth for fake data
for epoch from 1 to epochs **do**
    for each batch in the dataset **do**
        For the discriminator:
        if *sampling* **then**
            Sample 8 positive and (batch_size − 8) negative examples.
        else
            Sample data randomly.
        end if
        Obtain real data (samples) and labels (labels), and shuffle them.
        For the generator:
        Generate random noise from a normal distribution of shape (batch_size, *latent_dim*).
        Use the generator to create synthetic samples from noise and labels.
        Label smoothing:
        Apply label smoothing to valid and fake labels.
        Train the discriminator:
        Train the discriminator with real samples using valid labels.
        Train the discriminator with synthetic samples using fake labels.
        Calculate average discriminator loss as mean of both losses.
        Train the generator:
        Make the discriminator untrainable.
        Train the generator using the combined model with noise and randomly generated labels, targeting valid labels.
        Logging and Monitoring:
        Print discriminator and generator losses periodically based on *sample_interval*.
        Store losses for future analysis.
    end for
end for
Testing:
Feed test dataset to GAN model to predict class labels.
Visualize confusion matrix for GAN performance on test dataset.

8. Results of Generative Adversarial Network

8.1. For original dataset

	precision	recall	f1-score	support
0	0.90	0.94	0.92	7452
1	0.77	0.66	0.71	2317
accuracy			0.87	9769
macro avg	0.84	0.80	0.82	9769
weighted avg	0.87	0.87	0.87	9769

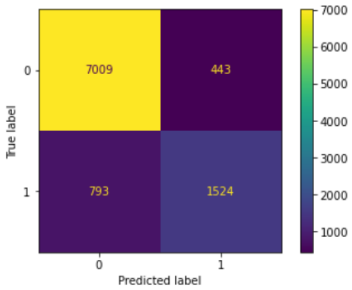

Figure 21: Confusion matrix of LGBM classifier on original dataset

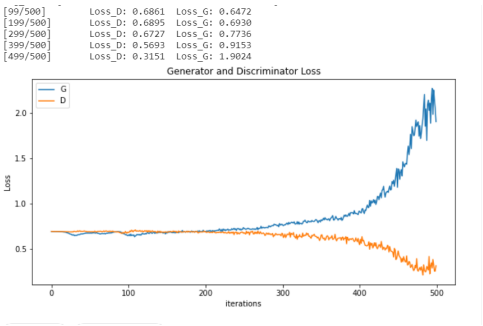

Figure 22: The Loss of Discriminator and Generator for 500 Epochs

	precision	recall	f1-score	support
0	0.76	0.91	0.83	7452
1	0.25	0.10	0.15	2317
accuracy			0.72	9769
macro avg	0.51	0.50	0.49	9769
weighted avg	0.64	0.72	0.67	9769

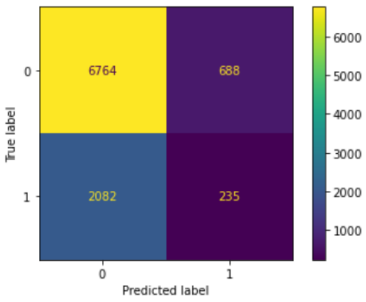

Figure 23: Confusion matrix of LGBM Classifier on Generated dataset

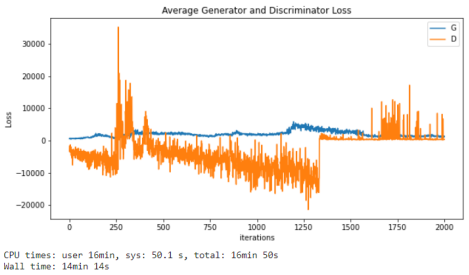

Figure 24: The plot showing the Loss of discriminator and generator as the epochs progress

8.2. For t-closeness applied dataset

The same LGBM Classifier, GAN was applied on t-closeness dataset to obtain these results which are similar to results obtained on original dataset.

	precision	recall	f1-score	support
0	0.89	0.95	0.92	7394
1	0.79	0.65	0.71	2375
accuracy			0.87	9769
macro avg	0.84	0.80	0.82	9769
weighted avg	0.87	0.87	0.87	9769

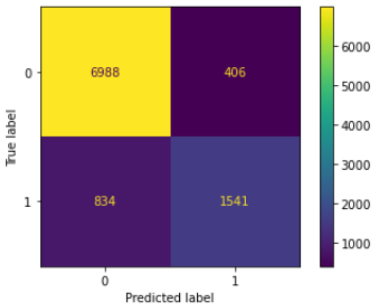

Figure 25: Confusion matrix of LGBM classifier on Anonymised dataset

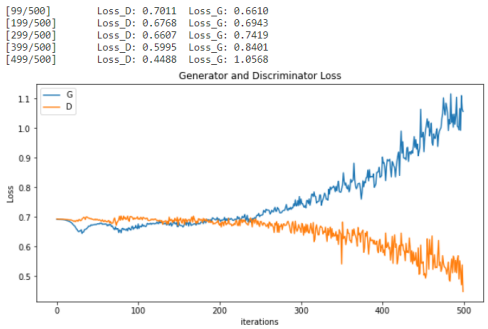

Figure 26: Loss of Discriminator and Generator for 500 epochs

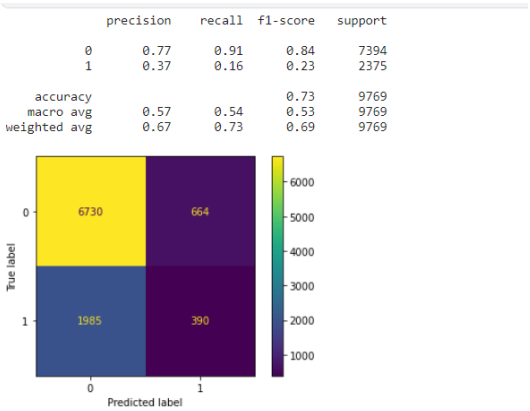

Figure 27: Confusion matrix of LGBM Classifier on Generated dataset

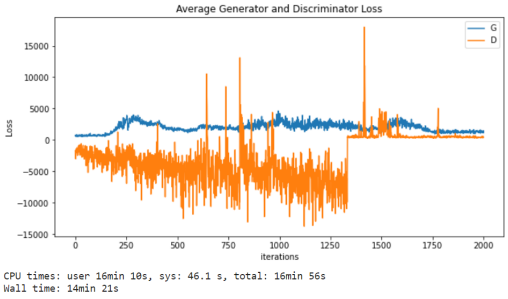

Figure 28: The plot showing the loss of Discriminator and Generator as epochs progress

The Fig 21 and Fig 25 show that the LGBM classifier gives similar results on original and anonymised dataset. This implies that the data utility is preserved in the process of anonymisation. Fig 22 and Fig 26 show that the GANs trained on original and t-closeness dataset has similar Loss_D (Loss of Discriminator) and Loss_G (Loss of Generator) are similar and the plots show the same trend for first 500 epochs of training.

Fig 23 and Fig 27 show the confusion matrix of LGBM classifier on both original and anonymised datasets give similar results . This implies the GANs trained on both the datasets has produced similar kinds of data.

Fig 24 and Fig 28 shows the Plots of the Loss_D and Loss_G for 5 Runs of 2000 epochs each for original and anonymised dataset . In both these plots , as the GAN model gets trained , the Losses drop close to zero.

9. How Attacks are Mitigated

9.1. Unsorted matching attack

An unsorted matching attack is a privacy attack that uses the order of tuples in a released table to reidentify anonymized data. It can be corrected, by randomly sorting the tuples of the solution table. Otherwise, the release of a related table can leak sensitive information.

age	edu-num
39	13
50	13
38	9
53	7
28	13
37	14
49	5
52	9
31	14
42	13

PT

age	edu-num
38.49435	13
49.74986	13
38	9
52.49153	7
29.43481	13
38.49435	14
49	5
49.74986	9
29.43481	14
41.48542	13

AT1

age	edu-num
39	11.97636
50	10.94894
38	9
53	7.423729
28	13.29949
37	11.97636
49	5
52	10.94894
31	13.29949
42	10.60292

AT2

age	edu-num
26.69767	8.124394
25.74711	10
74.79297	13.85938
38.49435	11.97636
26.69767	8.124394
39.4799	9
29.43481	13.29949
29.43481	13.29949
26.69767	8.124394
58.4814	10.75874

FT1

Figure 29: Depiction of the original table, anonymised tables and Final Table

PT: Original Table with 2 attributes age and edu-num.
AT1: Table Anonymised with age as quasi identifier with k=5
AT2: Table Anonymised with edu-num as quasi identifier with k=5
FT: k-anonymised, l-diversified, t-closeness Table Anonymised with both age,edu-num as quasi identifier after random sorting

With Release of AT1 and subsequent release of AT2 , there is a possibility that a adversary can correlate the data based on the relative position of the tuples in the anonymised tables and original table to re-identify the tuple . This way the sensitive information may get leaked. In the approach adopted in the project , we have introduced three levels of protection. We have applied l-diversity , t-closeness techniques on the top of k-anonymity to strengthen the privacy. Instead of anonymizing with single quasi identifier, we have used two quasi identifiers . We have also randomly sorted the resulting table before releasing to prevent re-identification. This is shown in FT in the figure 29.

9.2. Homogeneity attack

This attack exploits the scenario where all individuals within a k-anonymous group share the same sensitive attribute value. L-diversity addresses this by ensuring a certain level of diversity for sensitive attributes within each group. This makes it harder to link a specific sensitive value to an individual within the group.

age	edu-num	income
25	10	<=50k
25	9	<=50k
25	10	<=50k
25	11	<=50k
25	10	<=50k
26	11	<=50k
26	9	<=50k
26	9	>50k
26	11	>50k
26	10	<=50k

k-anonymised table k=5

age	edu-num	income
25.49	10	>50k
25.49	10	<=50k
25.49	10	<=50k
25.49	10	<=50k
25.49	10	<=50k
25.49	10	<=50k
25.49	10	<=50k
25.49	10	<=50k
25.49	10	>50k
25.49	10	<=50k

l-diversified table k=5 l=2

Figure 30: Depiction of K-anonymised and L-diversified table

The cluster with age group 25 has only ¡=50k as the sensitive attribute value. This can be prone to homogeneity attack. When we apply l-diversity , this cluster has got combined with another cluster with age 26 to give mixture of ¡=50k and ¿50k as the sensitive attribute values with age 25.49 and edu-num 10. This is shown in Fig 30 above.

9.3. Skewness Attack

This attack leverages the skewed distribution of a sensitive attribute in the overall data. Even with l-diversity, if a sensitive attribute is heavily skewed in one direction, an attacker might be able to make educated guesses about individuals within a group. T-closeness addresses this by requiring the distribution of the sensitive attribute within each group to be statistically close to the overall distribution.

We have a dataset with income information (sensitive attribute) labeled as more than 50k (positive) and less than or equal to 50k (negative). The overall population is skewed, with 80% earning less than or equal 50k and 20% earning more than 50k.

l-diversity might create anonymized groups with an equal number of positive and negative records . This appears diverse based on the number of positives and negatives within the group. If someone belongs to a group with 50% chance of earning more than 50k, that's significantly higher than the actual 20% chance in the real population. This is a privacy risk for individuals in that group.

How t-closeness Strengthens Privacy:

t-closeness tackles this problem by considering the overall income distribution. Here's the process:

Setting the Threshold (t): We define a threshold (t) that specifies how much the income distribution within a group can differ from the overall distribution. For example, a small t might require the group's distribution to be very close to the overall one, while a larger t might allow for some deviation.

Comparing Distributions: t-closeness compares the proportion of more than 50k earners within each group to the overall proportion (20%) in the entire dataset.

10. Comparison to previous works done in this field

We have compared our work with the existing work on the basis of 3 parameters namely Purpose for using Privacy technique, Amount of Protection offered against various attacks, Data utility.

10.1. Purpose for using the Privacy Techniques:

Existing literature surveyed typically uses k-anonymity for traditional data systems, which provides a baseline level of privacy protection. This approach is susceptible to homogeneity attacks and may lead to significant data utility loss due to excessive generalization or suppression.

Our work employs a combination of k-anonymity, l-diversity, and t-closeness for anonymizing datasets before feeding them to Generative AI models. This approach addresses the limitations of k-anonymity by providing more robust privacy protection and controlling the distribution of sensitive attributes.

10.2. Privacy Protection:

Literature Surveyed rely on k-anonymity, which may be sufficient for some data systems but can lead to privacy vulnerabilities such as homogeneity attacks and attribute disclosure. Attacks such as Background knowledge attacks can be launched on k-anonymised datasets because of the lack of diversity in the sensitive data attribute in the clusters. There is a need to protect the data against such kinds of attacks. Hence, we need further level of security.

By integrating k-anonymity, l-diversity, and t-closeness, our approach offers enhanced privacy protection. l-Diversity addresses homogeneity issues, while t-closeness controls the distribution of sensitive attributes, making it more resistant to distribution attacks and providing stronger privacy guarantees. Also we are able to preserve the distribution of original dataset at the same time anonymise it for privacy purposes.

10.3. Data Utility:

The use of k-anonymity alone may lead to a loss of data utility due to extensive generalization, affecting the usefulness of the data for machine learning tasks. It may perturbate the data distribution during the clustering process. Hence this leads to changes in the analysis that may be dependent on the original data distribution. By combining multiple anonymization techniques, our approach aims to strike a better balance between privacy protection and data utility. l-Diversity and t-closeness allow for more fine-grained control over the data, preserving its utility while ensuring privacy.

11. CONCLUSION

In conclusion, protecting data privacy is paramount, especially when utilizing sensitive data for tasks like training Generative Adversarial Networks (GANs). This project explored the effectiveness of k-anonymity, l-diversity, and t-closeness techniques, implemented through the Mondrian process and generalization process. Dataset anonymized using Mondrian, is fed into a GAN model. The results of the GAN trained on both original and anonymized dataset gives similar and satisfactory results proving that the data utility, distribution and semantic meaning are unperturbed and maintained. Discernability metric was used as a evaluation metric to measure the level of anonymization. We have also discussed the different case studies for different types and number of quasi-identifiers. The effect of these variations on the discernability metric and hence the level of anonymization has been discussed extensively. Confusion matrices and plots are used to graph the loss of discriminator and generator to show the performance of the GAN. The LGBM Classifier has been used to classify the generated data of GAN to compare against the classification results of real datasets This approach successfully addressed the challenge of balancing privacy with data utility by ensuring a certain level of indistinguishability among data points while preserving the information crucial for the GAN's training process. The project anonymizes the dataset using generalization technique, a different approach to anonymization which puts the datapoints into data intervals to achieve the goal of data masking. Overall, this project demonstrates the potential of these anonymization techniques for facilitating the use of sensitive data in machine learning applications while upholding data privacy standards.

References

[1] Ahmed Abdeen Hamed, Malgorzata Zachara-Szymanska, XindongWu,Safeguarding authenticity for mitigating the harms of generative AI: Issues, research agenda, and policies for detection, fact-checking, and ethical AI, iScience ,Volume 27, Issue 2,2024.

[2] . J. -H. Weng and P. -W. Chi, "Multi-Level Privacy Preserving K-Anonymity," 2021 16th Asia Joint Conference on Information Security (AsiaJCIS), Seoul, Korea, Republic of, 2021, pp. 61-67.

[3] Mei Yu, Yu Du, Tianyi Xu, Jian Yu and Yaqing Liu, "A K-anonymity model with strongly identifiable attributes," 2013 10th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), Shenyang, China, 2013, pp. 428-432 .

[4] L.Sweeney.Achieving k-anonymity privacy protection using generalization and suppression. International Journal on Uncertainty,Fuzziness and Knowledge-basedSystems,10(5),2002;571588.

[5] Jin Qian, Haoying Jiang, Ying Yu, Hui Wang, DuoqianMiao,Multi-level personalized k-anonymity privacy-preserving model based on sequential three-way decisions, Expert Systems with Applications, Volume 239,2024.

[6] . Fadel M. Megahed, Ying-Ju Chen, Joshua A. Ferris, Sven Knoth and L. Allison Jones-Farmer (2024) How generative AI models such as ChatGPT can be (mis)used in SPC practice, education, and research? An exploratory study, Quality Engineering, 36:2, 287-315.

[7] Huang, Weijie Chen, Xi. (2024). A Levy Scheme for User-Generated-Content Platforms and Its Implication for Generative AI Providers.

[8] Gangarde, Rupali Shrivastava, Deepshikha Sharma, Dr Amit Tandon, Tanishka Pawar, Dr. Ambika Garg, Rachit. (2022). Data anonymization to balance privacy and utility of online social media network data. Journal of Discrete Mathematical Sciences and Cryptography. 25. 829-838. 10.1080/09720529.2021.2016225..

[9] Nah, Fiona Zheng, Ruilin Cai, Jingyuan Siau, Keng Chen, Langtao. (2023). Generative AI and ChatGPT: Applications, challenges, and AI-human collaboration. Journal of Information Technology Case and Application Research. 25. 1-28. 10.1080/15228053.2023.2233814.

[10] . J. Cao et al., "Hiding Among Your Neighbors: Face Image Privacy Protection with Differential Private k-anonymity," 2022 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Bilbao, Spain, 2022, pp. 1-6.

[11] . Zhang, G., Liu, B., Zhu, T. et al. Visual privacy attacks and defenses in deep learning: a survey. Artif Intell Rev 55, 4347–4401 (2022). https://doi.org/10.1007/s10462-021-10123-y

[12] Brijesh B. Mehta, Udai Pratap Rao,Improved l-diversity: Scalable anonymization approach for Privacy Preserving Big Data Publishing,Journal of King Saud University - Computer and Information Sciences,Volume 34, Issue 4,2022,Pages 1423-1430,ISSN 1319-1578

[13] T. Xu and Z. Wei, "Waveform Defence Against Deep Learning Generative Adversarial Network Attacks," 2022 13th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP), Porto, Portugal, 2022, pp. 503-508, doi: 10.1109/CSNDSP54353.2022.9907905.

[14] Muthulakshmi, V., Francis H. Shajin, J. Dhiviya Rose, and P. Rajesh. "Generative Adversarial Networks Classifier Optimized with Water Strider Algorithm for Fake Tweets Detection." IETE Journal of Research, (2023), 1–16. doi:10.1080/03772063.2023.2172466.

[15] P. Cobelli, S. Nesmachnow and J. Toutouh, "A comparison of Generative Adversarial Networks for image super-resolution," 2022 IEEE Latin American Conference on Computational Intelligence (LA-CCI), Montevideo, Uruguay, 2022, pp. 1-6.

[16] Anande, T. J., Al-Saadi, S., and Leeson, M. S. (2023). Generative adversarial networks for network traffic feature generation. International Journal of Computers and Applications, 45(4), 297–305.

[17] Z. Shi, J. Teng, S. Zheng and K. Guo, "Exploring the Effects of Various Generative Adversarial Networks Techniques on Image Generation," 2023 IEEE 11th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), Chongqing, China, 2023, pp. 1796-1799.

[18] V. s. Krishna Katta, H. Kapalavai and S. Mondal, "Generating New Human Faces and Improving the Quality of Images Using Generative Adversarial Networks(GAN)," 2023 2nd International Conference on Edge Computing and Applications (ICECAA), Namakkal, India, 2023, pp. 1647-1652.

[19] Romero Moreno, F. (2024). Generative AI and deepfakes: a human rights approach to tackling harmful content. International Review of Law, Computers Technology, 1–30.

[20] Jabar, M., Chiong-Javier, E., Pradubmook Sherer, P. (2024). Qualitative ethical technology assessment of artificial intelligence (AI) and the internet of things (IoT) among filipino Gen Z members: implications for ethics education in higher learning institutions. Asia Pacific Journal of Education, 1–15. .

[21] Long, William J., and Marc Pang Quek. "Personal Data Privacy Protection in an Age of Globalization: The US-EU Safe Harbor Compromise." Journal of European Public Policy 9, no. 3 (2002): 325–44.

[22] Alankrita Aggarwal, Mamta Mittal, Gopi Battineni,Generative adversarial network: An overview of theory and applications,International Journal of Information Management Data Insights, Volume 1, Issue 1,2021,100004,ISSN 2667-0968,

[23] Yu, Liangwen and Zhu, Jiawei and Wu, Zhengang and Yang, Tao and Hu, Jianbin and Chen, Zhong. (2012). Privacy Protection in Social Networks Using l-Diversity. 435-444.

[24] Latanyasweeney,. (2012). ACHIEVING k-ANONYMITY PRIVACY PROTECTION USING GENERALIZATION AND SUPPRESSION. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems.

[25] A. Machanavajjhala, J. Gehrke, D.Kifer, and M. Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. In Proc. 22nd Intnl. Conf. Data Engg. (ICDE), page 24, 2006.