Differentially Private Synthetic Data Generation using Large Language Models

SAAD KHALIL, University of Twente, The Netherlands

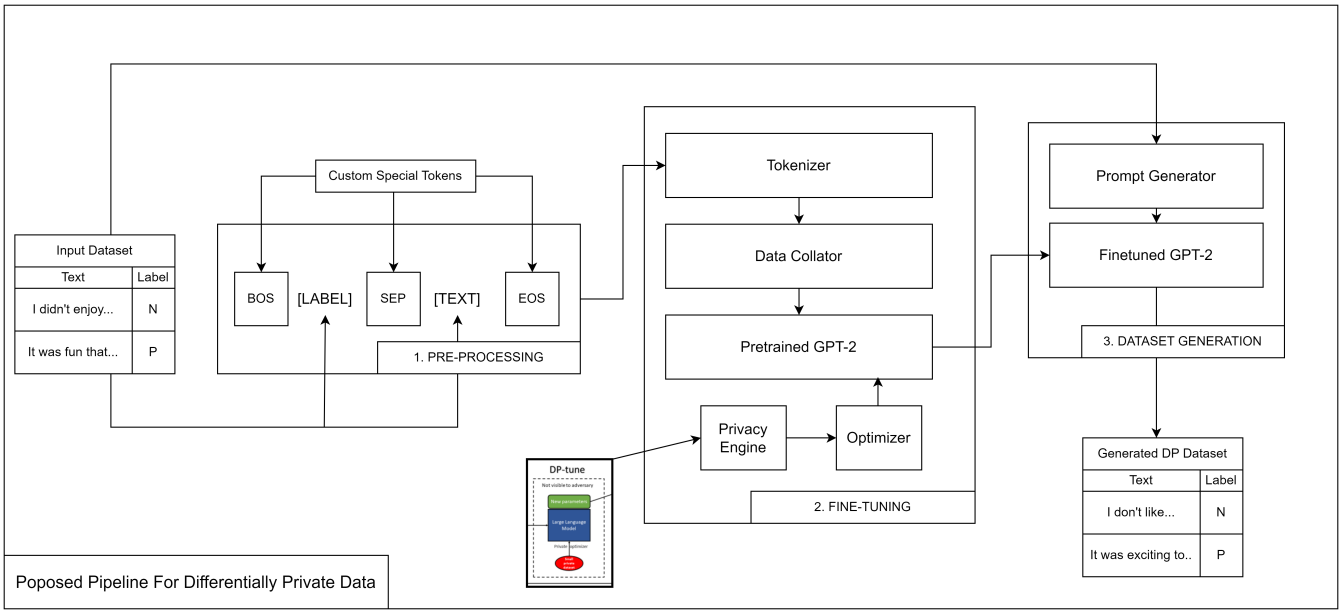

Fig. 1. The proposed end-to-end pipeline

This paper presents an approach that integrates differential privacy with Large Language Models (LLMs) for generating synthetic data, focusing on sensitive content such as user chats. Unlike traditional methods reviewed in the literature, our methodology employs a more heuristic approach to guide generation, thereby enhancing utility and fidelity while maintaining computational efficiency. Our method conditions LLMs with labels and the initial words of input text using special tokens to ensure the preservation of context and semantic integrity, a crucial aspect for sensitive data sources. This approach contrasts with complex data generation methods that are computationally intensive for larger datasets and do not guarantee high utility and fidelity, nor the preservation of style in the synthetic dataset. We assess data utility and fidelity through a comparative analysis of the original and generated synthetic datasets, focusing on semantic and syntactic properties. We notably observe a decrease in data utility and fidelity as privacy levels increase. Non-private synthetic data show a 7% loss in utility scores, while private synthetic data show a 35% loss, with semantic similarity scores reflecting similar trends. This research underscores the complexities of balancing privacy against the functional usefulness of synthetic data. Our findings highlight the challenges in managing sensitive information, particularly private chats, emphasizing the importance of balancing privacy protection with the effectiveness of synthetic data. This balance is critical for advancing research methodologies in sensitive fields without compromising data confidentiality.

Additional Key Words and Phrases: Differential Privacy, LLMs, Text Generation, Synthetic Data

1 INTRODUCTION

Advancements in AI, particularly in Large Learning Models, have significantly impacted synthetic data generation. Gartner's 2021 report [1] highlights this trend: synthetic training data constituted only 1% of all data but is projected to rise to 60% by the end of 2024. This shift underscores the growing relevance of synthetic data across various sectors. In healthcare, synthetic data aids in training medical professionals. In the financial industry, it plays a crucial role in risk mitigation and fraud detection. Compared to traditional data, synthetic data is more cost-effective and ethical. Its primary advantage lies in preserving privacy, as it eliminates the risk of exposing sensitive real-world data.

The University of Twente presents a case in point. During COVID-19, the University adopted Discord, a messaging and VoIP platform, for student-teacher interactions. These communications offer valuable insights into evolving educational dynamics since the introduction of AI tools like ChatGPT. However, utilizing this data for research is constrained by the GDPR (General Data Protection Regulation, EU 2016/679), which mandates strict data privacy guidelines and requires explicit consent for data usage. This is where synthetic data becomes vital. By ensuring true anonymization—making it impossible to trace data back to individuals—researchers can be compliant with GDPR restrictions. This approach enables the ethical use of data in research, particularly in sensitive areas where privacy is important.

In 2006, Netflix released an anonymized dataset of subscriber movie ratings for the Netflix Prize challenge. This incident unintentionally revealed the limitations of traditional data anonymization methods. Narayanan and Shmatikov [11] demonstrated that by using additional information from IMDb, they could de-anonymize this dataset. Their research exposed a critical vulnerability: anonymized datasets could be re-identified when merged with external data, compromising privacy. Responding to this challenge, Dwork et al. [7] introduced Differential Privacy. This method offers a mathematical guarantee of individual privacy protection. It significantly reduces the likelihood of identifying individual data within a dataset. Differential Privacy represents a departure from conventional anonymization, providing a more reliable and measurable privacy safeguard. It not only defends against linkage attacks, where separate data sources are combined to identify individuals, but also offers universality, composability, and flexibility. However, as the use of synthetic data grows, integrating differential privacy into the synthetic data generation process becomes crucial. This integration is essential to maintain the highest standards of privacy. Yet, the exploration of this integration remains underdeveloped. Further research in this area is vital to ensure that synthetic data upholds privacy protections.

To address this, our research proposes an approach to integrate differential privacy with Large Language Models, aiming to generate synthetic data that maintains the utility of real data whilst ensuring that it is private. This research will involve modifying the Large Language Models (LLMs) training process to incorporate differential privacy and evaluating the generated synthetic data against the original dataset in terms of privacy, utility for its potential real-world application, and fidelity in replicating key characteristics of the original dataset.

Our research aims to integrate differential privacy into the training process of Large Language Models (LLMs), thereby generating synthetic data that retains the real data's utility while ensuring privacy. This integration will involve modifying the LLMs' training process to implement differential privacy using dp-transformers [17] library. Our evaluation will focus on three critical aspects: the privacy level of the synthetic data compared to the original dataset, its utility for potential real-world applications, and its fidelity in replicating the original dataset's key characteristics.

2 PAPER STRUCTURE

The next section, Related Works, provides an overview of the existing research. The Background section provides the theoretical background necessary for understanding this research. The Proposed Methodology section details our research approach. The Experimental Setup discusses this approach technically. The Results section then presents the research findings, leading into the Discussion where these results are interpreted and their implications are explored. The paper concludes with the Conclusion, summarizing the research and its broader impact.

3 RELATED WORKS

The study on Differentially Private Data Synthesis [20] presents an innovative algorithm for creating synthetic datasets that are differentially private. This research uniquely addresses the delicate balance between reducing information loss and retaining significant data correlations, a critical aspect in ensuring both data utility and privacy.

The paper also reviews insights from the NIST Differential Privacy Data Synthesis Challenges [16], which detail practical experiences in applying differential privacy to data synthesis. This research contributes significantly to understanding the real-world challenges and solutions in maintaining data privacy while ensuring the informativeness of the data.

Furthermore, the research on Synthetic Text Generation with Differential Privacy [19], presented at the Association for Computational Linguistics meeting, offers a practical approach to generating synthetic text within the boundaries of differential privacy. This study aligns closely with creating privacy-preserving synthetic data using Large Language Models, emphasizing the maintenance of data utility alongside privacy. Furthermore, the research on Synthetic Text Generation with Differential

This research addresses critical gaps in existing literature, particularly in the conditioning of Large Language Models (LLMs) using labels and initial text segments for processing sensitive data. Previous studies have not thoroughly investigated this methodology, nor have they adequately analyzed the balance between maintaining privacy and preserving the utility of synthetic data generation. Furthermore, the generation of synthetic data from highly sensitive sources like personal conversations is a domain that has remained largely unexplored in past research.

A key contribution of our work lies in its practical validation, which extends the utility of these methods beyond theoretical constructs and into real-world applications. This aspect of our research is particularly crucial, as it demonstrates the feasibility and effectiveness of these methodologies in practical scenarios, something that has been lacking in prior studies. Therefore, our research contributes to advancing the field of synthetic data generation under the constraints of differential privacy, paving the way for future investigations and practical implementations in this area.

4 BACKGROUND KNOWLEDGE

4.1 Large Language Models

Large Language Models underwent a significant transformation with the introduction of the Transformer model by Vaswani et al. [15] in 2017. Central to this model is an attention mechanism that revolutionizes the processing of sequential data. This mechanism dynamically allocates varying levels of focus to different segments of input data, depending on their assessed relevance. Such an approach allows Transformer-based models to contextualize each word within the entire sequence, thereby enhancing the interpretation of data. This contextual understanding is pivotal in language modeling, particularly in generating synthetic data that closely mirrors the original dataset. The comprehension afforded by the Transformer model elevates the fidelity of synthetic data generation, ensuring greater accuracy and relevance in linguistic applications. This attention mechanism is given by the following equation:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

In the attention mechanism, the components Queries (Q), Keys (K), and Values (V) play crucial roles, represented as matrices. The queries, denoted as Q, are representations of the current token, effectively capturing its context within the input sequence. The keys, represented by K, encapsulate the representations of all tokens in the input sequence and are instrumental in computing attention scores. Values, indicated by V, are also representations of the input tokens but are distinct from keys in their function; they are utilized to construct the output of the attention layer. The dimensionality of the keys d_k, is a critical parameter that influences the effectiveness of the attention mechanism by determining the scale of the dot products used in calculating attention scores. This dimensionality plays a vital role in balancing the model's sensitivity to the input sequence's various features.

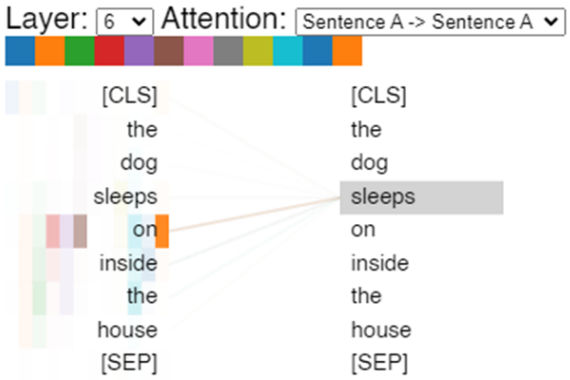

Fig. 2. Attention for different words using BertViz

The Transformer model is composed of two principal components: encoders and decoders. The encoders process input sequences in parallel, transforming them into a set of attention vectors. These vectors are then utilized by the decoder, in conjunction with its own input, to generate the output sequence. Within this architecture, BERT and GPT can be used as exemplary models. BERT uses the encoder to process text inputs, enabling it to contextualize a word within the range of surrounding words. Conversely, GPT uses the decoder component to sequentially generate text, predicting each subsequent word based on the cumulative context of preceding words.

The computation of attention weights for each token as given by the previous equation is a critical feature of this model. These weights allow the model to retain and integrate contextual information throughout the entire sequence. Consequently, the final representation of each token is informed by the entire sequence, rather than being limited to local neighboring tokens. This ability to incorporate and utilize extensive context makes the Transformer model particularly adept at complex language tasks, such as text generation, by providing a nuanced and comprehensive understanding of language.

Figure 2 shows a representation of the attention mechanism from the sixth layer of a transformer-based model. Specifically, this figure illustrates a self-attention heatmap for a single sentence input using BertViz, a visualization tool for attention. Each word, or token, in the sentence ("the", "dog", "sleeps", "on", "inside", "the", "house") is aligned in two columns, with the left column representing the focus tokens and the right column representing the context tokens to which the attention is being paid. The attention scores are visualized through the gradient shading and connecting lines between the tokens. For instance, the word "on" is connected to "sleeps" with a prominently thick line and a highlighted box, indicating a strong attention link, suggesting that in the context of this sentence, the model has learned that "on" and "sleeps" have a significant contextual relationship. The special tokens "[CLS]" and "[SEP]" are also visible, denoting the start and end of the input sequence, respectively. These tokens are part of the input formatting convention for certain transformer models like BERT, where "[CLS]" is used for classification tasks and "[SEP]" is used to separate or conclude input sequences. This visualization helps understand and visualize how transformer models process and relate different parts of the input data to generate an output or make a prediction.

4.2 How is text generated?

Text generation in models like the Generative Pre-trained Transformer (GPT) begins with an initial text prompt. This prompt undergoes tokenization-splitting text into smaller tokens and is subsequently converted into a vector of numerical representations through embeddings. These embeddings capture both the semantic and syntactic properties of the prompt. The vector then progresses through the transformer layers, ultimately reaching the Language Modeling Head. This component contains comprehensive information about all words known to the model and plays a crucial role in the generation process.

In the Language Modeling Head, the vectors are mapped from their numerical representations back to actual words, utilizing the hidden states derived from the transformer layers. The head calculates the probability of each word in the model's vocabulary being the next word in the sequence, given the initial prompt. Decoder models employ an autoregressive approach, wherein the model incorporates its previously generated outputs as part of the input for generating subsequent outputs. After generating a word, this word is appended to the input sequence, and the model iterates this process for each subsequent word. Consequently, each word generated is dependent on the context provided by the preceding words in the sequence. The probability of a sequence of words in this model is determined by the product of these conditional probabilities, given as:

$$P(\text{word}_1, \text{word}_2, \ldots, \text{word}_N) = \prod_{i=1}^{N} P(\text{word}_i | \text{word}_1, \text{word}_2, \ldots, \text{word}_{i-1}) \tag{2}$$

4.3 Differential Privacy

Differential Privacy is a privacy-preserving framework that offers a formalized privacy guarantee. This guarantee asserts that the addition or removal of a single data point within a dataset does not substantially influence the outcome of an analysis. The framework

achieves this by the addition of controlled random noise to the data, thereby masking individual data points. One of the key strengths of Differential Privacy is its resilience against linkage attacks, where an adversary attempts to re-identify individuals in anonymized datasets. Additionally, its flexibility and composability allow it to be integrated with other privacy-preserving frameworks effectively.

From a mathematical perspective, a randomized mechanism M adheres to ε-differential privacy if, for any two datasets D and D' that differ by at most one element, and for all outcome sets S within M's output space, the following condition holds:

$$\Pr[M(D) \in S] \leq e^{\varepsilon} \times \Pr[M(D') \in S] \tag{3}$$

This inequality signifies that the probability of any outcome from dataset D occurring within set S is bounded by the exponential function of ε times the probability of the same outcome from dataset D'. Here, ε (epsilon) is a non-negative parameter that quantifies the privacy loss, with lower values indicating stronger privacy guarantees.

The effectiveness of Differential Privacy depends on several key factors:

(1) Noise Distribution: The type of noise added, such as Laplace or Gaussian noise, plays a crucial role in masking individual data points.
(2) Privacy Budget: Defined by ε, the privacy budget determines the strength of the privacy guarantees. A lower ε value typically means stronger privacy protection but may impact the utility of the data.
(3) Query Sensitivity: This refers to the potential change in output value with the removal of any one record from the dataset. Differential Privacy is most effective with low-sensitivity queries, where the output is less affected by changes in individual data points.

4.4 DP-SGD

Differentially Private Stochastic Gradient Descent (DP-SGD) is an adaptation of the traditional gradient descent algorithm, modified to incorporate differential privacy, suitable for training machine learning models. The fundamental intuition behind this approach involves adding noise to the gradients during the model's learning process. In practice, gradients are calculated for a small batch of data and then clipped to a predefined threshold. This clipping serves to limit the influence of any single data point, ensuring that individual contributions do not disproportionately affect the gradient calculations. Subsequently, noise is introduced into the aggregated gradients, with the magnitude of this noise being directly proportional to the desired privacy level ε. These modified gradients, now imbued with noise, are utilized for updating the model parameters. The updated model parameters are calculated as follows:

$$\theta_{\text{new}} = \theta_{\text{old}} - \eta \left(\frac{1}{n} \sum_{i=1}^{n} \text{Clip}_C(\nabla L(\theta, x_i)) + \mathcal{N}(0, \sigma^2 C^2) \right) \tag{4}$$

In this equation, η represents the learning rate, n denotes the number of samples in the batch, Clip_C is the clipping function applied to the gradients, and $N(0, \sigma^2 C^2)$ is the Gaussian noise added to ensure privacy. This integration of noise and clipping into the gradient descent process is pivotal in aligning the model training with differential privacy principles.

5 PROPOSED METHODOLOGY

The proposed methodology, illustrated in Figure 1, is a framework for generating a differentially private dataset using a fine-tuned Large Language Model (LLM).

5.1 Input Dataset

We focus on the synthesis of datasets with labels. Labeled datasets enable a direct comparison between models trained on synthetic data and those trained on original data. This comparison is vital for quantifying the utility of synthetic text and assessing its accuracy in replicating real-world data. While labels are not inherently required to create realistic synthetic text, they play a pivotal role in training differentially private models. Labeled datasets are particularly useful for synthetic text generation, as they provide the model with clear guidance regarding the data's context and intended outputs.

Hence, our primary hypothesis is that by appending the first three words of the input text with the label to guide generation, the model will produce synthetic text that closely mirrors the original in terms of utility and fidelity. This approach is also expected to maintain text privacy due to the integration of differential privacy. We hypothesize that this method will preserve the inherent patterns and the initial label of the text, thereby guiding the model to generate high-quality, differentially private synthetic data. This hypothesis is supported by the findings of Taub et al. (2020) [14] where they demonstrate that labeled input data can improve the utility of synthetic data by reducing uncertainty about the reliability and validity of results derived from them.

5.2 Pre-processing

Conditioning a Large Language Model (LLM), as introduced by Keskar [9], is a technique designed to generate text based on specific contexts or attributes. This method involves a strategic modification of the training data during the pre-processing phase. Each data entry is prefixed with a relevant label or attribute, followed by special tokens indicating the beginning of the sentence [BOS] and separation from the label [SEP]. Through this approach, the model is trained to generate text that is directly influenced by the prepended labels, effectively conditioning the output. When a prompt is presented with an associated label, the model demonstrates a heightened propensity to produce text that is coherent with the specified label. This technique offers refined control over the generated content without necessitating changes to the model's underlying architecture. Moreover, it enhances the model's ability to adhere to the semantics, styles, tones, and topics inherent in the training data, thereby amplifying its utility and fidelity in text generation tasks. This conditioning approach, symbolized by the sequence "[BOS] [Label] [SEP]," is instrumental in guiding the model towards generating contextually relevant and stylistically consistent text.

In this stage, we ready the input dataset for processing by the language model. Custom special tokens—BOS to denote the beginning, SEP to separate label and text, and EOS to mark the end—are

appended to each text entry. The labels (e.g., 'N' for negative, 'P' for positive) as shown in Figure 1 are also integrated, providing the model with explicit cues about the nature of the text to follow.

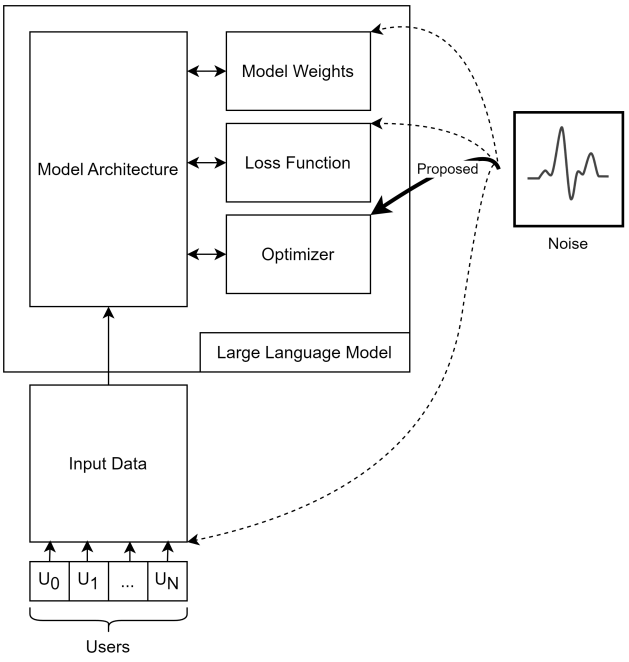

Fig. 3. The dotted arrows represent the potential areas where noise can be added. The solid arrow is the proposed area.

5.3 Fine-Tuning

In this stage, the training involves fine-tuning a pre-trained model with pre-processed data. Initially, the data undergoes tokenization using a specialized tokenizer, and then a data collator further adapts this tokenized data for training. A critical component in this process is the privacy engine, which is tasked with integrating differential privacy. This is achieved by meticulously adding noise to the training process in a controlled manner.

The underlying intuition of our approach is to develop a generalized model that does not overly rely on memorizing specific training data to achieve high accuracy. This concept aligns with the principles of differential privacy, where the objective is to enable accurate analysis while preventing the exposure of individual data contributions. In a typical Large Language Model as shown in Figure 3, essential elements include the model architecture, its weights, the loss function, and the optimizer. The optimizer plays a pivotal role in minimizing losses. Adding noise later in the training process is preferable, as introducing it too early necessitates adding more noise overall. The primary goal is to ensure that the training is both noisy and private, thereby achieving the desired privacy guarantees with a minimized addition of noise. For this reason, we propose that the optimizer, responsible for updating model parameters, is where the noise is added to ensure the preservation of privacy.

5.4 Dataset Generation

The final stage uses the fine-tuned model to create a new dataset. A prompt generator uses a prompt that retains the structure of the input data, which the fine-tuned model uses to generate new text data. This process results in a 'Generated DP Dataset'—a collection of texts paired with their corresponding labels that resemble the original data in utility and fidelity but are generated to ensure differential privacy.

6 EXPERIMENTAL SETUP

6.1 Dataset

The dataset we are using is the dair-ai/emotion dataset [13], which is publicly available. This dataset comprises English Twitter messages labeled with six basic emotions: anger, fear, joy, love, sadness, and surprise. The data fields in this dataset include a "text" string feature and a "label" classification label, with the labels representing the six emotions. The dataset is available in two configurations: a split version with a total of 20,000 examples divided into train, validation, and test splits, and an unsplit version with a total of 416,809 examples in a single train split. The dataset is intended for educational and research purposes only, aligning with the objectives of this project. Specifically, this dataset, which consists of English Twitter messages labeled with emotions like joy, sadness, anger, etc., will be utilized to test and evaluate the model's ability to generate synthetic data. By applying differential privacy, we can explore the model's capacity to produce text that retains the original data's utility while ensuring privacy, especially in contexts involving sensitive personal information. This approach will help assess the effectiveness of integrating differential privacy with large language models in generating privacy-preserving synthetic data.

6.2 Model Configuration

In our experimental setup, we utilize two distinct models: Bert-base-uncased [5] and GPT2[12]. The selection of GPT2 is strategic, chosen for its advantageous balance between computational efficiency and the quality of synthetic data it generates [4]. Considering that differential privacy significantly escalates computational costs, employing a relatively smaller model like GPT2 is essential to test our hypothesis effectively. We use Bert-base-uncased to perform a comparative analysis of the synthetic data against the original dataset and to compute the semantic similarity between them.

6.3 Preprocessing

The preprocessing stage comprises two critical steps. Firstly, we ensure the absence of personal or confidential information in the text inputs using Named Entity Recognition (NER). For this purpose, we deploy the Bert-base-NER model to identify and subsequently remove entities such as people and locations. This step is imperative to ensure the anonymization of the data. Secondly, we append labels to the actual text and incorporate special tokens. This is achieved through a mapping process, thereby preparing the data for subsequent stages.

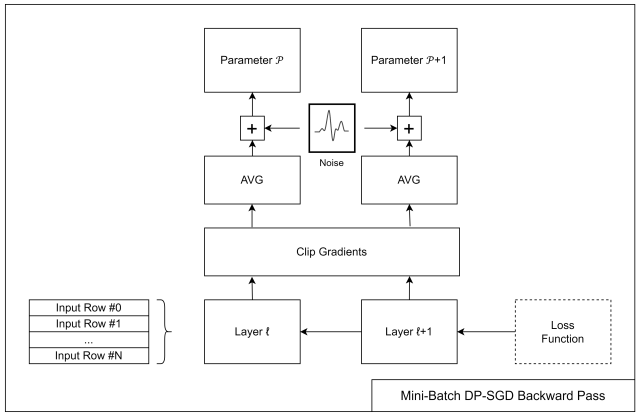

Fig. 4. Differential Privacy SGD Backwardpass

6.4 Differential Privacy Pipeline

We take a closer look at how differential privacy works in our pipeline. Initially, the dataset is partitioned into minibatches of specific sizes (e.g., 32, 64, 128). In the forward pass, this data is fed into the first layer of the network, where it operates with the model's parameters. The resulting weighted sum is then propagated through subsequent layers. The deviation of the model's predictions from the true labels or values is utilized to compute the loss value.

During the backward pass as shown in Figure 4, the loss function is propagated in reverse across each layer, leading to the update of the model parameters. Before updating the parameters, however, the gradients are subject to clipping. This is crucial, as unclipped gradients may disproportionately reflect the influence of certain data points, thereby increasing the sensitivity of the data. To manage this sensitivity, the gradients are clipped to a predetermined threshold. Subsequently, these clipped gradients are aggregated, and the predetermined noise is added to them. The resulting noisy averaged, and clipped gradients are then used to update the model parameters, ensuring the integration of differential privacy into the training process.

6.5 Experminental Process

6.5.1 Finetuning. The preprocessed training dataset is given as

$$D_{\text{train}} = \{(x_i, y_i)\}_{i=1}^{n} \tag{5}$$

We tokenize the preprocessed dataset D_{train} and truncate and pad it so it has the same sequence length. We load our model M and set it to training. We use the dptransformers library for their data collator which adapts the data preparation by automatically creating the input tensors and using their trainer class to set the training arguments and dynamically attach the differentially private optimizer to the trainer. This trainer class handles the different hyperparameters used in the finetuning. This library is a modified version of Opacus which is the standard library used in integrating differential privacy whilst allowing the integration into models like GPT2. The transformers implementation of GPT2 uses a custom layer type which is incompatible with Opacus which is why this library is used to implement differential privacy. The trainer trains and saves the model weights and parameters which can then be used to generate text. The model M is specifically fine-tuned to the semantics of dataset D_{train} as well as the labels to give us the finetuned model M'.

6.5.2 Generating Text. We generate two sets of synthetic datasets: one incorporating differential privacy and the other without it. For both scenarios, we follow the previously described procedure to produce the fine-tuned model M'. To explore the effects of differential privacy, we use varying ε epsilon settings for each version of the generated model. These models are then utilized to generate text, adhering to a specific prompting format. Specifically, we load the models and initiate text generation by prompting them with a sequence that follows our hypothesis: beginning-of-sentence (BOS) label, separator (SEP), and the initial three words. For instance, a prompt could be structured as "[BOS] sad [SEP] I will be."

We aim to generate datasets that mirror the original dataset's composition, particularly in terms of the number of samples for each label. The generated text is assigned the same class as the original text, based on our hypothesis that the synthetic text retains the same class as its original counterpart. This approach allows us to assess whether the differentially private synthetic dataset maintains the fidelity and class consistency of the original dataset while preserving privacy.

6.6 Evalutation Criteria

6.6.1 Utility. To measure the utility of the synthetic data generated, we train a classifier C_{synth}over the synthetic data D_{synth} and evaluate its performance over the original test data D_{test}. We compare the performance of this classifier with the original training dataset D_{train} which we take as the benchmark. For each dataset generated under different epsilon settings, we calculate the accuracy of the model over the test dataset. For the sake of standardization, we keep the hyperparameters identical. The hyperparameters are listed in the Appendix in Table 4.

6.6.2 Fidelity. To measure the fidelity of the synthetic dataset to the original dataset, we perform semantic similarity. We use the model sentence-transformers/all-MiniLM-L6-v2 which given an input text, outputs a vector that captures the semantic information, to calculate the similarity between the original data point and the synthetic data point. We calculate the pairwise similarities for all the pairs of sentences as this model uses sentence-level embeddings and then aggregate the scores. The intuition here is that this provides us with insight into the fidelity of the synthetic dataset.

6.6.3 Privacy. We measure the privacy of the synthetic dataset using the privacy accountant. The privacy accountant calculates the privacy loss whilst training. We measure two different measures of privacy, RDP and PRV. RDP is based on the Renyi divergence which is a generalization of Kullback-Leibler divergence. This provides a more realistic epsilon value as it allows for a more efficient privacy guarantee which leads to better utility. PRV is a more theoretical measure and provides a more robust guarantee of privacy. Both of them are measured in the training and the final calculated values are used to assess the level of privacy protection. In our training,

the RDP and PRV values always reached the desired epsilon level. Hence, we do not include this in our results.

7 RESULTS

The tables in this section show the utility and semantic similarity scores for text generated at different epsilon (ε) values, which are parameters for differential privacy. Sample-generated texts are also given to visualize these scores. The 'Benchmark' refers to the original data. The ∞ values are generated after fine-tuning with no differential privacy.

7.1 Utility

Table 1. Utility scores at different epsilon values

Epsilon (ε)	Utility
Benchmark	0.925
∞	0.861
16	0.652
8	0.592
3	0.569

This table shows that as ε decreases (which means privacy increases), the utility scores decrease. This implies that texts generated with higher privacy constraints are less useful or informative compared to the benchmark.

7.2 Fidelity

Table 2. Semantic similarity scores at different epsilon values

Epsilon (ε)	Semantic Similarity
∞	0.511
16	0.428
8	0.407
3	0.322

This table illustrates the semantic similarity between the generated texts and the original data. A higher ε value (lower privacy) retains more of the original text's semantic properties, while a lower ε value indicates that the texts are less similar in meaning to the original data.

7.3 Generated Text Samples

The table shows text samples generated, specifically for Label 3: Anger. The samples are grouped under different values of epsilon (ε). The 'Benchmark' row represents the original text for comparison, while the other rows show texts generated at varying levels of privacy (with '∞' indicating no privacy and descending values indicating increasing levels of privacy protection). As (ε) decreases, privacy increases, meaning the generated texts are less likely to reveal information about individuals in the dataset. This table demonstrates the effect of different privacy levels on the quality and coherence of generated texts.

Table 3. Text Samples generated from dair-ai/emotion dataset for Label 3: Anger

Epsilon (ε)	Text
Benchmark	i am just so sick of feeling like this and i just want opinions please nothing rude and imature
∞	i am just feeling overwhelmed with all the things that i need to do in order to get to the point where i feel like i am going to be able to do what i want to do with my life
16	i am just feeling a little annoyed at myself for not being able to keep up with what is going on in my life and i feel like i am wasting my time and energy trying to figure out what is wrong with me and how to
8	i am just feeling a bit overwhelmed with all the things that i have to do to make my life a better place for myself and my family and i feel like i am in the wrong place at the wrong time in my life and i
3	i am just feeling a little dazed and confused about what to do and how to do it and i feel like i am wasting my time and energy trying to figure out what i should do and what i need to do to get there

8 DISCUSSION

In this paper, we present an approach to generate differentially private synthetic data using Large language models. We hypothesize that we can condition the model during its training by incorporating the label and generating high-quality data by prompting the label with the first three words of the original text to guide text generation. The results indicate that the quality of data is directly correlated to the privacy level and it has a significant impact on the utility and fidelity. Specifically, as the epsilon value decreases so does the utility and the fidelity of the generated data. While non-private data shows only a 7% loss in utility as illustrated in Table 1, suggesting partial validation of our hypothesis, it's important to take into account that when a model is trained on synthetic data, it learns more efficiently, which could be because the data is less diverse and more uniform yet it remains representative of the real-world data.

At lower privacy levels, utility and fidelity significantly decrease, introducing more noise and raising the model's perplexity. This makes predictions less reliable and the synthetic data less representative of the original dataset. The benchmark represents the optimal scores without privacy constraints, providing a comparison to demonstrate the impact of increased privacy. The trade-off between privacy and data quality is apparent: higher ε values (lower privacy) maintain more utility and similarity, but as ε decreases, these metrics diminish. For instance, moving from ε=∞ to ε=16 shows a notable decline, and further decreases to ε=8 and ε=3 indicate diminishing returns for utility against increased privacy. The choice of ε reflects

a balance between the need for data quality and privacy, depending on the sensitivity of the data and the requirements of the research.

This research validates our proof of concept, demonstrating that we can successfully guide and conditionally generate text in a manner that captures the semantics as shown in Table 3 and retains the utility of the original data. However, it's important to recognize that there are limitations to this approach. While we achieved success in certain areas, such as accurately replicating data semantics, challenges remain in enhancing the model's adaptability to diverse data contexts. Further research could focus on refining this approach or exploring more sophisticated methods to condition and guide the text generation process.

The implications of our findings are particularly significant for the University of Twente, as they offer a viable solution for generating differentially private synthetic data to support research. This approach not only upholds data privacy standards but also ensures the integrity and usefulness of the data for academic research. Future research at the university level could therefore concentrate on expanding the application of this methodology to various research fields, thereby maximizing its utility and impact.

9 CONCLUSIONS

Our research validates the hypothesis that LLMs can be effectively conditioned to generate high-quality, differentially private synthetic data, but with certain limitations. The research demonstrates a clear trade-off between maintaining data privacy and the quality of synthetic data, with higher privacy levels (lower epsilon values) leading to diminished data utility and fidelity. However, the findings also underscore the potential of our approach in scenarios requiring high data fidelity and utility, particularly in sensitive areas where privacy is of utmost importance.

From a business perspective, our findings hold significant value. In industries where handling sensitive data is a norm, such as healthcare and communication services, the ability to generate high-quality synthetic data while ensuring robust privacy protection is invaluable. It enables these industries to leverage large datasets for research and development, machine learning training, and analytical purposes without risking the confidentiality of individual data. This not only helps in complying with data privacy laws but also builds trust with customers and stakeholders, enhancing brand reputation and competitive advantage

Furthermore, the study underscores the critical need for businesses to balance data utility with privacy. In an era where data is a key asset, our approach provides a pathway for companies to innovate and extract value from their data assets in an ethically responsible manner. It opens avenues for enhanced data-driven decision-making, product development, and personalized customer experiences, all while upholding the highest standards of privacy.

The research opens for further exploration in improving the conditioning and generation process of differentially private synthetic data, offering a viable solution for privacy-preserving data utilization in research and beyond.

ACKNOWLEDGMENTS

I would like to thank Muhammad Hamis Haider for his exceptional knowledge and help in this project. I would also like to thank my supervisors Dr Faizan and Dr Nacir for allowing me to work under them and for their guidance throughout this project.

REFERENCES

[1] 2023. Most AI training data could be synthetic by next year - Gartner. https://techmonitor.ai/technology/ai-and-automation/ai-synthetic-data-edge-computing-gartner

[2] Vincent Bindschaedler, Reza Shokri, and Carl A. Gunter. 2017. Plausible Deniability for Privacy-Preserving Data Synthesis. https://doi.org/10.48550/arXiv.1708.07975 arXiv:1708.07975 [cs, stat].

[3] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. https://doi.org/10.48550/arXiv.2005.14165 arXiv:2005.14165 [cs].

[4] Alberto Cano. 2018. A survey on graphic processing unit computing for large-scale data mining. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8 (2018). https://doi.org/10.1002/widm.1232

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805 http://arxiv.org/abs/1810.04805

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. https://doi.org/10.48550/arXiv.1810.04805 arXiv:1810.04805 [cs].

[7] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating Noise to Sensitivity in Private Data Analysis. In *Theory of Cryptography (Lecture Notes in Computer Science)*, Shai Halevi and Tal Rabin (Eds.). Springer, Berlin, Heidelberg, 265–284. https://doi.org/10.1007/11681878_14

[8] James Jordon, Jinsung Yoon, and Mihaela van der Schaar. 2018. PATE-GAN: Generating Synthetic Data with Differential Privacy Guarantees. https://openreview.net/forum?id=S1zk9iRqF7

[9] Nitish Shirish Keskar, Bryan McCann, Lav R. Varshney, Caiming Xiong, and Richard Socher. 2019. CTRL: A Conditional Transformer Language Model for Controllable Generation. arXiv:1909.05858 [cs.CL]

[10] H. Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. 2018. Learning Differentially Private Recurrent Language Models. https://doi.org/10.48550/arXiv.1710.06963 arXiv:1710.06963 [cs].

[11] Arvind Narayanan and Vitaly Shmatikov. 2006. How To Break Anonymity of the Netflix Prize Dataset. (2006). https://doi.org/10.48550/ARXIV.CS/0610105

[12] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners. (2019).

[13] Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. 2018. CARER: Contextualized Affect Representations for Emotion Recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Brussels, Belgium, 3687–3697. https://doi.org/10.18653/v1/D18-1404

[14] J. Taub, M. Elliot, and J. Sakshaug. 2020. The Impact of Synthetic Data Generation on Data Utility with Application to the 1991 UK Samples of Anonymised Records. *Trans. Data Priv.* 13 (2020), 1–23.

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention Is All You Need. https://doi.org/10.48550/arXiv.1706.03762 arXiv:1706.03762 [cs].

[16] Tianhao Wang, Ninghui Li, and Zhikun Zhang. 2021. DPSyn: Experiences in the NIST Differential Privacy Data Synthesis Challenges. *Journal of Privacy and Confidentiality* 11, 2 (Sep. 2021). https://doi.org/10.29012/jpc.775

[17] Lukas Wutschitz, Huseyin A. Inan, and Andre Manoel. 2022. dp-transformers: Training transformer models with differential privacy.

[18] Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. 2019. Modeling Tabular data using Conditional GAN. https://doi.org/10.48550/arXiv.1907.00503 arXiv:1907.00503 [cs, stat].

[19] Xiang Yue, Huseyin Inan, Xuechen Li, Girish Kumar, Julia McAnallen, Hoda Shajari, Huan Sun, David Levitan, and Robert Sim. 2023. Synthetic Text Generation with Differential Privacy: A Simple and Practical Recipe. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (Eds.). Association for Computational Linguistics, Toronto, Canada, 1321–1342. https://doi.org/10.18653/v1/2023.acl-long.74

[20] Zhikun Zhang, Tianhao Wang, Ninghui Li, Jean Honorio, Michael Backes, Shibo He, Jiming Chen, and Yang Zhang. 2021. PrivSyn: Differentially Private Data Synthesis. In *30th USENIX Security Symposium (USENIX Security 21)*. USENIX Association, 929–946. https://www.usenix.org/conference/usenixsecurity21/presentation/zhang-zhikun

A HYPERPARAMETERS

Table 4. Hyperparameters used in the model training

Hyperparameter	**Value**
nproc per node	2
model name	gpt2
per device train batch size	32
gradient accumulation steps	16
evaluation strategy	epoch
save strategy	epoch
log level	info
per device eval batch size	64
eval accumulation steps	1
seed	42
target epsilon	4.0
per sample max grad norm	1.0
weight decay	0.01
remove unused columns	False
num train epochs	50
logging steps	10
max grad norm	0
sequence len	128
learning rate	0.0001
lr scheduler type	constant
dataloader num workers	2