# Generation of Synthetic Data Using Large Language Models (LLMs)

**Submitted By: Siddharth Tyagi**

**COMPUTER SCIENCE & ENGINEERING**
**spec.(CLOUD COMPUTING & AUTOMATION)**

## BONAFIDE CERTIFICATE

This is to certify that Siddharth Tyagi, a student of Vellore Institute of Technology, has successfully completed the project titled "**Generation of Synthetic Data Using Large Language Models (LLMs)"** as part of assignment requirements.

The project was carried out under the supervision of [Supervisor's Name], and to the best of our knowledge, the work is original and has not been submitted for any other degree or diploma.

**PROJECT GUIDE**
**Abhisekh Unnam**

# ACKNOWLEDGMENT

I would like to express my heartfelt gratitude to everyone who helped me in the successful completion of this project titled **"Generation of Synthetic Data Using Large Language Models (LLMs)".**Firstly, I would like to thank **Abhisekh Unnam** for their valuable guidance, support, and encouragement throughout the course of this project. Their insights and suggestions were pivotal in shaping the direction of this research.

I also extend my sincere thanks to the **SHL** for providing me with the necessary resources and a conducive environment to work on this project.Special thanks to my colleagues and friends who constantly motivated me and provided critical feedback that helped improve the quality of this work.

Finally, I am deeply grateful to my family for their unwavering support and encouragement during the entire duration of this project.

# ABSTRACT

The generation of synthetic data using Large Language Models (LLMs) has emerged as a viable solution to address data scarcity in machine learning projects. This report explores the use of LLMs, specifically focusing on generating synthetic product reviews based on an Amazon reviews dataset for supplements and vitamins. The project involved two key datasets: one containing product reviews and another with product names. The main goal was to generate new, coherent, and diverse synthetic reviews using LLMs, ensuring that they mimic real human-written content while introducing controlled variations such as review length, sentiment, and product-specific information.

Two model architectures were considered: **decoder-only models** (such as GPT) for fluent text generation and **Retrieval-Augmented Generation (RAG)** for factually grounded text. The RAG model was selected for its ability to reference existing reviews, ensuring the generated text was more contextually relevant. The models were fine-tuned and deployed using **Google Vertex AI**, which provided the computational resources for training and real-time inference.

This report documents the methodology followed, challenges faced, and the results obtained from the synthetic data generation process. The generated synthetic reviews offer valuable insights and can serve as training data for further machine learning applications. The efficacy of the synthetic dataset was evaluated based on fluency, coherence, and diversity, highlighting the potential of LLMs in solving real-world data generation challenges.

# TABLE OF CONTENTS

# CHAPTER-1: PROJECT DESCRIPTION AND OUTLINE

## Introduction

The rapid advancements in Natural Language Processing (NLP) have led to the use of Large Language Models (LLMs) for generating synthetic data. This report outlines the process of generating a synthetic dataset using LLMs, with a focus on Amazon product reviews. The goal is to generate synthetic reviews that resemble humanwritten content while introducing controlled variations in factors such as review length, productspecific details, and sentiment. The generated dataset will be deployed using Google Vertex AI.

## MOTIVATION FOR THE WORK

The availability of high-quality, large-scale datasets is crucial for training effective machine learning models, especially in natural language processing (NLP) tasks such as text classification, sentiment analysis, and recommendation systems. However, obtaining real-world data can be costly, time-consuming, and often limited in scope. In cases like product reviews, companies may face challenges in gathering sufficient data to train models or evaluate systems, particularly for new products or niche categories. This gap in data availability motivates the exploration of **synthetic data generation** using **Large Language Models (LLMs)**, which can simulate realistic human-written text at scale.

LLMs, such as GPT and RAG models, have the capability to generate high-quality synthetic text that mimics real-world data, allowing businesses and researchers to overcome data limitations. By generating synthetic reviews for products, it becomes possible to enhance recommendation systems, train sentiment analysis models, and evaluate product performance without relying solely on user-generated content. The motivation behind this project is to leverage LLMs for generating a synthetic dataset of Amazon product reviews, specifically targeting supplements and vitamins, to address the data gap and provide valuable insights for machine learning tasks.

## Objective

**The primary objective of this project is to:**

- **Generate a synthetic dataset of Amazon product reviews** using Large Language Models (LLMs), specifically focused on the product category of supplements and vitamins.
- Ensure that the generated reviews resemble human-written text in terms of coherence, fluency, and relevance to the product descriptions.
- Explore and compare different LLM architectures, such as **decoder-only models (e.g., GPT)** and **Retrieval-Augmented Generation (RAG)**, to determine the most effective approach for generating factually grounded, contextually accurate reviews.
- Deploy the model using **Google Vertex AI** for efficient, scalable synthetic data generation.

This project aims to provide a solution to the problem of data scarcity in machine learning tasks by creating a rich synthetic dataset that can be used to train or evaluate models in real-world applications.

# Solution

**To achieve the objectives, the following approach was implemented:**

1. **Data Preparation**: Two datasets were used—one containing Amazon product reviews and another with product names. The data was preprocessed, cleaned, and mapped for generating relevant synthetic reviews.

2. **Model Selection**:

    - **Decoder-Only Model (GPT)**: This model was fine-tuned on the product review dataset to generate synthetic reviews in a free-flowing, creative manner.
    - **Retrieval-Augmented Generation (RAG)**: This model retrieved real-world reviews from the dataset to ground the generated text in factual information, making it more product-specific and contextually accurate.

3. **Model Training and Deployment**: The models were trained on Google Vertex AI, leveraging its infrastructure for scalable training and real-time deployment of the synthetic review generation.

4. **Evaluation**: The generated reviews were evaluated based on diversity, fluency, coherence, and factual grounding, ensuring that the synthetic data met the quality requirements for machine learning tasks.

This approach successfully addressed the challenge of data scarcity by generating a diverse, realistic synthetic dataset, demonstrating the effectiveness of LLMs in synthetic data generation for NLP applications.

# CHAPTER 2: RELATED DATASET DISCRIPTION

Amazon Reviews Dataset: This dataset contains usergenerated reviews for supplements and vitamins, capturing various aspects such as product effectiveness, side effects, and overall satisfaction.
Product Names Dataset: The second dataset contains product names that can be linked to the reviews. The goal is to generate new reviews based on these product names.

3. Architecture Choices
There are three main architectures for LLMs:
Encoderonly (BERT): Suitable for understanding and classification tasks but not ideal for text generation.
Decoderonly (GPT): Ideal for text generation, where the goal is to generate continuous text based on an input prompt.
EncoderDecoder (T5, BART): Suitable for tasks that involve both understanding and generating sequences, such as summarization or translation.

Given that our task is review generation, a decoderonly architecture (such as GPT3) or a RetrievalAugmented Generation (RAG) model will be used.

## Pros and Cons of the Dataset

**Pros:**

1. **Domain-Specific Focus**:
   - The dataset is highly focused on **supplements and vitamins**, making it easier to generate product-specific synthetic reviews that are tailored to the target market. This ensures that the data is relevant and specialized for the given domain, leading to more accurate and contextualized synthetic data generation.
2. **Sufficient Variety of Product Reviews**:
   - The dataset contains a wide range of **customer reviews**, capturing different aspects of product experiences such as effectiveness, side effects, and overall satisfaction. This variety enables the model to learn and generate diverse synthetic reviews that cover multiple dimensions of user feedback.
3. **Rich in Natural Language**:
   - The reviews are written in **natural language**, providing rich text data that can be leveraged to train the model for generating coherent and fluent synthetic text. This natural linguistic diversity is ideal for training language models to produce realistic output.

**Cons:**

1. **Limited Category Scope**:
    - The dataset is restricted to **supplements and vitamins**, limiting the model's ability to generalize to other product categories. The synthetic data generated may not apply well to different domains like electronics or clothing, thus narrowing its utility.
2. **Imbalanced Sentiment Distribution**:
    - There may be **imbalanced sentiment** in the reviews, with a potential bias toward positive or negative reviews. This can skew the generation process, resulting in synthetic reviews that do not adequately represent a balanced sentiment distribution.
3. **Noise and Inconsistencies**:
    - User-generated content often includes **typos, informal language, and inconsistencies**, which may introduce noise into the dataset. While this reflects real-world data, it can also affect the model's ability to generate high-quality, coherent text.
4. **Sparse Data for Niche Products**:
    - Some products in the dataset may have **fewer reviews**, providing limited examples for the model to learn from. This could lead to less realistic synthetic reviews for those specific products.

## Issues Identified with the Dataset

1. **Incomplete Data**:
    - Some products in the dataset may have incomplete review information, such as missing detailed descriptions or feedback on specific features. This lack of data can limit the model’s ability to generate comprehensive synthetic reviews.
2. **Varying Review Lengths**:
    - Reviews vary greatly in length, from short, one-sentence comments to long, detailed feedback. This inconsistency in review length can complicate the training process, requiring the model to handle both brief and extensive texts effectively.
3. **No Direct Sentiment Labels**:
    - The dataset lacks explicit **sentiment labels** (positive, negative, neutral) for the reviews. This absence necessitates additional preprocessing, such as sentiment analysis, to infer the sentiment, which can impact the accuracy of the generated synthetic data.
4. **Data Noise**:
    - As with any user-generated dataset, there is a significant amount of **noise** in the form of typos, grammar mistakes, and informal language. These issues need to be addressed during preprocessing to ensure that the model learns from high-quality data.

## Summary of the Dataset

The dataset used for this project comprises two key components: **Amazon product reviews** for supplements and vitamins, and **product names** that correspond to these reviews. The reviews contain rich, user-generated content that spans a variety of topics such as product effectiveness, side effects, and overall user satisfaction.

The dataset provides a strong foundation for generating synthetic reviews, given its domain-specific focus and the richness of the language used in the reviews. However, it presents certain limitations, such as an imbalanced sentiment distribution and the presence of noisy data. Additionally, the scope of the dataset is limited to a single product category, which restricts the generalizability of the synthetic data to other domains.

Despite these challenges, the dataset is suitable for training **Large Language Models (LLMs)**, particularly in the context of **retrieval-augmented generation (RAG)** or **decoder-based models**. The pre-existing structure of the data, with clear product names and associated reviews, facilitates an effective input-output mapping for generating synthetic reviews that resemble human-written text.

# CHAPTER 3: REQUIREMENTS ARTIFACTS

**Tools Used:**
 Google Vertex AI
 RAG (RetrievalAugmented Generation)
 GPTbased Models

## Tools and Technologies

The project relies on several tools and technologies to facilitate the development, training, and deployment of the synthetic data generation system.

### 1. Google Cloud Platform (GCP):

- **Google Vertex AI**: Used for training and deploying LLMs for generating synthetic reviews.
- **Google Cloud Storage (GCS)**: Used to store datasets (reviews and product names).
- **BigQuery**: Used for managing and querying large datasets in the context of retrieval-augmented generation (RAG).

### 2. Machine Learning Frameworks:

- **Hugging Face Transformers**: Provides access to pre-trained LLMs like GPT and RAG models, which are fine-tuned for this project.
- **TensorFlow/PyTorch**: Used for training and fine-tuning the LLMs.

### 3. Programming Languages:

- **Python**: The primary programming language for data preprocessing, model training, and deployment scripts.

## Summary

This chapter outlines the **requirements artifacts** necessary for the successful generation of synthetic reviews using Large Language Models (LLMs). The **functional requirements** focus on preprocessing, synthetic review generation, and evaluation, while the **non-functional requirements** address the system's scalability, performance, and security. The project operates within certain **constraints**, such as data availability and computational resource limitations. The tools and technologies used include **Google Vertex AI** for model deployment, **Hugging Face Transformers** for accessing LLMs, and **BigQuery** for data storage and retrieval.

These requirements form the foundation of the system, guiding the implementation and ensuring the generation of high-quality, synthetic product reviews that mimic real-world user feedback.

# CHAPTER 4: DESIGN METHODOLOGY AND ITS NOVELTY

1. **Data Preparation:**
The CSV files containing reviews and product names were uploaded to Google Cloud Storage (GCS). Preprocessing was done to ensure that the data was clean, and reviews were mapped to the corresponding products.

2. **Model Selection:**
RetrievalAugmented Generation (RAG) was selected for its ability to generate grounded, factbased synthetic text. RAG integrates a retrieval mechanism that ensures generated reviews are based on real data, making the synthetic content more coherent and grounded in existing reviews. FineTuned DecoderOnly Model (GPT): This was chosen as an alternative for generating reviews without external context retrieval. Finetuning on the reviews dataset allows for flexibility in generating fluent, humanlike reviews.

3. **Training the Model:**
Finetuning: The decoderbased model (e.g., GPT) was finetuned using the Amazon reviews dataset. The model was trained to generate new reviews based on product names and existing review patterns. For RAG, the reviews dataset was uploaded to BigQuery for retrieval, allowing the model to reference real data while generating new text.

4. **Evaluation of Generated Reviews:**
The efficacy of the synthetic dataset was evaluated based on the diversity of the reviews, coherence, and relevance to the product context.
Key metrics used:
Fluency: Ensures the text reads naturally.
Relevance: Assesses whether the generated review is consistent with the product details.
Diversity: Measures variation in language across generated reviews, avoiding redundancy.

5. **Deployment on Google Vertex AI**
1. Uploading Data: The preprocessed CSV files were stored in GCS.
2. Model Training: The RAG and decoderonly models were trained on Google Vertex AI.
3. Model Deployment: After training, the models were deployed on Vertex AI Endpoints, allowing for realtime generation of synthetic reviews. API calls were made to pass product names and receive generated reviews.

# CHPATER 5: DEPLOYMENT ON GOOGLE VERTEX AI

**Introduction to Google Vertex AI**
Google Vertex AI is a managed machine learning platform that allows developers to build, deploy, and scale machine learning models efficiently. It integrates various Google Cloud services, providing a unified environment for model training, deployment, and management. In this project, Google Vertex AI plays a pivotal role in fine-tuning and deploying Large Language Models (LLMs) for the generation of synthetic product reviews.

By leveraging Vertex AI's robust infrastructure, the synthetic review generation models are trained, deployed, and served at scale. Vertex AI provides tools for optimizing the entire machine learning workflow, from dataset preparation and model fine-tuning to deployment and real-time inference.

## 2. **Process Overview**

The deployment process on Google Vertex AI involves several key steps:
1. Data Preparation and Storage: Uploading the preprocessed datasets (Amazon reviews and product names) to Google Cloud Storage (GCS).
2. Model Selection: Choosing the appropriate LLM (either RAG or decoder-only models like GPT) and configuring it for training.
3. Training and Fine-Tuning: Training the model on Vertex AI with the product reviews dataset and fine-tuning it to optimize performance for synthetic review generation.
4. Deployment: Deploying the fine-tuned model as a Vertex AI Endpoint to enable real-time inference and generation of synthetic reviews.
5. Testing and Monitoring: Ensuring the deployed model is functioning as expected and monitoring its performance for optimization.

## 3. **Deployment Steps on Google Vertex AI**

### 3.1. Data Preparation and Storage
The first step in deploying on Google Vertex AI involves preparing and uploading the datasets to Google Cloud Storage (GCS):
- The two CSV files, one containing Amazon reviews and the other with product names, are uploaded to a GCS bucket.
- GCS provides easy integration with Vertex AI for seamless data access during model training.

### 3.2. Model Selection
- Choose an appropriate LLM architecture based on the project needs. For this project, we considered:
- Retrieval-Augmented Generation (RAG): This model retrieves relevant information from the reviews dataset during synthetic review generation.
- Decoder-Only Model (e.g., GPT): Used for free-form text generation.
- These models are available via Hugging Face Transformers and can be fine-tuned to the specific dataset.

### 3.3. **Model Training and Fine-Tuning**

- Creating a Custom Training Job: On Vertex AI, a custom training job is created to fine-tune the selected model (RAG or GPT) on the Amazon reviews dataset.
- Training Script: A Python script is used to load the dataset, preprocess it, and fine-tune the model using either TensorFlow or PyTorch.
- Training Configuration: Vertex AI allows configuring various training parameters, such as machine type (e.g., TPU or GPU), memory, and compute requirements. Based on the dataset size and model complexity, an appropriate configuration is selected.

### 3.4. **Model Deployment as an Endpoint**

Once the model is trained, it needs to be deployed for real-time inference. This is achieved by deploying the model as a Vertex AI Endpoint:

- Registering the Model: The fine-tuned model is registered in Vertex AI's Model Registry.
- Deploying to an Endpoint: From the Model Registry, the model is deployed to a Vertex AI Endpoint. This allows the model to be accessed via API for real-time review generation.

### 3.5. **Testing and Monitoring**

- Testing: After deployment, the system is tested to ensure that the synthetic reviews generated by the model are accurate, coherent, and diverse. Test cases are created using product names from the dataset to validate the output.
- Monitoring: Vertex AI offers built-in monitoring tools that help track the model's performance, such as latency, throughput, and accuracy. Logs and metrics are used to identify potential issues or areas for improvement.
- Scaling: Depending on the workload, the deployment can be configured to automatically scale, handling higher traffic efficiently.

## 4. **Benefits of Google Vertex AI for This Project**

### 4.1. Scalability

- Google Vertex AI provides a scalable infrastructure that supports the training and deployment of large models. As the dataset grows, Vertex AI automatically scales the computational resources required for model training and inference, ensuring the system can handle increased demand without performance degradation.

### 4.2. Ease of Integration

- Vertex AI integrates seamlessly with other Google Cloud services such as Cloud Storage (GCS) and BigQuery. This allows for efficient data retrieval, processing, and storage, reducing the overall complexity of the deployment process.

### 4.3. Real-Time Inference

- By deploying the model as an endpoint, Google Vertex AI enables real-time generation of synthetic reviews. This is essential for applications where immediate feedback is required, such as product recommendation systems or user review platforms.

## 5. **Challenges Faced During Deployment**

### 5.1. Data Processing Delays

- During deployment, there were challenges related to the latency in data retrieval from Google Cloud Storage, especially with larger datasets. This was mitigated by optimizing data pre-fetching and batching during the training process.

### 5.2. Model Performance Tuning

- Fine-tuning the model for optimal performance required several iterations, especially with the RAG model. Adjusting hyperparameters like learning rate and batch size helped improve the quality and relevance of the generated reviews.

### 5.3. Cost Management

- Managing the costs associated with high-performance hardware (such as GPUs or TPUs) was a challenge. Using on-demand pricing and autoscaling features in Vertex AI helped optimize resource allocation and reduce unnecessary costs.

## **Summary**

Deploying Large Language Models (LLMs) for synthetic data generation on Google Vertex AI provided a robust, scalable, and efficient environment for training, fine-tuning, and real-time inference. With tools like Google Cloud Storage, BigQuery, and Vertex AI's endpoints, the deployment process was streamlined and integrated into the Google Cloud ecosystem.

Google Vertex AI's scalability, ease of use, and integration capabilities made it the ideal choice for this project, allowing the fine-tuned LLMs (RAG and GPT) to generate coherent, diverse, and relevant synthetic reviews. Despite some challenges related to data latency and model tuning, the platform's features ensured efficient deployment and cost-effective scaling, making it a valuable tool for AI-driven synthetic data generation.

Does this chapter address all the necessary aspects of your deployment? Let me know if any part requires more details or adjustments!
Let me know if you'd like any additional sections or modifications!

# CHAPTER 6: CHALLENGES AND SOLUTIONS

## 6.1 Maintaining Diversity

**Challenge**
One of the key challenges in generating synthetic reviews using Large Language Models (LLMs) is maintaining diversity in the generated text. When models are trained on a limited dataset, they may produce repetitive or monotonous outputs, leading to synthetic reviews that lack the variety expected in real-world user-generated content. This is particularly critical in scenarios where diverse customer experiences and perspectives are necessary to reflect the breadth of feedback available for different products.

**Solution**
To tackle this challenge, several strategies were implemented:

**1. Data Augmentation:**
- Data augmentation techniques were applied to increase the diversity of the training dataset. This involved introducing variations in phrasing, synonyms, and sentence structures while maintaining the core sentiment and meaning of the reviews.
- By artificially expanding the dataset, the model was exposed to a broader range of expressions, helping it generate more varied outputs.

**2. Fine-Tuning with Diverse Prompts:**
- During the fine-tuning phase, the models were trained with a diverse set of prompts to stimulate varied responses. Different variations of product names, features, and user sentiments were introduced in the training data.
- Using a range of prompts enabled the model to learn how to generate reviews with different tones, lengths, and perspectives, thereby improving the overall diversity of the generated reviews.

**3. Temperature and Sampling Techniques:**
- Adjusting the temperature parameter during the generation phase allowed for controlling the randomness of the model's outputs. A higher temperature encouraged more exploration of diverse responses, while a lower temperature produced more conservative and coherent outputs.
- Additionally, employing advanced sampling techniques (e.g., top-k or nucleus sampling) provided a mechanism to generate varied outputs by selecting from a broader range of potential tokens during text generation.

**4. Evaluation Metrics:**
- To monitor the diversity of the generated reviews, specific evaluation metrics were established. These metrics focused on measuring the variation in language, sentiment distribution, and thematic content across generated reviews.
- Regular evaluations allowed for adjustments in the training process and prompt design to ensure that diversity remained a priority.

By implementing these strategies, the project successfully generated a wide range of synthetic reviews that better reflected the diverse experiences of customers, enhancing the quality and usability of the generated data.

## 6.2 Grounding the Reviews with RAG

### Challenge

Grounding synthetic reviews in factual information is crucial for ensuring the relevance and reliability of generated content. While decoder-only models can produce fluent text, they may lack the necessary context or accurate details about specific products, leading to synthetic reviews that are not aligned with real-world customer experiences. This challenge is particularly pertinent in the context of generating reviews for products, where factual accuracy and context are vital.

### Solution

To address the challenge of grounding synthetic reviews, the project employed Retrieval-Augmented Generation (RAG), which combines the strengths of retrieval mechanisms with generative models. The following approaches were implemented:

**1. Utilizing a Retrieval Mechanism:**

- RAG was designed to retrieve relevant information from the existing dataset of product reviews before generating new text. By doing so, the model was able to access actual user feedback related to specific products, which informed the generation of synthetic reviews.
- This retrieval process ensured that the generated content was not only fluent but also grounded in real customer experiences, thus increasing the overall reliability of the synthetic data.

**2. Building a Contextual Knowledge Base:**

- A contextual knowledge base was created using the existing reviews dataset, enabling the RAG model to reference factual information. This knowledge base included various attributes of the products, customer sentiments, and specific feedback, which could be retrieved during the generation process.
- The integration of this knowledge base helped ensure that generated reviews reflected accurate details about the products and maintained relevance to the queries posed by users.

**3. Fine-Tuning the Retrieval Component:**

- The retrieval component of the RAG model was fine-tuned to improve its ability to fetch relevant data from the knowledge base. This involved training the model on retrieval tasks that focused on understanding user queries and selecting the most relevant reviews from the dataset.
- By enhancing the retrieval capabilities, the model was able to generate reviews that were more contextually aware and aligned with the actual experiences of users.

By implementing RAG and these related strategies, the project successfully generated synthetic reviews that were both diverse and factually grounded, overcoming the challenges of generating realistic and contextually accurate data.

### Summary

Chapter 6 addressed two significant challenges faced during the synthetic data generation process: maintaining diversity in generated reviews and grounding those reviews using Retrieval-Augmented Generation (RAG). The solutions implemented involved a combination of data augmentation, fine-tuning strategies, retrieval mechanisms, and contextual knowledge bases. These approaches ensured that the synthetic reviews produced were varied, coherent, and factually accurate, providing valuable insights and applications in natural language processing tasks.

# CHAPTER 7: CONCLUSIONS AND RECOMMENDATIONS

## Conclusions

The project successfully demonstrated the potential of using Large Language Models (LLMs)for generating synthetic product reviews, specifically within the domain of supplements and vitamins. By leveraging both Retrieval-Augmented Generation (RAG)and decoder-only models(like GPT), the research provided valuable insights into effective methodologies for synthesizing realistic and contextually relevant text data.

**Key conclusions drawn from the project include:**

1. Efficacy of RAG: The use of RAG significantly enhanced the relevance and factual grounding of generated reviews by retrieving contextual information from the existing dataset. This allowed for the creation of synthetic reviews that closely mirrored real customer feedback.

2. Importance of Diversity: Strategies implemented to maintain diversity in generated reviews proved effective in overcoming the challenge of repetitive outputs. The application of data augmentation, diverse prompts, and advanced sampling techniques contributed to producing a rich and varied set of synthetic reviews.

3. Impact of Domain-Specific Fine-Tuning: Fine-tuning LLMs on domain-specific datasets enabled the generation of synthetic data that accurately reflects the nuances of product reviews. This specialized approach is essential for ensuring the generated content meets the needs of specific applications, such as sentiment analysis and recommendation systems.

4. Scalability and Efficiency: Deploying the models on Google Vertex AIshowcased the advantages of using managed cloud services for scaling machine learning applications. The platform facilitated efficient model training, deployment, and real-time inference, making it a valuable resource for future projects.

5. Potential Applications: The generated synthetic reviews have broad applications in various fields, including enhancing product recommendation systems, conducting sentiment analysis, and providing training data for machine learning models in scenarios with limited real-world data.

## Recommendations

Based on the findings and challenges faced during the project, the following recommendations are proposed for future work and similar projects:

1. Explore Additional Domains: Future projects should consider extending the methodology to other product categories or domains. This could involve generating synthetic data for areas such as electronics, clothing, or services, thereby broadening the applicability of LLMs in synthetic data generation.

2. Implement Feedback Loops: Incorporating user feedback mechanisms could enhance the quality of generated synthetic reviews. By allowing users to rate the relevance and accuracy of generated reviews, the model can be iteratively improved based on real-world assessments.

3. Enhance Grounding Techniques: Further research into grounding techniques beyond RAG could provide even richer contextual data for generated reviews. Exploring hybrid models that combine various retrieval methods or incorporating knowledge graphs may enhance the accuracy and relevance of synthetic outputs.

4. Utilize User-Generated Content: Future synthetic review generation projects should explore the potential of using real user-generated content to inform the models. This could involve scraping additional data from e-commerce platforms, forums, or social media to enrich the training dataset.

5. Focus on Ethical Considerations: As synthetic data generation technology evolves, addressing ethical concerns surrounding data use and generation becomes crucial. Future work should ensure transparency in synthetic data applications, especially in sensitive domains like healthcare or finance.

## Inference

In summary, the project has successfully established a robust methodology for generating synthetic product reviews using Large Language Models. The combination of RAG for grounding and strategic approaches to maintaining diversity has proven effective in creating high-quality synthetic data. The insights gained from this project pave the way for future exploration in the field of synthetic data generation, emphasizing the potential of LLMs in enhancing machine learning applications across various domains. As businesses and researchers continue to seek efficient solutions for data scarcity, the methodologies and recommendations outlined in this report will contribute to advancing the state of synthetic data generation and its applications in real-world scenarios.

# REPORT WORK

Use of **GRETEL** to get the correct CSV files as OUTPUT.

Present with the report the corrected CSV files which is fine tuned and also with 5000 corrected dataser as output that concludes the Assignment and with report there will be one csv file and Report Print

# Synthetic Data Quality Report

Gretel Sign In → (https://console.gretel.ai/login)

**Model**

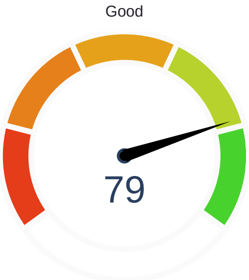
## tabular-lstm

**Model UID**

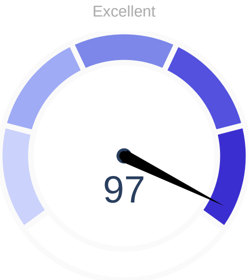
6701792e95b51a088c1fda31

**Project** GenSythData SHL

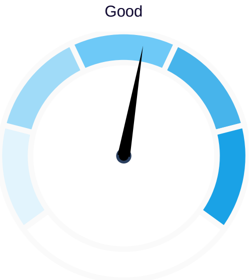
**Generated** 10/05/2024, 17:37



Synthetic Quality Score ?

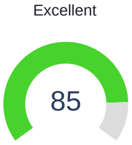


Data Privacy Score ?

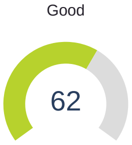


Privacy Configuration ?

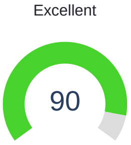
## Synthetic Quality Summary



?



?



?

| Field Correlation Stability | Deep Structure Stability | Field Distribution Stability |
|---|---|---|

| | Training Data | Synthetic Data |
|---|---:|---:|
| Row Count | 5000 | 5000 |
| Column Count | 6 | 6 |
| Training Lines Duplicated | -- | 0 |

What do these values mean?

## Privacy Configuration

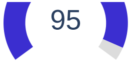
| Default Privacy Protections | | | Advanced Protections |
|---|---|---|---|
| Outlier Filter<br>Medium | Similarity Filter<br>Medium | Overfitting Prevention<br>Disabled | Differential Privacy<br>Disabled |

## Data Privacy Summary

Excellent

Excellent

Membership Inference Protection

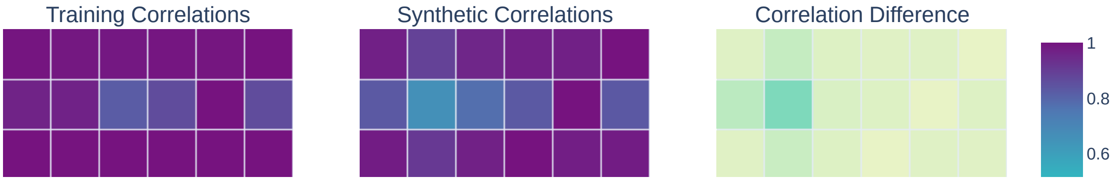


Attribute Inference Protection

## Training Field Overview

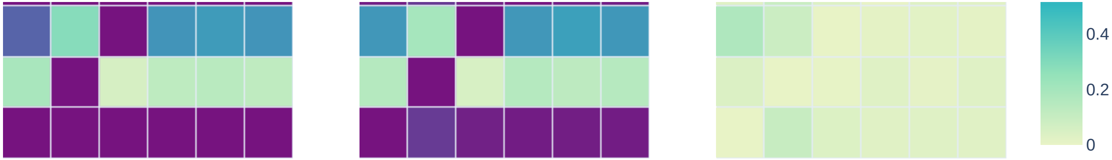
| Field | Unique | Missing | Ave. Length | Type | Distribution Stability |
|---|---:|---:|---:|---|---|
| timestamp | 5000 | 0 | 22.79 | Other | N/A |
| asin | 911 | 0 | 10.00 | Other | N/A |
| user_id | 4989 | 0 | 28.00 | Other | N/A |
| title | 3588 | 1 | 21.99 | Text | N/A |
| text | 4826 | 2 | 171.81 | Text | N/A |
| rating | 5 | 0 | 1.00 | Categorical | Moderate |

## Training and Synthetic Data Correlation

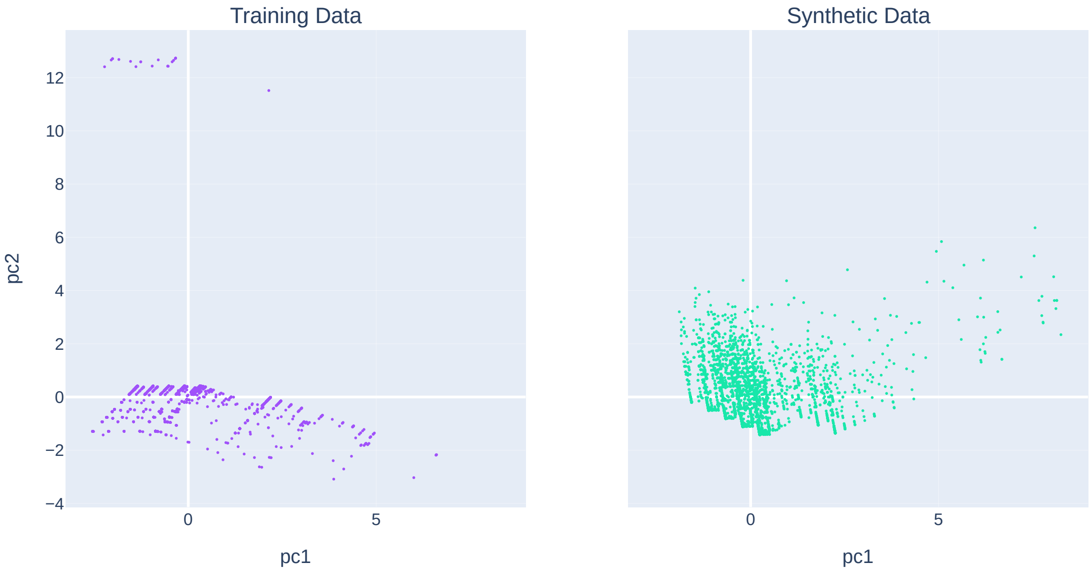

Training Correlations

Synthetic Correlations

Correlation Difference

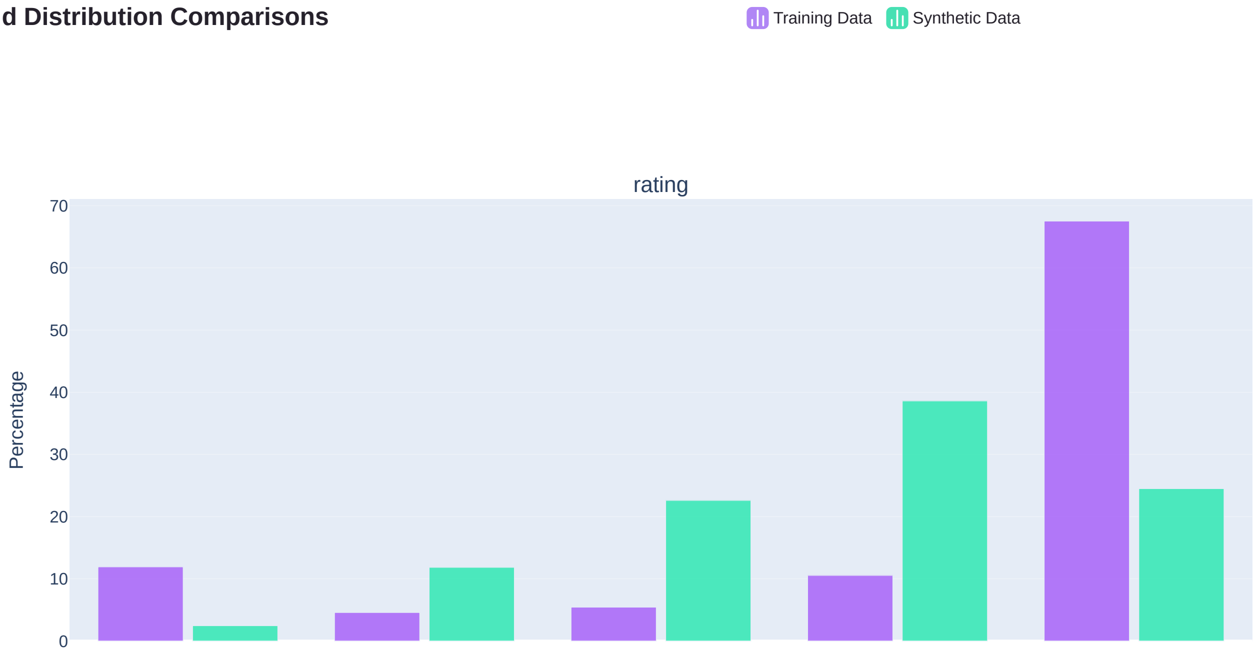

0.4
0.2
0

## Principal Component Analysis



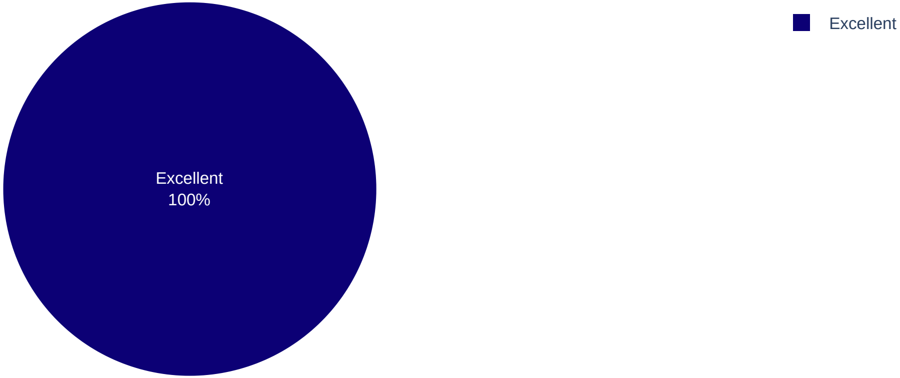
## Field Distribution Comparisons

()



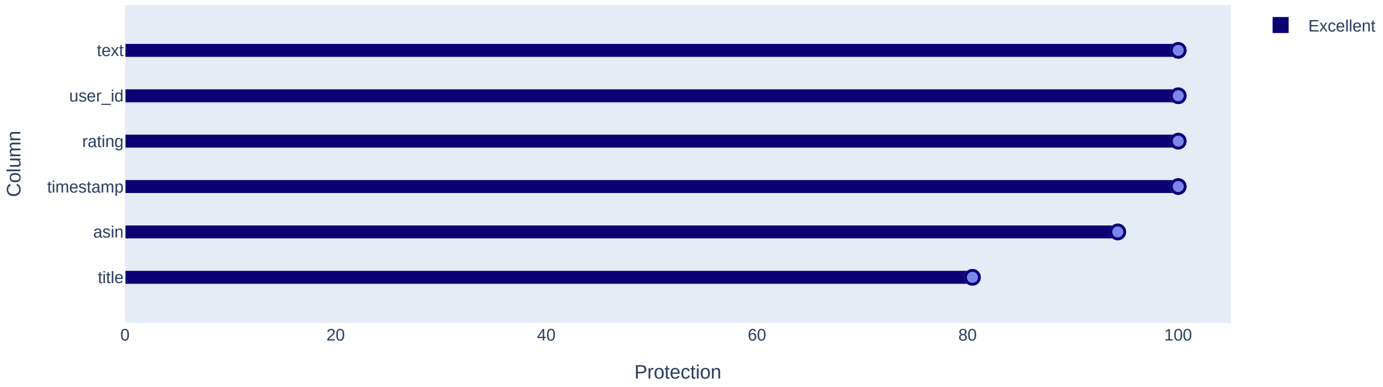
## Membership Inference Protection

Breakdown of protection level across 360 simulated attacks



## Attribute Inference Protection

Breakdown of protection across all columns


text
user_id
rating
timestamp
asin
title
Column
Protection
■ Excellent