Enhancing Chatbot Performance through Synthetic Data Augmentation

Undergraduate Research Thesis

Presented in partial fulfillment of the requirements for graduation with honors research distinction in Linguistics in the undergraduate colleges of The Ohio State University

by

Henry Leonardi

The Ohio State University

April 2023

Project Advisor: Professor Michael White, Department of Linguistics

Table of Contents

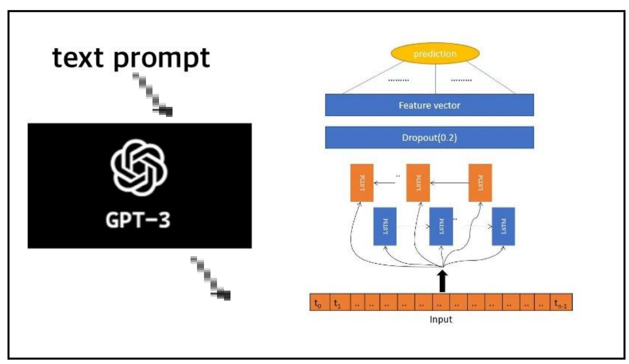
Abstract

Large language models like GPT-3 have shown remarkable generative capabilities in producing human-like text. However, their slow and costly nature limits their applicability in certain domains, such as specialized chatbot systems. In this paper, we investigate the use of GPT-3 for synthetic data augmentation to improve the performance of a recurrent neural network (RNN) classifier model employed in a chatbot for the Columbus museum COSI (Center of Science and Industry). The goal is to generate user questions that help train the model to classify user inputs under specific response labels, ultimately providing users with the best possible answers from a predefined answer dataset. We experimented with various prompting strategies, including providing GPT-3 with answers, real-world context, and few-shot examples. Our findings indicate that augmenting an existing dataset with GPT-3 generated data can increase the classifier model's accuracy, even without using a few-shot strategy. Furthermore, GPT-3 can effectively create datasets from scratch, although traditional user data still outperforms it. This demonstrates the potential to transfer knowledge from a large model like GPT-3 to a smaller, faster model for practical applications, such as the museum's virtual guide. Additionally, we explored validation strategies to retain only the relevant data, which further improved the results. This paper highlights the promising capability of using large language models like GPT-3 for synthetic data augmentation in improving the performance of lightweight models in real-world applications.

1. Introduction

Large language models (LLMs) such as GPT-3 (Brown et al., 2020) have shown remarkable generative capabilities in producing human-like text across various natural language tasks. However, their slow and costly nature limits their applicability in certain domains, such as chatbots that require fast and accurate responses to user inputs. One way to overcome this limitation is to use LLMs for synthetic data augmentation, i.e., generating additional data to train other models that are more suitable for the target domain. Wang et al. (2021) demonstrated that using GPT-3 as a low-cost data labeler can save 50% to 96% of labeling cost compared to using labels from humans, while achieving the same or better performance on the downstream models for a variety of natural language understanding (NLU) and natural language generation (NLG) tasks. They also proposed a novel framework that combines pseudo labels from GPT-3 with human labels, which further improves the performance with limited labeling budget.

Figure 1: An outline of this study's workflow



In this paper, we investigate the use of GPT-3 for synthetic data augmentation to improve the performance of an RNN classifier model employed in a chatbot for the COSI museum. The goal is to generate user questions that help train the model to classify user inputs under specific response labels, ultimately providing users with the best possible answers from a predefined answer dataset. We experiment with various prompting strategies, including providing GPT-3 with answers, real-world context, and few-shot approaches. Our goal is to show that augmenting an existing dataset with GPT-3 generated data can increase the classifier model's accuracy. Furthermore, we are interested in seeing if GPT-3 can effectively create datasets from scratch. We view this process as a way to transfer knowledge from a large model like GPT-3 to a smaller, faster model for practical applications, such as the museum's virtual guide.

2. Literature Review

Data augmentation is a technique that aims to generate additional data from existing data to improve the performance of machine learning models, especially in domains where data is scarce or expensive to obtain. Data augmentation can be applied to various types of data, such as images, audio, text, etc. In this paper, we focus on text data augmentation for chatbot systems that require fast and accurate responses to user inputs.

Text data augmentation can be broadly categorized into two types: rule-based and model-based. Rule-based methods rely on predefined rules or heuristics to manipulate the text, such as replacing words with synonyms, inserting or deleting words, swapping word order, etc. Model-based methods use machine learning models, such as neural networks, to generate new text based on the existing text, such as paraphrasing, summarizing, translating, etc.

Rule-based methods are simple and efficient, but they have some limitations. First, they may introduce noise or errors into the text, such as changing the meaning or breaking the grammar. Second, they may not capture the semantic and syntactic variations of natural language. Third, they may not be able to generate diverse and novel text that covers different aspects of the target domain.

Model-based methods can overcome some of these limitations by leveraging the power of large-scale pre-trained language models (LMs), such as GPT-3 (Brown et al., 2020). LMs are neural networks that are trained on massive amounts of text data to learn the statistical patterns and representations of natural language. LMs can generate fluent and coherent text that preserves the meaning and style of the original text. LMs can also capture the semantic and syntactic variations of natural language by using different prompting strategies, such as providing context, examples, keywords, etc.

However, model-based methods also have some challenges. First, they require access to large-scale pre-trained LMs, which are costly and slow to use. Second, they may generate irrelevant or inconsistent text that does not match the target domain or task. Third, they may suffer from exposure bias, which means that they tend to generate text that is similar to the existing text rather than exploring new possibilities.

To address these challenges, some researchers have proposed hybrid methods that combine rule-based and model-based methods to achieve a balance between efficiency and diversity. For example, Bird et al. (2021) proposed a framework that augments human-sourced data with artificial paraphrasing using T5 (Raffel et al., 2019), a state-of-the-art LM for text generation. They showed that their framework can improve the performance of various transformer-based text classification models for chatbot-like tasks.

Another line of research has focused on using synthetic data augmentation to transfer knowledge from large-scale LMs to smaller and faster models that are more suitable for the target domain or task. For example, Wang et al. (2021) demonstrated that using GPT-3 as a low-cost data labeler can save 50% to 96% of labeling cost compared to using labels from humans, while achieving the same or better performance on the downstream models for a variety of natural language understanding (NLU) and natural language generation (NLG) tasks. They also proposed a novel framework that combines pseudo labels from GPT-3 with human labels, which further improves the performance with limited labeling budget.

In this paper, we follow this direction and investigate the use of GPT-3 for synthetic data augmentation to improve the performance of an RNN classifier model employed in a chatbot for COSI. Our goal is to show that augmenting an existing dataset with GPT-3 generated data can increase the classifier model's accuracy. Furthermore, we are interested in seeing if GPT-3 can effectively create datasets from scratch. We view this process as a way to transfer knowledge from a large model like GPT-3 to a smaller, faster model like RNN.

3. Methods

3.1 Data Collection and Experimental Setup

The primary objective of this research is to enhance the performance of the chatbot employed at COSI, which uses a virtual guide to interact with and answer questions from visitors, particularly children. The chatbot functions by matching user queries to corresponding labels within its dataset, which are then used to provide appropriate responses. For instance, if a visitor asks about the location of the restroom, the chatbot must identify the "Where is the bathroom?" label and produce the pre-written response, such as "The bathroom is located to the left."

The chatbot was designed using BERT embeddings, training an RNN to associate these embeddings with the appropriate labels. However, the chatbot's accuracy was low, prompting the decision to look for ways to improve its performance. The most obvious way to address this issue was to collect user data over time, with each interaction between chatbot and visitor recorded alongside the chatbot's provided response. Subsequently, annotators would evaluate the correctness of the response and, if incorrect, assign the most suitable label from the dataset of 120 labels.

This process generated a dataset comprising more than 1,000 labeled user interactions, which was then employed to train the chatbot model. As a result, the chatbot achieved a label-matching accuracy of approximately 58%, significantly surpassing random chance and proving to be a valuable asset for users.

Despite these improvements, the limited dataset size of around 1,000 labeled interactions, combined with the time-consuming and labor-intensive annotation process, prompted the need for a more efficient method of dataset expansion. The primary goal of the experiment discussed in this paper is to identify strategies for increasing the dataset size without relying on the continuous collection of user data and subsequent manual annotations.

In the following sections, we will detail the specific methods and techniques employed to achieve this objective, with a focus on leveraging the existing user interaction dataset to further improve the chatbot's accuracy and F1 score.

3.2 Data Augmentation and Prompt Engineering

In this section, we detail the methods used for data augmentation with GPT-3 to generate synthetic examples for training the chatbot model. The primary goal of data augmentation is to increase the diversity and coverage of the training data, thereby improving the chatbot's capability to recognize and respond to a wide range of user questions.

We used GPT-3 to generate synthetic examples by providing it with the 120 answers associated with the question labels. For instance, given the answer, "COSI was founded in 1964" to the question label, "When was COSI founded?", GPT-3 was prompted to generate 10 synthetic user questions that could be satisfied by the given answer. We experimented with three different prompting strategies to explore their impact on the quality and diversity of the generated data: Simple Prompting, Informal Prompting, and Few-Shot Prompting.

3.2.1 Simple Prompting Strategy

In the Simple Prompting strategy, GPT-3 was instructed to generate 10 examples of questions that a user might ask, which could be answered by the given answer (e.g., "COSI was founded in 1964"). This strategy aimed to create a more diverse set of user questions for the chatbot model to learn from.

3.2.2 Informal Prompting Strategy

The Informal Prompting strategy aimed to generate examples that better reflect the casual interactions users might have with the chatbot. As many children interact with the chatbot, we instructed GPT-3 to generate questions as if asked by a seven-year-old. This approach aimed to create a more informal and diverse set of questions, better suited to real-world use.

3.2.3 Few-Shot Prompting Strategy

The Few-Shot Prompting strategy involved providing GPT-3 with some of the user data collected and described in the first section of the methods section, including annotated user questions matched to their respective labels. GPT-3 was also given the associated answer and was asked to generate examples similar to the provided user questions. This strategy aimed to better mimic the style of real user questions and enhance the quality of the generated data.

3.2.4 Dataset Generation and Comparison

The synthetic examples generated by GPT-3 were formatted and tokenized before being added to the training dataset. We created three different datasets by augmenting the original user dataset with data from each prompting strategy: Simple, Informal, and Few-Shot. Additionally, we created datasets containing only the augmented data to compare their performance with the combined datasets. In total, six datasets were generated for training and evaluation purposes, allowing us to assess the effectiveness of each prompting strategy in enhancing the chatbot's performance.

3.3 Validation Strategies for Synthetic Data Filtering

In this section, we discuss the validation strategies we employed to filter out low-quality synthetic data points generated by GPT-3. Our approach involved using GPT-3 as a judge to assess the quality of the generated queries. To achieve this, we provided GPT-3 with one of the questions generated during the data generation process and the top five most semantically similar answers from the dictionary of 120 question-answer pairs. Almost always, one of these labels would be the actual answer, given the semantic similarity between the question and answer labels.

We asked GPT-3 to select the most appropriate answer label from the top five options. If the selected answer label matched the one used to generate the question, we considered the synthetic data point to be of high quality, as it would likely be matched to the correct answer in a chatbot interaction setting. This validation process allowed us to remove a significant amount of generated questions that would not have matched the best answer.

To measure how closely the meaning of a question matches the meaning of an answer, we used sentence transformers (Reimers et al. 2019) and the open-source "all-mpnet-base-v2" model. However, this method has a drawback: it may limit the variety of questions we can generate, because GPT-3 may not consider questions that are very different in meaning from their answers.

Due to the high computational cost associated with making numerous requests to GPT-3, we only tested this validation strategy on two datasets. The first dataset consisted of synthetic data generated using the simple prompting strategy, and the second dataset was generated using the highest-performing model to that point, the informal prompting strategy, which asked GPT-3 to generate queries as if it were a 7-year-old.

3.4 RNN Model

RNNs are a class of neural networks that can handle sequential data, such as natural language sentences. RNNs process the input sequence element by element and maintain an internal memory that stores the information of the previous elements. RNNs can learn to capture the long-term dependencies and the semantic and syntactic structures of natural language (Goodfellow et al., 2016).

However, standard RNNs have some limitations, such as the difficulty of accessing the information from distant elements and the vanishing or the exploding gradient problem. To overcome these limitations, various variants of RNNs have been proposed, such as long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997) and gated recurrent unit (GRU) (Cho et al., 2014). These variants introduce gating mechanisms that control the information flow and the memory update in the RNN unit.

Another improvement of RNNs is to use bidirectional architectures, which process the input sequence from both directions and concatenate the outputs from the forward and backward directions. Bidirectional RNNs can capture the context from both sides of each element and thus enhance the representation power of RNNs (Schuster and Paliwal, 1997). In this paper, we use a bidirectional GRU block as the first component of our neural network architecture for classifying user questions into one of 120 answer labels. This was based on an implementation by Adam Stiff of a self attentive RNN described in Lin et al. (2017).

3.5 Evaluation

To evaluate the effects of the prompting strategies, we employed a k-fold cross-validation method with five folds to assess the performance of the models. The user data collected at the COSI served as the foundation for this evaluation.

In this k-fold cross-validation process, the data was partitioned into five random subsets, with each subset undergoing a train-test split. The models' performance was measured using the average accuracies and F1 scores from the five subsets. This evaluation technique aimed to address the limited user data available and provide a more robust understanding of the models' effectiveness.

For each fold, we expanded the training data using the augmentation strategies discussed in the previous section. It is crucial to emphasize that although the augmented data was present in every fold, the specific training data within each fold varied due to the random selection method.

The test data employed for evaluation exclusively contained the user data from the test portion of each fold, ensuring that the assessment focused on the models' performance with genuine user interactions. This approach was deliberately designed to tackle the challenge of scarce user data and offer a comprehensive evaluation of the models trained on both synthetic and augmented datasets.

4. <u>Results</u>

4.1 Augmentation Results

In this section, we present the results of our study comparing the performance of models trained on datasets augmented with three different prompt engineering strategies: simple prompting, informal prompting, and few-shot prompting. We evaluated three models using five-fold cross-validation and reported the average accuracy and F1 scores.

As a baseline, we first trained a model on user data alone, which yielded an accuracy of 0.589 and an F1 score of 0.444. We then proceeded to assess the impact of the three prompting strategies on model performance.

1. Simple Prompting Strategy: The model trained on the dataset augmented with the simple prompting strategy exhibited an accuracy of 0.608, an increase of approximately one percentage point compared to the baseline. The F1 score improved by a few points, reaching 0.468.

2. Informal Prompting Strategy: The model trained on the dataset augmented with the informal prompting strategy achieved an accuracy of 0.625, a more substantial increase compared to the baseline. The F1 score also improved to 0.484. This strategy appeared to be the most effective of the three.

3. Few-shot Prompting Strategy: The model trained on the dataset augmented with the few-shot prompting strategy demonstrated an accuracy of 0.600 and an F1 score of 0.456.

Strategy Employed	Average
Baseline Accuracy	0.589
Baseline F1	0.444
Simple Augmented Accuracy	0.608
Simple Augmented F1	0.468
Informal Augmented Accuracy	**0.625**
Informal Augmented F1	**0.484**
Few-shot Augmented Accuracy	0.600
Few-shot Augmented F1	0.456

Figure 2: Accuracy and F1 of models trained on different augmented datasets

In summary, the prompt engineering strategies did not have a significant impact on model performance, with the exception of the informal prompting strategy, which resulted in modest improvements in both accuracy and F1 score. The gains of approximately three percentage points in accuracy and over four in F1 score are noteworthy, suggesting that the informal prompting strategy could be beneficial for enhancing model performance.

4.2 Solo Models

In the second part of the results section, we shift our focus to the performance of models trained exclusively on synthetic data. Three purely synthetic datasets were generated using each of the three prompting strategies: simple, informal, and few-shot. The models were then trained on these datasets, and their performances were compared to the baseline model that was trained on non-augmented data.

For the simple prompting strategy, the model achieved an accuracy of 0.447 and an F1 score of 0.322. Remarkably, this corresponds to 75.9% of the baseline accuracy and 72.5% of the baseline F1 score, despite not being exposed to any user data during training.

The informal prompting strategy resulted in even better performance, with the model attaining an accuracy of 0.482 and an F1 score of 0.352. In this case, the model was able to recover 81.8% of the baseline accuracy and 79.7% of the baseline F1 score, all without any access to user data.

Finally, the few-shot strategy, which did utilize some user data to generate examples, yielded an accuracy of 0.457 and an F1 score of 0.334. This corresponds to 76.2% of the baseline accuracy and 73.2% of the baseline F1 score.

It is important to note that none of these strategies outperformed the baseline model in terms of accuracy or F1 score. However, their relative success in recovering a significant proportion of the baseline performance is noteworthy, particularly for the simple and informal strategies that did not rely on any user data. The informal strategy, which achieved the highest scores among the three prompting strategies, only had access to the answer labels and was able to create a dataset for model training based solely on the basic facts required for answering the prompts. This demonstrates the potential of synthetic data generation for reducing the reliance on user data while still achieving reasonable model performance.

Strategy Employed	Average	Ratio to Baseline
Simple Solo Acc.	0.447	0.759
Simple Solo F1	0.322	0.725
Informal Solo Acc.	**0.482**	**0.818**
Informal Solo F1	**0.354**	**0.797**
Few-shot Solo Acc.	0.457	0.762
Few-shot Solo F1	0.334	0.732

Figure 3: Accuracy and F1 of models trained only on synthetic data

4.3 Validation Strategy Results

In this section, we discuss the results obtained after applying our validation strategy on two of the datasets: simple prompting strategy and informal prompting strategy. Our validation strategy involved using GPT-3 as a judge to identify and remove less relevant generated queries in the synthetic data. Subsequently, we retrained the models on the filtered segmented datasets, consisting of both baseline user data and the remaining synthetic data.

For the simple prompting strategy, the initial model achieved an accuracy of 60.8% and an F1 score of 0.468. After applying the validation strategy, the performance considerably improved to 63.1% accuracy and an F1 score of 0.492. The improvement suggests that the removal of less relevant paraphrases allowed the model to learn better from the synthetic data.

On the other hand, the informal prompting strategy initially resulted in the highest accuracy among the augmented datasets. However, after applying the validation strategy, the accuracy slightly decreased to 62.1% and the F1 score reduced to 0.473. This result indicates that the informal prompting strategy generated a substantial amount of useful data, and the removal of any data points did not improve the model's performance but rather caused a decline.

In summary, the validation strategy led to our highest accuracy model when applied to the simple prompting strategy dataset. The removal of less relevant generated queries allowed the model to learn more effectively from the synthetic data. However, the same approach did not yield improvement for the informal prompting strategy dataset, suggesting that the quality of synthetic data generated using different prompting strategies can significantly impact the effectiveness of the validation strategy.

Strategy Employed	Average
Baseline Accuracy	0.589
Baseline F1	0.444
Simple Augmented Accuracy	0.608
Simple Augmented F1	0.468
Informal Augmented Accuracy	0.625
Informal Augmented F1	0.484
Simple Augmented + Filter Accuracy	**0.631**
Simple Augmented + Filter F1	**0.492**
Informal Augmented + Filter Accuracy	0.621
Informal Augmented + Filter F1	0.473

Figure 3: Accuracy and F1 of models trained only on synthetic data

5. Conclusion

5.1 Summary of Findings

In the first part of our conclusion, we summarize the key findings of our study on synthetic data augmentation using GPT-3 for chatbot systems. Our results demonstrated that we were able to successfully augment user data by generating synthetic examples through the implementation of various prompting strategies. Among these strategies, the most effective appeared to be the informal prompting approach, which involved simulating how users, particularly children, would interact with our system.

In addition to the augmentation of existing datasets, our findings also revealed the potential for creating new datasets without relying on collected user data. Although the accuracy of such datasets was relatively low, achieving approximately 50% accuracy for a 120-label classification task was a promising outcome, particularly considering the inherent challenges of the task and the prompting strategies employed.

Furthermore, we discovered that GPT-3 can be utilized both as a generator and a filter of data. This dual functionality proved beneficial in improving the accuracy of our RNN classifier for classification tasks, which served as a model for our chatbot scenario. GPT-3 demonstrated the ability to augment existing datasets and perform reasonably well even in the absence of user data, making it suitable for handling sparse or no data situations, provided that the labels are available.

While the improvements in accuracy did not meet our initial expectations, the overall results were encouraging, especially given GPT-3's performance in the absence of user data. This highlights the potential of synthetic data augmentation using GPT-3 for chatbot systems and similar applications.

5.2 Limitations and Future Research

In the second part of the conclusion, we discuss the limitations of our study and propose avenues for future research. One primary limitation is that our data augmentation strategy using GPT-3 struggled to generate data that perfectly simulated users' grammar and nuanced questions, particularly due to the limited context provided about the specific tasks. For example, we did not supply GPT-3 with adequate information about the museum center where the chatbot was to be deployed, which in turn affected its ability to accurately model user interactions. Providing more context would have increased the cost of each request, a trade-off we opted not to make in this experiment.

However, we believe that newer models with larger context windows could potentially address this issue by taking in more information about the users they are attempting to simulate. The success of prompting GPT-3 to simulate a seven-year-old highlights this potential improvement. Additionally, as the cost of using models like GPT-3 decreases, we are intrigued by the prospect of utilizing them as choosers or judges for various classification tasks. The recent release of the more affordable GPT-3.5 Turbo model presents a valuable opportunity to explore its capabilities for annotation and curation tasks, building on our experience with query validation in this study.

In summary, while we acknowledge the limitations of our current approach, we are optimistic about the potential for advancements in AI technology to address these challenges and further improve data augmentation and validation strategies. Future research should focus on overcoming the limitations identified in this study and explore the potential of newer, more advanced models to enhance chatbot data sets and improve user interaction simulations.

6. References

1. Bird J.J., Ekárt A., Faria D.R., Chatbot Interaction with Artificial Intelligence: human data augmentation with T5 and language transformer ensemble for text classification. Journal of Ambient Intelligence and Humanized Computing (2021). https://doi.org/10.1007/s12652-021-03439-8
2. Brown T.B., Mann B., Ryder N., Subbiah M., Kaplan J., Dhariwal P., Neelakantan A., Shyam P., Sastry G., Askell A., Agarwal S., Herbert-Voss A., Krueger G., Henighan T., Child R., Ramesh A., Ziegler D.M., Wu J., Winter S., Hesse C., Chen K., Sigler E., Litwin M., Gray S., Chess B., Clark J., Berner C., McCandlish S., Radford A., Sutskever I. and Amodei D. Language Models are Few-Shot Learners. Advances in Neural Information Processing Systems 33 (2020).
3. Cho K., Van Merriënboer B., Gulcehre C., Bahdanau D., Bougares F., Schwenk H. and Bengio Y. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) (2014) 1724-1734.
4. Goodfellow I., Bengio Y. and Courville A. Deep Learning. MIT Press (2016).
5. Hochreiter S. and Schmidhuber J. Long Short-Term Memory. Neural Computation 9 (1997) 1735-1780.
6. Lin Z., Feng M., dos Santos C.N., Yu M., Xiang B., Zhou B. and Bengio Y. A Structured Self-attentive Sentence Embedding. arXiv preprint arXiv:1703.03130 (2017).
7. Raffel C., Shazeer N., Roberts A., Lee K., Narang S., Matena M., Zhou Y., Li W. and Liu P.J. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. Journal of Machine Learning Research 21 (2020) 1-67.

8. Reimers N. and Gurevych I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (2019) 3982-3992.
9. Schuster M. and Paliwal K.K. Bidirectional Recurrent Neural Networks. IEEE Transactions on Signal Processing 45 (1997) 2673-2681.
10. Wang Y.-C, Hsu W.-T, Chen Y.-N, Kao H.-Y and Ku L.-W. GPT Understands, Too. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (2021)