Abstract

Data augmentation alleviates the problem of data scarcity when training language models (LMs) by generating new examples based on the existing data. A successful approach to generate new samples is to fine-tune a pre-trained LM on the task-specific data and then sample from the label-conditioned LM. However, fine-tuning can be difficult when task-specific data is scarce. In this work, we explore whether large pretrained LMs can be used to generate new useful samples without fine-tuning. For a given class, we propose concatenating few examples and prompt them to GPT-3 to generate new examples. We evaluate this method for few-shot intent classification on CLINC150 and SNIPS and find that data generated by GPT-3 greatly improves the performance of the intent classifiers. Importantly, we find that, without any LM fine-tuning, the gains brought by data augmentation with GPT-3 are similar to those reported in prior work on LM-based data augmentation. Experiments with models of different sizes show that larger LMs generate higher quality samples that yield higher accuracy gains.

1 Introduction

A key challenge in creating task-oriented conversational agents is gathering and labeling training data. The realistic training data resulting from actual human interaction with the agent does not exist until the system is launched. Prior to the launch, data gathering options include manual authoring and crowd-sourcing. Both of these options are tedious and expensive. Data augmentation methods aim to alleviate the data acquisition issue by automatically generating more examples based on available ones.

A particularly promising trend in recent research on data augmentation for natural language processing is using large pretrained language models (LMs) (Peters et al., 2018; Devlin et al., 2018) for this purpose. The general paradigm of these approaches is to fine-tune a LM on the task-specific data and then to generate new examples from a label-conditioned LM (Wu et al., 2018; Kumar et al., 2019, 2021; Anaby-Tavor et al., 2020; Lee et al., 2021). A potential issue with this approach is that when the task-specific data is scarce, fine-tuning a large LM on very few available examples can become the bottleneck.

In this work we investigate if large LMs can be used for generating new examples without any task-specific fine-tuning. We are inspired by GPT-3 results (Brown et al., 2020), which show that giant LMs can perform text classification when prompted with just a few examples. To use GPT-3 for data generation instead, we apply the example extrapolation approach by Lee et al. (2021). Namely, to create the prompt we concatenate several example utterances for a class (see Figure 1). Crucially, unlike Lee et al. (2021) we do not fine-tune the LM, and instead investigate if useful additional data can be generated by an off-the-shelf pretrained LM.

We focus this study on the task of predicting the user’s intent, i.e. what the user of the task-oriented chatbot wants to accomplish. We show that in few-shot intent classification setups

Figure 1: Method. We first generate new samples by prompting GPT-3 with a few examples from each class. Then we fine-tune a BERT classifier with the augmented dataset composed of the initial samples plus the GPT-3 generated samples.
based on CLINC150 (Larson et al., 2019) and SNIPS (Coucke et al., 2018) datasets, the data generated by GPT-3 greatly improves performance of an intent classifier. Importantly, the improvements that our data generation method brings are comparable to those reported by Lee et al. (2021) who fine-tune a T5 model on held-out classes with many examples per class. Our intrinsic evaluation (Kumar et al., 2021) with an oracle classifier, shows that when given enough examples in the prompt, GPT-3 indeed latches onto the examples’ intent and reliably generates more examples of the same intent. Our experiments with models of different sizes show that the LM’s data generation abilities get better as the LM gets bigger.

2 Method

We consider the task of training an intent classifier. An intent is a type of request that the conversational agent supports; e.g. the user may want to change the language of the conversation, play a song, transfer money between accounts, etc. Collecting many example utterances that express the same intent can be difficult and expensive. In this paper, we experiment with an extremely simple method to augment the training data available for an intent. Namely, as shown in Figure 1, we select \( K \) examples for an intent, concatenate them with newlines and feed the resulting string as a prompt to a large generic LM, such as e.g. GPT-3 (Brown et al., 2020).

3 Experimental Setup

3.1 Datasets

We use CLINC150 (Larson et al., 2019) and SNIPS (Coucke et al., 2018) intent classification datasets in our experiments. CLINC150 has 23,700 example utterances, out of which, 1200 utterances belong to a special out-of-scope (OOS) class. The rest of the dataset covers 10 domains each consisting of 15 distinct intents. The dataset is balanced, with 100 training examples per intent, 20 (100 OOS) for validation, and 30 (1000 OOS) for testing. SNIPS is a dataset collected from the Snips personal voice assistant. The training set contains 13,084 utterances, whereas, the test and validation sets contain 700 utterances each. These utterances cover 7 different intents.

3.2 Setup

We simulate data sparsity in two ways. First, to produce comparable results to the example extrapolation approach (Ex2, Lee et al. (2021)), we consider the partial few-shot setting. In this setup, for a few-shot subset of intents, we truncate training data to \( K \) examples per intent.\(^1\) For CLINC150, we use different domains as the few-shot subsets, whereas for SNIPS, we use the different intents. Second, similar to Vulić et al. (2021), we experiment with a more challenging full few-shot setting, in which we only keep \( K \) examples per intent for all the intents. When data augmentation is performed, we augment the few-shot intents to have \( N \) examples, where \( N \) is the median number of examples per intent of the original data.

To precisely describe the training and test data in all settings we will use \( D_{\text{part}} \) to refer to dataset parts, i.e. train, validation, and test. We use \( D_F \) for few-shot data and \( D_M \) to refer data-rich intents (in the partial few-shot setting). This notation is defined for all parts, therefore, \( D_{\text{part}} = D_{\{F,\text{part}\}} \cup D_{\{M,\text{part}\}} \). \(^2\) When GPT-3 is used to augment the training data we generate \( N - K \) examples per intent and refer to the resulting data as \( D_{F,\text{train}} \). We experiment with four different-sized GPT-3 models\(^2\) to obtain \( D \): Ada, Babbage, Curie, and Davinci. In order, Ada is the smallest model (nearly 350M parameters) and Davinci is the largest (near 175B parameters).\(^3\)

3.3 Training and Evaluation

We fine-tune BERT-large (Devlin et al., 2018) on the task of intent classification by adding a linear layer on top of the [CLS] token (Wolf et al., 2019).

Partial few-shot. In this setup, we train \( S \) intent classifiers, choosing a different few-shot subset of intents every time to obtain \( D_F \). We then average the metrics across these \( S \) runs. For CLINC150, \( S = 10 \) corresponding to the 10 different domains, whereas for SNIPS, \( S = 7 \) corresponding to the 7 different intents. We evaluate our method on the following three scenarios introduced by Lee et al. (2021): (i) Baseline: models are trained without data augmentation on \( D_{\{F,\text{train}\}} \cup D_{\{M,\text{train}\}} \). (ii) Upsampled: \( D_{\{F,\text{train}\}} \) is upsampled to have \( N \) examples per intent. Then models are trained on upsampled

\(^1\)We use the truncation heuristic provided by Lee et al. (2021): https://github.com/google/example_extrapolation/blob/master/preprocess_clinc150.py

\(^2\)https://beta.openai.com/docs/engines

\(^3\)https://blog.eleuther.ai/gpt3-model-sizes/
\[ D_{\{F,\text{train}\}} \cup D_{\{M,\text{train}\}}. \] (iii) Augmented: models are trained on \( D_{\{F,\text{train}\}} \cup D_{\{F,\text{train}\}} \cup D_{\{M,\text{train}\}}. \)

For each scenario, we report the 1) overall in-scope accuracy on the complete test set \( D_{\text{test}} \), i.e. intent classification accuracy excluding OOS samples in the test set, 2) out-of-scope recall (OOS recall) on \( D_{\text{test}} \) that we compute as percentage of OOS examples that the model correctly labeled as such, and 3) few-shot classification accuracy of the models on \( D_{\{F,\text{test}\}}. \) We also train an oracle \( \mathcal{O} \) on \( D_{\text{train}} \cup D_{\text{test}} \) and use it to measure the fidelity of samples generated by GPT-3. We define fidelity as the accuracy of the oracle classifier on the generated samples. A higher value denotes that the generated samples are more faithful to original data distribution.

Full few-shot. In this setup, we treat all the intents as few-shot and evaluate our method on the following three scenarios: (i) Baseline: all the intents are truncated to \( K = 10 \) samples per intent. (ii) Augmented: models are trained on \( D_{\{F,\text{train}\}} \cup D_{\{M,\text{train}\}}. \) (iii) Augmented+Relabeled: we use the oracle \( \mathcal{O} \) to relabel the generated samples from GPT-3 and then train the intent classifier on relabeled \( D_{\{F,\text{train}\}} \cup D_{\{M,\text{train}\}}. \) The purpose of this experiment is to estimate what further gains can be achieved if the data generated by GPT-3 were labeled by the human.

For each scenario in this setup, we report (i) the overall in-scope classification accuracy, and (ii) out-of-scope recall. For both partial few-shot and full few-shot settings we report means and standard deviations over 10 repetitions of each experiment.

4 Experimental Results

Table 1 shows the results of our experiments on the CLINC150 and SNIPS datasets. By augmenting the dataset with GPT-3 generated samples, the few-shot accuracy improves by up to 2.7% on CLINC150 and 18.12% on SNIPS when compared to the baseline setting. We observe that using larger GPT-3 models improves the classifier performance across all metrics. Specifically, we show that the examples generated by davinci, the largest GPT-3 model, bring the largest boost to both few-shot and overall accuracies (this conclusion is significant at \( p < 0.01 \) level according to the standard T-test). Finally, our method achieves competitive results compared to Ex2 (Lee et al., 2021), both in terms of absolute accuracies and the relative gains brought by data augmentation. Note that Ex2 uses T5-XL (Roberts et al., 2020) with nearly 3 billion parameters as its base intent classifier, while our method uses BERT-large with only 340 million parameters.

Table 2 shows results of our full few-shot experiments. Data augmentation by GPT-3 is also helpful in this scenario, especially when larger engines are used. The Ex2 method is not applicable because there is no data-rich intents to train the example extrapolator on. We compare instead to (Vulić et al., 2021) who consider a similar full few-shot setting with \( K = 10 \) examples and report similar results. Notably, augmenting the training data with Davinci brings an improvement of the same magnitude as that achieved by Vulić et al. (2021), who change the classification approach. Lastly, relabeling the
Table 2: Full few-shot results on CLINC150.

Table 3: Qualitative results. For each class we show some input samples and we cherry picked some generated samples. Green samples are considered good ones and red ones are considered bad ones.

Table 1: Partial few-shot results on CLINC150 and SNIPS datasets. Refer to Section 3.3 for more details.

4.1 Analysis

Figure 2 shows how validation few-shot accuracies and OOS recall vary when different generation temperatures are used for GPT-3. We observe that for all engines the generated data is most helpful with temperature around 1.0, although lower temperatures result in higher OOS recall. We also observe that the fidelity of the generated samples decreases as we increase the temperature (i.e. higher diversity, see Figure 2c). This suggests better fidelity does not always imply better quality samples as the language model may produce less diverse utterances at lower temperatures. In Appendix A, we perform a human evaluation, reaching similar conclusions as when using an oracle to approximate fidelity.

We refer the reader to Table 3 for examples of generated utterances.

5 Conclusion

We propose to prompt large pretrained language models to perform data augmentation in the few-shot intent classification regime. Experiments on CLINC150 and SNIPS show that GPT-3-generated examples significantly improve the performance of few-shot intent classifiers without finetuning and that larger models produce more useful additional data. Our oracle experiments suggest that further gains can be achieved if the generated data were labeled by a human. In future work we will experiment with data generation by other language models, as well as with approaches to identify the generated examples that require human relabeling.
References


Appendix

A Human Evaluation

In Figure 2 we evaluate the fidelity of the samples generated by GPT-3 with respect to the original set of sentences used to prompt it. Fidelity is approximated by the classification performance of an "oracle" intent classifier trained on the whole dataset \((D_{\text{train}} \cup D_{\text{test}})\) and evaluated over the generated samples. In order to assess whether the oracle predictions are comparable to those of a human, we perform a human evaluation study.

![Figure 3: Human evaluation. Error rate of human evaluators at the task of finding whether any sentence in a group of 5 was generated by GPT-3 or not. Each color represents a different GPT-3 engine. Higher error rate indicates that humans could not correctly identify generated samples and thus it also indicates higher fidelity. The standard error is displayed as a vertical line on top of each bar.](image)

We consider that a model produces sentences with high fidelity if a human is unable to distinguish them from a set of human-generated sentences belonging to the same intent. Therefore, for each intent in the CLINC150 dataset, we sample five random examples and we randomly choose whether to replace one of them by a GPT-3 generated sentence from the same intent. We generate sentences with each of the four GPT-3 models considered in the main text with two different temperatures (0.8 and 1.0). The sentence to replace is randomly selected. Finally, the five sentences are displayed to a human who has to choose which of the sentences is generated by GPT-3, if any.

The task is presented to human evaluators in the form of a web application (see Figure 4). We placed a button next to each sentence in order to force human evaluators to individually consider each of the examples. Once annotated, the evaluator can either submit, discard, or leave the task to label later. We used a set of 15 voluntary evaluators from multiple backgrounds, nationalities, and genders. Each evaluator annotated an average of 35 examples, reaching a total of 500 evaluated tasks.

For each model and temperature, we report the error rate of humans evaluating whether a task contains a GPT-generated sample. We consider that evaluators succeed at a given task when they correctly find the sentence that was generated by GPT or when they identify that none of them was generated. Thus, the error rate for a given model and temperature is calculated as \#failed / total_evaluated.

Results are displayed in Figure 3. We find that human evaluators tend to make less mistakes when the temperature used to sample sentences from GPT-3 is smaller. This result is expected since lowering the temperature results in sentences closer to those prompted to GPT-3, which are human-made. We also observe that models with higher capacity such as Davinci tend to generate more indistinguishable sentences than lower-capacity models such as Ada, even for higher temperatures. These results are in agreement with the "oracle" fidelity results introduced in Figure 2.

![Figure 4: Human evaluation tool. Example of a question for the human evaluators. Human evaluators are asked to flag which example is GPT-3 generated if any among the 5 presented ones.](image)