

# ChatAug: Leveraging ChatGPT for Text Data Augmentation

Haixing Dai\*, Zhengliang Liu\*, Wenxiong Liao\*, Xiaoke Huang, Zihao Wu, Lin Zhao, Wei Liu, Ninghao Liu, Sheng Li, Dajiang Zhu, Hongmin Cai, Quanzheng Li, Dinggang Shen, Tianming Liu, and Xiang Li

**Abstract**—Text data augmentation is an effective strategy for overcoming the challenge of limited sample sizes in many natural language processing (NLP) tasks. This challenge is especially prominent in the few-shot learning scenario, where the data in the target domain is generally much scarcer and of lowered quality. A natural and widely-used strategy to mitigate such challenges is to perform data augmentation on the training data to better capture the data invariance and increase the sample size. However, current text data augmentation methods either can not ensure the correct labeling of the generated data (lacking faithfulness) or can not ensure sufficient diversity in the generated data (lacking completeness), or both. Inspired by the recent success of large language models, especially the development of ChatGPT, which demonstrated improved language comprehension abilities, in this work, we propose a text data augmentation approach based on ChatGPT (named ChatAug). ChatGPT is trained on data with unparalleled linguistic richness and employs a reinforcement training process with large-scale human feedback, which endows the model with affinity to the naturalness of human language. Our text data augmentation approach ChatAug rephrases each sentence in the training samples into multiple conceptually similar but semantically different samples. The augmented samples can then be used in downstream model training. Experiment results on few-shot learning text classification tasks show the superior performance of the proposed ChatAug approach over state-of-the-art text data augmentation methods in terms of testing accuracy and distribution of the augmented samples.

**Index Terms**—Large language model, few-shot learning, nature language processing, data augmentation.

## 1 INTRODUCTION

THE effectiveness of natural language processing (NLP) heavily relies on the quality and quantity of the training data. With limited training data available, which is a common issue in practice due to privacy concerns or the high cost of human annotation, it can be challenging to train an accurate NLP model that generalizes well to unseen samples. The challenge of training data insufficiency is especially prominent in few-shot learning (FSL) scenarios, where the model trained on the original (source) domain data is expected to generalize from only a few examples in the new (target) domain [1]. Many FSL methods have

shown promising results in overcoming this challenge in various tasks [2]. Existing FSL methods mainly focus on improving the learning and generalization capability of the model via better architectural design [3], [4], [5], leveraging pre-trained language models as the basis and then fine-tuning it using limited samples [6] with meta-learning [4], [7] or prompt-based methods [8], [9], [10], [11]. However, the performance of these methods is still intrinsically limited by the data quality and quantity in both the source and target domains.

Besides model development, text data augmentation can also overcome the sample size limit and work together with other FSL methods in NLP [12], [13]. Data augmentation is usually model-agnostic and involves no change to the underlying model architecture, which makes this approach particularly practical and applicable to a wide range of tasks. In NLP, there are several types of data augmentation methods. Traditional text-level data augmentation methods rely on direct operations on the existing sample base. Some frequently used techniques include synonym replacement, random deletion, and random insertion [14]. More recent methods utilize language models to generate reliable samples for more effective data augmentation, including back-translation [15] and word vector interpolation in the latent space [16]. However, existing data augmentation methods are limited in the accuracy and diversity of the generated text data, and human annotation is still mandatory in many application scenarios [14], [17], [18].

The advent of (very) large language models (LLMs) such as the GPT family [8], [19] brings new opportunities for generating text samples that resemble human-labeled data, which significantly alleviates the burden of human anno-

- \* Co-first authors.
- Haixing Dai, Zhengliang Liu, Zihao Wu, Lin Zhao, Ninghao Liu and Tianming Liu are with the School of Computing, University of Georgia, Athens, GA, USA. (e-mail: {hd54134, zl18864, zw63397, lin.zhao, ninghao.liu, tliu}@uga.edu).
- Wenxiong Liao, Xiaoke Huang, Hongmin Cai are with the School of Computer Science and Engineering, South China University of Technology, China. (e-mail: {cswxliao@mail.scut.edu.cn, csxkhuang@mail.scut.edu.cn, hmcai@scut.edu.cn}).
- Wei Liu is with the Department of Radiation Oncology, Mayo Clinic, Phoenix, AZ, USA. (e-mail: liu.wei@mayo.edu)
- Sheng Li is with the School of Data Science, University of Virginia, Charlottesville, VA, USA. (email: shengli@virginia.edu)
- Dajiang Zhu is with the Department of Computer Science and Engineering, The University of Texas at Arlington, Arlington, TX, USA. (e-mail: dajiang.zhu@uta.edu)
- Quanzheng Li and Xiang Li are with the Department of Radiology, Massachusetts General Hospital and Harvard Medical School, Boston, Massachusetts, USA. (e-mail: li.quanzheng@mgh.harvard.edu, xiangli.shawn@gmail.com)
- Dinggang Shen is with School of Biomedical Engineering, ShanghaiTech University, Shanghai 201210, China. He is also with Shanghai United Imaging Intelligence Co., Ltd., Shanghai 200230, China, and Shanghai Clinical Research and Trial Center, Shanghai, 201210, China. (e-mail: Dinggang.Shen@gmail.com)

tators [20]. LLMs are trained in self-supervised manners, which scale up with the near-infinite amount of text corpus available in the open domains. The large parameter space of LLMs also allows them to store a large amount of knowledge, while large-scale pre-training (e.g., the autoregressive objective in training GPTs) enables LLMs to encode rich factual knowledge for language generation. Furthermore, the training of ChatGPT follows that of Instruct-GPT [21], which utilizes reinforcement learning with human feedback (RLHF), thus enabling it to produce more informative and impartial responses to input.

Inspired by the success of applying language models in text generation, we propose a new data augmentation method named ChatAug, which leverages ChatGPT to generate auxiliary samples for few-shot text classification. We have tested the performance of ChatAug via experiments on both general domain and medical domain datasets. Performance comparison of the proposed ChatAug approach with existing data augmentation methods shows double-digit improvements in sentence classification accuracy. Further investigation into the faithfulness and completeness of the generated text samples reveals that ChatAug can generate more diversified augmented samples while simultaneously maintaining their accuracy (i.e., semantic similarity to the data labels). We envision that the development of LLMs will lead to human-level annotation performance, thus revolutionizing the field of few-shot learning and many tasks in NLP.

## 2 RELATED WORK

### 2.1 Data Augmentation

Data augmentation, the artificial generation of new text through transformations, is widely used to improve model training in text classification. In NLP, existing data augmentation methods work at different granularity levels: characters, words, sentences and documents.

Data augmentation at the character level refers to the method of randomly inserting, exchanging, replacing or deleting some characters in the text [22], which improves the robustness of the NLP model against noises in text data. Another method called optical character recognition (OCR) data augmentation generates new text by simulating the errors that occur when using OCR tools to recognize text from pictures. Spelling augmentation [23] deliberately misspells some frequently misspelled words. Keyboard augmentation [22] simulates random typo errors by replacing a selected key with another key close to it on the QWERTY layout keyboard.

Data augmentation also works at the word level. Random swap augmentation randomly exchanges two words in the text, and random deletion augmentation randomly deletes some words [24]. Synonym augmentation uses synonym databases such as PPDB [25] to replace randomly selected words [26]. WordNet [27] is also widely used as a reference for synonym augmentation. This method maintains semantic consistency in samples and is suitable for text classification tasks. Wang et al. [28] proposed a data augmentation method based on word embeddings, which replaces selected words with their top- $n$  similar words to create a new sentence. Different pre-trained word embeddings

are considered (e.g., GoogleNews Lexical Embeddings [29]). This method is based on the principle that words close to each other in the embedding space often appear in similar contexts, which might help with maintaining grammatical consistency.

However, a serious limitation of word embedding-based methods is that close words in the embedding space are not necessarily semantically similar, yet semantic changes can affect the classification results. For example, “hot” and “cold” usually appear in similar contexts, so their word embeddings are close, but they have exactly opposite semantic meanings. The counter-fitting embedding augmentation [30], [31] solves this problem by using a synonym dictionary and an antonym dictionary to adjust the initial word embeddings. Specifically, the distance between embeddings of synonyms will be shortened, and the distance between embeddings of antonyms will become enlarged.

Contextual augmentation [32], [33] is another word-level data augmentation method, which uses masked language models (MLMs) such as BERT [34], DistilBERT [35] and RoBERTa [36] to generate new text based on the context. Specifically, they insert  $\langle \text{mask} \rangle$  tokens in some positions of the text, or replace some words in the text with  $\langle \text{mask} \rangle$  tokens, and then let the MLM predict what words should be put in these masked positions. Since MLMs are pre-trained on a large number of texts, contextual augmentation can usually generate meaningful new texts.

Some text data augmentation methods work at the sentence and document level. For example, back translation augmentation [37] uses language translation models for data augmentation. Specifically, the language model first translates the text into another language, and then translates it back to the original language. Due to the randomness of the translation process, the augmented text is different from the original text, but semantic consistency is maintained. At the document level, Gangal et al. [38] proposed a method to paraphrase the entire document to preserve document level consistency.

In general, regardless of the granularity level or the text generation backbone (i.e., rule-based methods or language models), the goal of data augmentation is to produce sensible and diverse new samples that maintain semantic consistency.

### 2.2 Few-shot Learning

Deep learning has achieved remarkable success in various data-intensive applications. However, the performance of deep models could be affected if the dataset size is small in the downstream tasks. Few-shot Learning is a branch of science that focuses on developing solutions to address the challenge of small sample sizes [1], [39]. FSL research aims to leverage prior knowledge to rapidly generalize to new tasks that contain only a few labeled samples. A classic application scenario for few-shot learning is when obtaining supervised examples is difficult or not possible due to privacy, safety, or ethical considerations. The development of few-shot learning enables practitioners to improve the efficiency and accuracy of text classification in various scenarios and deploy practical applications.

Recent advances in few-shot learning have shown promising results in overcoming the challenges of limited

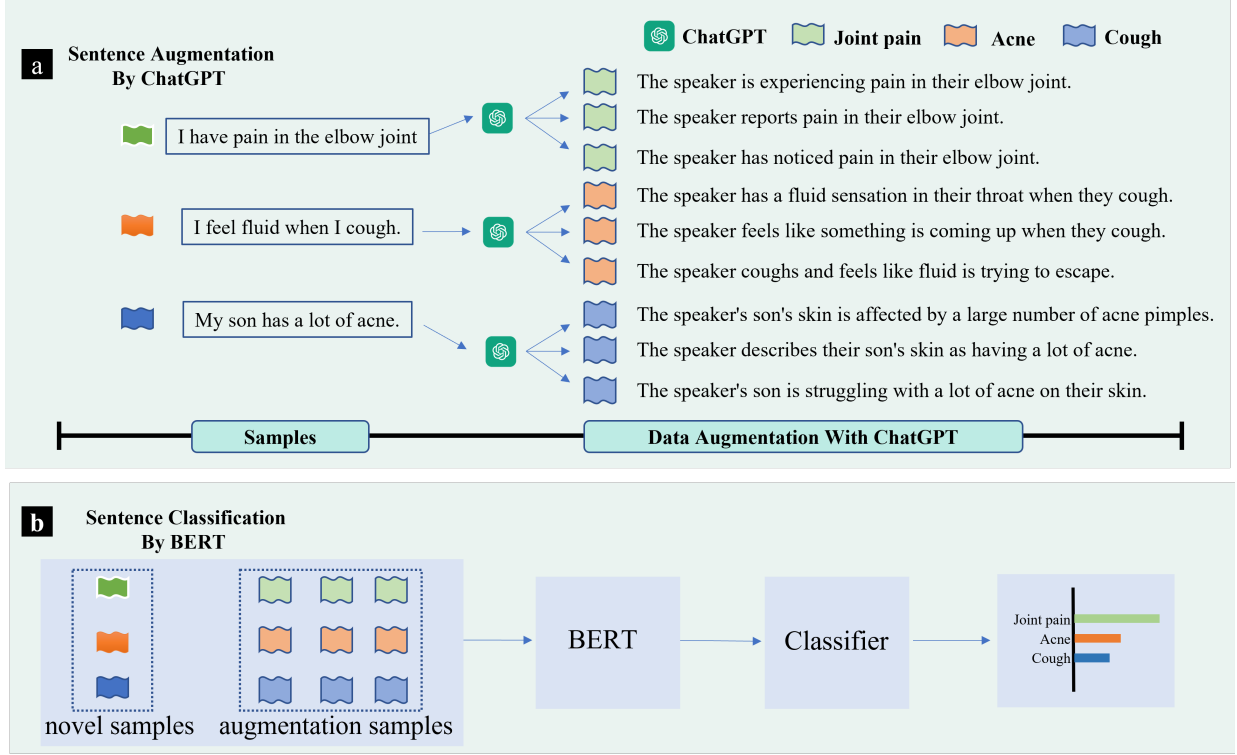


Fig. 1. The framework of ChatAug. a (top panel): First, we apply ChatGPT for data augmentation. We input samples of all classes into ChatGPT and prompt ChatGPT to generate samples that preserves semantic consistency with existing labelled instance. b (bottom panel): In the next step, we train a BERT-based sentence classifier on the few-shot samples and the generated data samples and evaluate the model’s classification performance.

training data for text classification. For example, a common approach in NLP is to use a pre-trained language model such as BERT [6] as a starting point and then fine-tune it with limited samples. Some of the most recent methodological developments [2], [4], [40] approaches that have gained traction include prompt-tuning [8], [9], [10], [11] and meta-learning [4], [7]. In general, existing FSL methods target either architectural design [3], [4], [5], data augmentation [12], [13] or the training process [41].

Despite the recent development of prompt-tuning and meta-learning methods, they suffer from some major limitations. For example, prompt engineering is a cumbersome art that requires extensive experience and manual trial-and-errors [42]. Meta-learning, on the other hand, suffers from problems such as training instability [43], [44], [45] and sensitivity to hyper-parameters [43], [44]. In addition, all these FSL pipelines demand deep machine learning expertise and acquaintance with complex model architectures and training strategies, which are not attainable by common practitioners and general developers. As discussed in section 2.1, data augmentation is an effective solution for FSL and can be combined with other FSL models. Thus, the ChatAug method proposed in this paper, which has demonstrated the capability to generate accurate and comprehensive training samples, can overcome the issues of current FSL methods and potentially change the landscape of few-shot learning in NLP.

## 2.3 Very Large Language Models

Pre-trained language models (PLMs) based on the transformer architecture, such as the BERT [6] and GPT [46] model families, have revolutionized natural language processing. Compared to previous methods, they deliver state-of-the-art performance on a wide range of downstream tasks and contribute to the rising popularity and democratization of language models. In general, there are three classes of pre-trained language models: autoregressive language models (e.g., the decoder-based GPT), masked language models (e.g., the encoder-based BERT) and encoder-decoder models (e.g., BART [47] and T5 [48]). These models typically contain between 100M and 1B parameters [19].

In recent years, NLP communities have witnessed the rise of very large language models such as GPT-3 (175B parameters) [8], PaLM (540B parameters) [49], Bloom (176B parameters) [50], OPT (up to 175B parameters) [51], and the FLAN series (FLAN has 137B parameters) [52]. At their core, these large language models are transformer models inspired by BERT and GPT, albeit at a much larger scale.

Large language models aim to learn accurate latent feature representations of input text. These representations are often context-dependent and domain-dependent. For example, the vector representation of the word “treat” might be vastly different between medical domains and the general domain. For smaller pre-trained language models, it is often necessary to continuously pre-train and fine-tune such models to attain acceptable performance [53]. However, very large language models can potentially eliminate the need for fine-tuning while maintaining competitive performance [8],

TABLE 1  
Generated Samples From ChatGPT on PubMed20K

Original Text	Altered Text
The follow-up rate after 5 years was 85%.	<ol style="list-style-type: none"> <li>1. The study showed a follow-up rate of 85% after 5 years.</li> <li>2. The 5-year follow-up rate was found to be significantly high at 85%.</li> <li>3. The participants had an impressive follow-up rate of 85% after 5 years.</li> <li>4. The data indicated a follow-up rate of 85% 5 years post-study.</li> <li>5. The 5-year follow-up rate was recorded as 85%, according to the findings.</li> <li>6. The results of the study showed a follow-up rate of 85% after a 5-year period.</li> </ol>
This means that the maturation of calluses was 27% faster in the LIPUS group.	<ol style="list-style-type: none"> <li>1. The LIPUS group showed a 27% acceleration in callus maturation.</li> <li>2. The results revealed that the callus maturation process was 27% faster in the LIPUS group.</li> <li>3. The study found that the callus maturation rate in the LIPUS group was 27% quicker.</li> <li>4. Callus maturation was 27% more rapid in the LIPUS group compared to others.</li> <li>5. The callus maturation process in the LIPUS group was 27% more efficient, according to the results.</li> </ol>
Test Registration: IRCT.ir IRCT2012071010230N1.	<ol style="list-style-type: none"> <li>1. The trial was registered with the IRCT under the identifier IRCT2012071010230N1.</li> <li>2. The study was officially registered with IRCT.ir, with the registration number IRCT2012071010230N1.</li> <li>3. The trial was documented with IRCT.ir with the identification code IRCT2012071010230N1.</li> <li>4. IRCT.ir served as the official registrar for the trial, with the registration number IRCT2012071010230N1.</li> <li>5. The study was recorded with IRCT under the registration number IRCT2012071010230N1.</li> <li>6. IRCT.ir recorded the trial with the identifier IRCT2012071010230N1.</li> </ol>
While behavioral and technological interventions can lead to some modest improvements in glycemic control, these interventions have not performed much better than conventional prevention in achieving glycemic control.	<ol style="list-style-type: none"> <li>1. The study found that although behavioral and technological interventions led to some slight improvements in glycemic control, they were not significantly more effective than typical care.</li> <li>2. Despite the modest improvement in glycemic control through behavioral and technological interventions, they did not perform better than the standard care.</li> <li>3. The results showed that while behavioral and technological interventions resulted in some minimal gains in glycemic control, they did not surpass the usual care in achieving glycemic control.</li> <li>4. Although behavioral and technological interventions showed some improvement in glycemic control, they were not found to be significantly superior to the usual care.</li> <li>5. The study showed that the usual care was not outperformed by behavioral and technological interventions in terms of achieving glycemic control, despite some small improvements.</li> </ol>

[54].

Existing studies indicate that pre-trained language models can help augment a dataset with new samples with similar semantic meaning [14], [18], which is of significant practical value to real-world applications. In this study, we aim to use ChatGPT, a popular LLM to conduct data augmentation. ChatGPT is based on GPT-3 [8], which was trained on massive web data with diverse and rich information. Furthermore, ChatGPT was trained through Reinforcement learning from Human Feedback (RLHF). During RLHF, human feedback is incorporated into the process of generating and selecting the best results. More specifically, a reward model is trained based on human annotators' ranking or generated results. In turn, this reward model rewards model outputs that are most aligned with human preference and human values. We believe these innovations make ChatGPT the best candidate for generating human-level quality data samples.

## 2.4 ChatGPT: Present and Future

ChatGPT is a game changer in natural language processing. Indeed, for the first time in human history, the power of large language models is accessible to the general public through a user-friendly chatbot interface. In turn, this common accessibility contributes to ChatGPT's unprecedented popularity. Millions of users further unlock the potential of language models, which introduces myriad possibilities for new use cases.

ChatGPT has emerged as a general-purpose problem solver for many NLP applications [55]. Qin et al. [55] evaluated ChatGPT on a comprehensive set of NLP tasks, including common benchmarks in natural language inference, arithmetic reasoning, named entity recognition, sentiment analysis, question answering, dialogue and summarization. They conclude that ChatGPT excels in most tasks, except for tasks that focus on specific details (e.g., sequence tagging).

ChatGPT is also a valuable solution for multilingual tasks. A recent empirical study [56] reports that ChatGPT excels at tasks involving high-resource languages (various European languages and Chinese) and is comparable with Google Translate, DeepL Translate and Tencent TranSmart. Nonetheless, ChatGPT performs poorly on low-resource languages and faces extra challenges handling distant language translation (i.e., English-German translation is considered to be less "distant", compared to English-Hindi translation). A later study [57] confirms that ChatGPT struggles with low-resource languages, although the authors observe that ChatGPT does better in understanding non-Latin scripts than generating them.

In addition, it is also possible to use the purely text-based ChatGPT to interact with multimodal data. A group of researchers [57] use HTML Canvas and Python Turtle graphics as media for text-to-image generation. ChatGPT can faithfully generate HTML and Python code, which can be then used to generate desired images. The authors designed a flag drawing task that required ChatGPT to generate code that can generate country flags. It was found

that ChatGPT could generate better flags when the prompt for code was preceded by a prompt that queries ChatGPT for the flag’s description. In other words, descriptive text prompts could improve multimodal task performance.

Beyond computer science, ChatGPT can be readily applied to medical report generation and comprehension [58], [59], education [60], [61], [62], rigorous math research [63] and finance [64]. Overall, ChatGPT is a versatile tool that promotes general AI usage.

However, researchers are also cautious about the possible negative impact of ChatGPT. Some of the more prominent concerns are related to bias [65], [66], ethics [67], [68], plagiarism [69], [70] and job replacement *en masse* [71], [72]. In response, a commentary published in Nature advocates for urgent attention to accountability, open-source large language models and societal embrace of AI [65].

### 3 DATASET

In this work, we use clinical natural language processing (clinical NLP) as the task and carry out our experiments on two popular public benchmarks. Data augmentation is particularly in demand in clinical NLP, because the significant burden of expert annotation and stringent privacy regulations make large-scale data labeling infeasible. We will describe these datasets in detail in the following sections.

#### 3.1 Symptoms Dataset

This dataset is published on Kaggle<sup>1</sup>. It contains the audio data of common medical symptom descriptions over 8 hours. We use the text transcripts corresponding to the audio data and perform sample de-duplication. The dataset after preprocessing includes 231 samples of 7 symptom categories.

#### 3.2 PubMed20k Dataset

PubMed20K is a widely used dataset in natural language processing (NLP) and text mining research. It consists of approximately 20,000 scientific abstracts from the biomedical domain that have been annotated with task-specific labels, such as named entities (e.g., genes, diseases, chemicals), relations between entities, and other semantic roles. The dataset has been used for developing and evaluating machine learning models for various NLP tasks, such as named entity recognition, relation extraction, and text classification.

PubMed20K is constructed based on the PubMed database, which is a large collection of biomedical literature maintained by the US National Library of Medicine. The abstracts in PubMed20K cover a wide range of topics in biomedicine, including genomics, pharmacology, and clinical medicine. Due to its size, diversity, and high-quality annotations, PubMed20K has become a popular benchmark dataset for evaluating the performance of machine learning models in biomedical NLP [73].

## 4 METHOD

### 4.1 Overall Framework

1. <https://www.kaggle.com/datasets/paultimothymooney/medical-speech-transcription-and-intent>

**Algorithm 1** The framework of ChatAug for few-shot text classification.

**Input:** base dataset  $D_b$  and novel dataset  $D_n$   
**Initialize:** Initialized pre-trained BERT *model*

**Definition:**  $D'$  is the dataset with the base dataset  $D_b$  and augmented dataset  $D_n^{aug}$ , and *chatGPT\_aug* is the data augmentation method based on ChatGPT

**Parameters:** Fine-tuning epochs of base dataset  $epoch_b$ , fine-tuning epochs of FSL  $epoch_f$

```

for epoch in  $epoch_b$  do
    train(model,  $D_b$ )
end for
 $D_n^{aug} = \text{chatGPT\_aug}(D_n)$ 
 $D' = D_b \cup D_n^{aug}$ 
for epoch in  $epoch_f$  do
    train(model,  $D'$ )
end for

```

Given a base dataset  $D_b = \{(x_i, y_i)\}_{i=1}^{N_b}$  with a label space  $y_i \in Y_b$ , a novel dataset  $D_n = \{(x_j, y_j)\}_{j=1}^{N_n}$  with a label space  $y_j \in Y_n$ , and  $Y_b \cap Y_n = \emptyset$ . In the few-shot classification scenario, the base dataset  $D_b$  has a relatively larger set of labeled samples, while the novel dataset  $D_n$  has only a few labeled samples. The performance of few-shot learning is evaluated on the novel dataset. Our goal is to train a model with both base and limited novel datasets, while achieving satisfying generalizability on the novel dataset.

The overall framework of ChatAug is shown in Fig 1, and the training steps are shown in Algorithm 1. First of all, we fine-tune BERT on  $D_b$ . Then, the  $D_n^{aug}$  is generated by data augmentation with ChatGPT. Finally, we fine-tune BERT with  $D' = D_b \cup D_n^{aug}$ .

### 4.2 Data Augmentation with ChatGPT

Similar to GPT [46], GPT-2 [74], and GPT-3 [8], ChatGPT belongs to the family of autoregressive language models and uses transformer decoder blocks [75] as the model backbone.

During pre-training, ChatGPT is regarded as an unsupervised distribution estimation from a set of samples  $X = \{x_1, x_2, \dots, x_n\}$ , and sample  $x_i$  composed of  $m$  tokens is defined as  $x_i = (s_1, s_2, \dots, s_m)$ . The objective of pre-training is to maximize the following likelihood:

$$L(x_i) = \sum_{i=1}^m \log P(s_i | s_1, \dots, s_{i-1}; \theta) \quad (1)$$

where  $\theta$  represents the trainable parameters of ChatGPT. The tokens are represented by token embedding and position embedding:

$$h_0 = x_i W_e + W_p \quad (2)$$

where  $W_e$  is the token embedding matrix and  $W_p$  is the position embedding matrix. Then  $N$  transformer blocks are used to extract the features of the sample:

$$h_n = \text{transformer\_blocks}(h_{n-1}) \quad (3)$$

where  $n \in [1, N]$ .

Finally, the target token is predicted:

$$s_i = \text{softmax}(h_N W_e^T) \quad (4)$$

where  $h_N$  is the output of top transformer blocks.

After pre-training, the developers of ChatGPT apply Reinforcement Learning from Human Feedback (RLHF) [21] to fine-tune the pre-trained language model. The RLHF aligns language models with user intent on a wide range of tasks by fine-tuning them according to human feedback. The RLHF of ChatGPT contains three steps:

**Supervised Fine-tuning (SFT):** Unlike GPT, GPT-2, and GPT-3, ChatGPT uses labeled data for further training. The AI trainers play as users and AI assistants to build the answers based on prompts. The answers with prompts build as supervised data for further training the pre-trained model. After further pre-training, SFT model can be obtained.

**Reward Modeling (RM):** Based on the SFT method, a reward model is trained to input a prompt and response, and output a scalar reward. The labelers rank the outputs from best to worst to build a ranking dataset. The loss function between two outputs is defined as follows:

$$\text{loss}(\theta_r) = E_{(x, y_w, y_l) \sim D_c} [\log(\sigma(r_{\theta_r}(x, y_w) - r_{\theta_r}(x, y_l)))] \quad (5)$$

where  $\theta_r$  is the parameters of reward model;  $x$  is the prompt,  $y_w$  is the preferred completion out of the pair of  $y_w$  and  $y_l$ ;  $D_c$  is the dataset of human comparisons.

**Reinforcement Learning (RL):** By using reward models, ChatGPT can be fine-tuned using Proximal Policy Optimization (PPO) [76]. In order to fix the performance regressions on public NLP datasets, the RLHF mix the pretraining gradients into the PPO gradients, which also known as PPO-ptx:

$$\text{objective}(\phi) = \gamma E_{x \sim D_{\text{pretrain}}} [\log(\pi_{\phi}^{\text{RL}}(x))] + E_{(x, y) \sim D_{\pi_{\phi}^{\text{RL}}}} [r_{\theta_r}(x, y) - \beta \log(\pi_{\phi}^{\text{RL}}(y | x) / \theta_{\text{SFT}}(y | x))] \quad (6)$$

where  $\pi_{\phi}^{\text{RL}}$  is the learned RL policy,  $\theta_{\text{SFT}}$  is the supervised trained model, and  $D_{\text{pretrain}}$  is the pretraining distribution. The  $\gamma$  is the pre-training loss coefficient that controls the strength of pre-training gradients, and the  $\beta$  is the KL (Kullback-Leibler) reward coefficient that controls the strength of the KL penalty.

Compared with previous data augmentation methods, ChatGPT is more suitable for data augmentation because of the following reasons:

- ChatGPT is pre-trained with large-scale corpus, so it has a broader semantic expression space, and is helpful to enhance the diversity of data augmentation.
- Since the fine-tuning stage of ChatGPT introduces a large number of manual annotation samples, the language generated by ChatGPT is more in line with human expression habits.
- Through reinforcement learning, ChatGPT can compare the advantages and disadvantages of different expressions and ensure that the augmentative data with higher quality.

Under the BERT framework, we introduce ChatGPT as the data augmentation tool for few-shot text classification. Specifically, ChatGPT is applied to rephrase each input sentence into six additional sentences, thereby augmenting the few-shot samples.

### 4.3 Few-shot Text Classification

We apply BERT [77] to train a few-shot text classification model. The output features  $h$  of the top layer of BERT can be written as:

$$z = [z_c, z_1, z_2, \dots, z_n], \quad (7)$$

where the  $z_c$  is the representation of the class special token CLS. For text classification, the  $z_c$  is usually fed into a task-specific classifier header for final prediction. However, in the scenario of FSL, it is difficult to achieve satisfactory performance through fine-tuning BERT because few-shot samples will easily lead to over-fitting and lack of generalization ability.

To effectively address the challenge of few-shot text classification, many approaches have been proposed. Generally, there are four categories of methods for few-shot text classification based on large language models: meta-learning, prompt-tuning, model design, and data augmentation. meta-learning refers to the process of *learning to learn* with tasks that update meta-parameters [4], [7]. Prompt-based methods guide large language models to predict correct results by designing templates [8], [9], [10], [11]. Model design methods guide the model to learn from few-shot samples by changing the structure of the model [78]. Data augmentation uses similar characters [22], similar word semantics [30], [31], or knowledge base [54], [79] to expand samples. Our method directly data augmentation through the language capabilities of large language models, which is a simple and efficient data augmentation method.

**Objective Function:** Our objective function of few-shot learning consists of two parts: cross entropy and contrastive learning loss. We feed  $z_c$  into a fully connected layer as the classifier for the final prediction:

$$\hat{y} = W_c^T z_c + b_c, \quad (8)$$

where  $W_c$  and  $b_c$  are trainable parameters, and take cross-entropy as one of the objective functions:

$$L_{CE} = - \sum_{d \in D'} \sum_{c=1}^C y_{dc} \ln \hat{y}_{dc}, \quad (9)$$

where  $C$  is the output dimension, which is equal to the union of label spaces of the base dataset and novel dataset, and  $y_d$  is the ground truth.

Then, to make full use of the prior knowledge in the base dataset to guide the learning of the novel dataset, we introduce the contrastive loss function to make the sample representation of the same category more compact, and the sample representation of different categories more separate. The contrastive loss between pairs of samples in the same batch is defined as follows:

$$L_{CL} = - \log \frac{\sum e^{\cos(v_i, v_{i'})}}{\sum e^{\cos(v_i, v_{i'})} + \sum e^{\cos(v_i, v_j)}}, \quad (10)$$

where  $v_i$  and  $v_{i'}$  are the  $z_c$  of samples that belong to the same category;  $v_i$  and  $v_j$  are the  $z_c$  of samples belong to different categories;  $\cos(\cdot; \cdot)$  is the cosine similarity.

In the BERT fine-tuning stage on the base dataset, we only use cross entropy as the objective function. In the

few-shot learning stage, we combine cross entropy and contrastive learning loss as the objective function:

$$L = L_{CE} + \lambda L_{CL}. \quad (11)$$

#### 4.4 Baseline Methods

In the experiment section, we compared our method with other popular data augmentation methods. For these methods, we use the implementation in open source libraries including nlpaug [80] and textattack [81].

- **InsertCharAugmentation.** This method inserts random characters at random locations in text, which improves the generalization ability of the model by injecting noise into the data.
- **SubstituteCharAugmentation.** This method randomly replaces selected characters with other ones.
- **SwapCharAugmentation** [22]. This method randomly exchanges two characters.
- **DeleteCharAugmentation.** This method randomly deletes characters.
- **OCRAugmentation.** OCRAugmentation simulates possible errors during OCR recognition. For example, OCR tool may wrongly identify "0" as "o", and wrongly identify "I" as "l".
- **SpellingAugmentation** [23]. It creates new text by deliberately misspelling some words. The method uses a list of English words that are most likely to be misspelled provided by Oxford Dictionary, for example, misspelling "because" as "becouse".
- **KeyboardAugmentation** [22]. It simulates typo error by replacing randomly selected characters with the adjacent characters in the QWERTY layout keyboard. For example, replacing 'g' with 'r', 't', 'y', 'f', 'h', 'v', 'b' or 'n'.
- **SwapWordAug** [24]. It randomly exchanges words in text. This method is a submethod of Easy Data Augmentation (EDA) proposed by Wei et al.
- **DeleteWordAug.** DeleteWordAug randomly deletes words in the text, which is also a submethod of EDA.
- **PPDBSynonymAug** [26]. It replaces words with their synonym in PPDB thesaurus. Synonym replacement can ensure semantic consistency and is suitable for classification tasks.
- **WordNetSynonymAug.** It replaces words with their synonym in WordNet thesaurus.
- **SubstituteWordByGoogleNewsEmbeddings** [28]. It replaces words with their top- $n$  similar words in the embedding space. The word embeddings used are pre-trained with GoogleNews corpus.
- **InsertWordByGoogleNewsEmbeddings** [80]. It randomly selects word from vocabulary of GoogleNews corpus and inserts it the random position of the text.
- **CounterFittedEmbeddingAug** [30], [31]. It replaces words with their neighbors in counter-fitting embedding space. Compared with GoogleNews word vectors used by SubstituteWordByGoogleNewsEmbeddings, counter-fitting embedding introduces the constraint of synonyms and antonyms, that is, the embedding between synonyms will be pulled closer, and vice versa.

- **ContextualWordAugUsingBert(Insert)** [32], [33]. This method uses BERT to insert words based on context, that is, add `< mask >` token at random position of the input text, and then let BERT predict the token at that position.
- **ContextualWordAugUsingDistilBERT(Insert).** This method uses DistilBERT to replace BERT for prediction, and the rest is the same as ContextualWordAugUsingBert(Insert).
- **ContextualWordAugUsingRoBERTA(Insert).** This method uses RoBERTA to replace BERT for prediction, and the rest is the same as ContextualWordAugUsingBert(Insert).
- **ContextualWordAugUsingBert(Substitute).** This method [32], [33] uses BERT to replace words based on context, that is, replace randomly selected words in text with `< mask >` token, and then let BERT predict the token at that position.
- **ContextualWordAugUsingDistilBERT(Substitute).** This method uses DistilBERT to replace BERT for prediction, and the rest is the same as ContextualWordAugUsingBert(Substitute).
- **ContextualWordAugUsingRoBERTA(Substitute).** This method uses RoBERTA to replace BERT for prediction, and the rest is the same as ContextualWordAugUsingBert(Substitute).
- **BackTranslationAug.** The method [37] translates the text into German and then into English, resulting in a new text that is different from the original but has the same semantics. We use wmt19-en-de and facebook/wmt19-de-en language translation models [82] developed by Facebook for translation.

#### 4.5 Evaluation Metrics

We employed cosine similarity and TransRate [83] as metrics to assess the completeness (i.e., whether features contain sufficient information about a target task) and compactness (i.e., whether features of each class are compact enough for good generalization) of our augmented data.

##### 4.5.1 Embedding Similarity

To evaluate the semantic similarity between the samples generated by data augmentation methods and actual samples, we adopt embedding similarity between the generated samples and the actual samples of the test dataset. Some of the most common similarity metrics include Euclidean distance, cosine similarity and dot product similarity. In this study, we select cosine similarity to capture the distance relationship in the latent space. The cosine similarity measures the cosine value of the angle between two vectors. This value increases when two vectors are more similar, and is bounded by a range between 0 and 1. We input sample into pre-trained BERT, and use the representation of the CLS token as sample embedding. The cosine similarity metric is commonly used in NLP [84] and we follow this convention.

$$\cos(\theta) = \frac{A \cdot B}{\|A\|_2 \|B\|_2}, \quad (12)$$

where A and B denote the two embedding vectors in comparison, respectively.

TABLE 2  
Data Augmentation Ablation Study on Symptoms

Data Augmentation	BERT	BERT+Constractive
Raw	0.636	0.606
BackTranslationAug	0.778	0.747
ContextualWordAugUsingBert(Insert)	0.697	0.677
ContextualWordAugUsingBert(Substitute)	0.626	0.667
ContextualWordAugUsingDistilBERT(Insert)	0.707	0.747
ContextualWordAugUsingDistilBERT(Substitute)	0.667	0.646
ContextualWordAugUsingRoBERTA(Insert)	0.758	0.707
ContextualWordAugUsingRoBERTA(Substitute)	0.727	0.667
CounterFittedEmbeddingAug	0.667	0.626
InsertCharAugmentation	0.404	0.475
InsertWordByGoogleNewsEmbeddings	0.636	0.677
KeyboardAugmentation	0.545	0.505
OCRAugmentation	0.768	0.778
PPDBSynonymAug	0.697	0.758
SpellingAugmentation	0.697	0.707
SubstituteCharAugmentation	0.535	0.586
SubstituteWordByGoogleNewsEmbeddings	0.727	0.727
SwapCharAugmentation	0.475	0.485
SwapWordAug	0.687	0.727
WordNetSynonymAug	0.616	0.758
ChatAug	<b>0.889</b>	<b>0.899</b>

TABLE 3  
Data Augmentation Ablation Study on PubMed20K

Data Augmentation	BERT	BERT+Constractive
Raw	0.792	0.798
BackTranslationAug	0.812	0.830
ContextualWordAugUsingBert(Insert)	0.802	0.811
ContextualWordAugUsingBert(Substitute)	0.815	0.830
ContextualWordAugUsingDistilBERT(Insert)	0.796	0.796
ContextualWordAugUsingDistilBERT(Substitute)	0.797	0.800
ContextualWordAugUsingRoBERTA(Insert)	0.815	0.814
ContextualWordAugUsingRoBERTA(Substitute)	0.782	0.782
CounterFittedEmbeddingAug	0.805	0.805
InsertCharAugmentation	0.826	0.831
InsertWordByGoogleNewsEmbeddings	0.786	0.784
KeyboardAugmentation	0.809	0.815
OCRAugmentation	0.789	0.789
PPDBSynonymAug	0.795	0.829
SpellingAugmentation	0.808	0.811
SubstituteCharAugmentation	0.816	0.821
SubstituteWordByGoogleNewsEmbeddings	0.807	0.822
SwapCharAugmentation	0.797	0.801
SwapWordAug	0.798	0.794
WordNetSynonymAug	0.761	0.757
ChatAug	<b>0.835</b>	<b>0.835</b>

#### 4.5.2 TransRate

TransRate is a metric that quantifies transferability based on the mutual information between the features extracted by a pre-trained model and their labels, with a single pass through the target data. The metric achieves a minimum value when the data covariance matrices of all classes are identical, making it impossible to distinguish between the data from different classes and preventing any classifier from achieving better than random guessing. Thus, a higher TransRate could indicate better learnability of the data. More specifically, knowledge transfer from a source task  $T_s$  to a target task  $T_t$  is measured as shown below:

$$TrR_{T_s \rightarrow T_t}(g) = H(Z) - H(Z|Y), \quad (13)$$

where  $Y$  represents the labels of augmented examples, and  $Z$  denotes the latency embedding features extracted by the

pre-trained feature extractor  $g$ .  $TrR$  means the TransRate value.  $H(\cdot)$  denotes the Shannon entropy [85].

## 5 EXPERIMENT RESULTS

In our experiments, we use BERT as the base model. First we train our model on the base dataset to get the pretrained model. Then we fine-tune the model with the few-shot samples, where we employ different data augmentation methods to generate the augmented samples. We feed those samples into BERT model to fine-tune the pretrained models. To evaluate the effectiveness of different data augmentation methods, we apply two different settings. The first one is the bare BERT model. In the second setting, we add contrastive loss during the training. In our experiments on the Symptoms dataset, we use a batch size of 8 for 150 epochs, set the maximum sequence length to 25,  $\lambda$  as 1 and use a learning rate of  $4e-5$ . Similarly, in our experiments



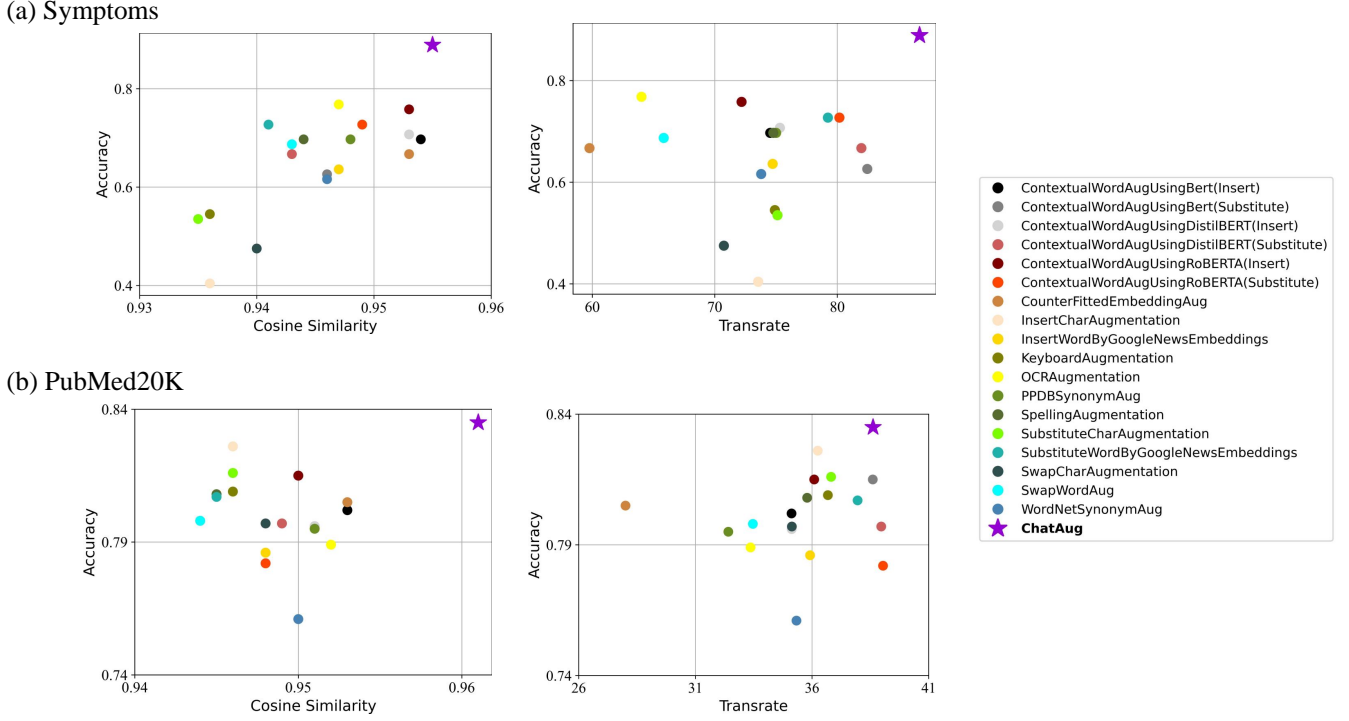


Fig. 2. We employed two evaluation metrics to assess the completeness and compactness of our newly augmented data. For the top left plot, we displayed the cosine similarity metric and final accuracy of all data augmentation methods on the Symptoms dataset. For the top right plot, we showed the TransRate metric and final accuracy of all data augmentation methods on the Symptoms dataset. In the bottom panel, we plotted the cosine similarity and TransRate values of all data augmentation methods on the PubMed20K dataset. And on the right side of the picture, we listed all the augmented methods with different colors and shapes.

on the PubMed20K dataset, we adopt the same training configuration, with the maximum sequence length set to 40.

### 5.1 Classification Performance Comparison

Table 2 and Table 3 show that ChatAug achieves the highest accuracy for both Symptoms and PubMed20K datasets. In the PubMed20K dataset, ChatAug achieves accuracies of 83.5% for both BERT and BERT with contrastive loss, whereas without data augmentation, the accuracy is only 79.2% and 79.8%, respectively. In the Symptoms dataset, the accuracy for BERT without data augmentation is only 63.6%, and 60.6% with Contrastive loss. However, our ChatAug approach significantly improves the accuracy to 88.9% and 89.9%, respectively. These results suggest that data augmentation using ChatGPT is more effective for enhancing the performance of machine learning models in various applications.

### 5.2 Evaluation of Augmented Datasets

In this section, we evaluate the performance of our augmented data in the latent space and visualize the results in Fig 2. Latent embeddings are evaluated using cosine similarity and the TransRate metric (see section 4.5 for more details). The horizontal axis represents the cosine similarity values and Transrate values, and the vertical axis describes the classification accuracy. Since embedded similarity measures the similarity between the augmentative data and the test dataset, the higher similarity means that the augmentative data more matched with the real data,

and with higher completeness and compactness. As higher TransRate could indicate better learnability of the data, the higher TransRate means the augmentative data with higher quality. The most ideal candidate method should be positioned at the top-right of the visualization. As shown in Fig 2, ChatAug produces high-quality samples in terms of both completeness and compactness on the Symptoms dataset and the PubMed20K dataset.

## 6 CONCLUSION AND DISCUSSION

In this paper, we proposed a novel data augmentation approach for few-shot classification. Unlike other methods, our model expands the limited data at the semantic level to enhance data consistency and robustness, which results in a better-performing trained model.

Although ChatAug has shown promising results in data augmentation, it has certain limitations. For example, in recognizing and expanding medical texts, it may produce incorrect augmentation data due to the lack of domain knowledge. In future research, we may fine-tune the original model first and then perform data augmentation to address this issue.

The proposed ChatAug method has shown promising results in text classification. A promising direction for future research is to investigate the effectiveness of ChatAug on a wider range of downstream tasks. For example, given the strong ability of ChatGPT to extract key points and understand sentences, we can foresee potential promising results

in text summarization. Specifically, ChatGPT might be valuable for domain-specific science paper summarization [86] and clinical report summarization [87]. Publicly available domain-specific science paper summarization datasets and clinical report datasets are rare and are often provided at small scales due to privacy concerns and the need for expert knowledge to generate annotated summaries. However, ChatGPT could address this challenge by generating diverse augmented summarization samples in different representation styles. The data generated from ChatGPT are typically concise, which can be valuable for further enhancing the generalization capabilities of the trained model.

The dramatic rise of generative image models such as DALL-E2 [88] and Stable Diffusion [89] provides opportunities for applying ChatAug to few-shot learning tasks in computer vision. For example, accurate language descriptions may be used to guide the generative model to generate images from text or to generate new images based on existing images as a data augmentation method for few-shot learning tasks, especially when combined with efficient fine-tuning methods [90], [91] such as LoRA for Stable Diffusion. Thus, prior knowledge from a large language model can facilitate faster domain adaptation and better few-shot learning of generative models in computer vision.

Recent research shows that large language models (LLMs), such as GPT-3 and ChatGPT, are capable of solving Theory of Mind (ToM) tasks, which were previously thought to be unique to humans [92]. While the ToM-like capabilities of LLMs may be an unintended byproduct of improved performance, the underlying connection between cognitive science and the human brain is an area ripe for exploration. Advancements in cognitive and brain science can also be used to inspire and optimize the design of LLMs. For example, it has been suggested that the activation patterns of the neurons in the BERT model and those in the human brain networks may share similarities and could be coupled together [93]. This presents a promising new direction for developing LLMs by utilizing prior knowledge from brain science. As researchers continue to investigate the connections between LLMs and the human brain, we may discover new means to enhance the performance and capabilities of AI systems, leading to exciting breakthroughs in the field.

## REFERENCES

- [1] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Generalizing from a few examples: A survey on few-shot learning," *ACM computing surveys (csur)*, vol. 53, no. 3, pp. 1–34, 2020.
- [2] M. Yang, "A survey on few-shot learning in natural language processing," in *2021 International Conference on Artificial Intelligence and Electromechanical Automation (AIEA)*. IEEE, 2021, pp. 294–297.
- [3] S. Sun, Q. Sun, K. Zhou, and T. Lv, "Hierarchical attention prototypical networks for few-shot text classification," in *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, pp. 476–485.
- [4] W. Yin, "Meta-learning for few-shot natural language processing: A survey," *arXiv preprint arXiv:2007.09604*, 2020.
- [5] C. Wang, J. Wang, M. Qiu, J. Huang, and M. Gao, "Transprompt: Towards an automatic transferable prompting framework for few-shot text classification," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 2792–2802.
- [6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.
- [7] H.-y. Lee, S.-W. Li, and N. T. Vu, "Meta learning for natural language processing: A survey," *arXiv preprint arXiv:2205.01500*, 2022.
- [8] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell et al., "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.
- [9] B. Lester, R. Al-Rfou, and N. Constant, "The power of scale for parameter-efficient prompt tuning," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 3045–3059.
- [10] X. Han, W. Zhao, N. Ding, Z. Liu, and M. Sun, "Ptr: Prompt tuning with rules for text classification," *AI Open*, vol. 3, pp. 182–192, 2022.
- [11] J. Wang, C. Wang, F. Luo, C. Tan, M. Qiu, F. Yang, Q. Shi, S. Huang, and M. Gao, "Towards unified prompt tuning for few-shot text classification," *arXiv preprint arXiv:2205.05313*, 2022.
- [12] J. Wei and K. Zou, "Eda: Easy data augmentation techniques for boosting performance on text classification tasks," *arXiv preprint arXiv:1901.11196*, 2019.
- [13] V. Kumar, H. Glaude, C. de Lichy, and W. Campbell, "A closer look at feature space data augmentation for few-shot intent classification," in *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, 2019, pp. 1–10.
- [14] S. Y. Feng, V. Gangal, J. Wei, S. Chandar, S. Vosoughi, T. Mitamura, and E. Hovy, "A survey of data augmentation approaches for nlp," *arXiv preprint arXiv:2105.03075*, 2021.
- [15] R. Sennrich, B. Haddow, and A. Birch, "Improving neural machine translation models with monolingual data," *arXiv preprint arXiv:1511.06709*, 2015.
- [16] A. Jindal, A. G. Chowdhury, A. Didolkar, D. Jin, R. Sawhney, and R. Shah, "Augmenting nlp models using latent feature interpolations," in *Proceedings of the 28th International Conference on Computational Linguistics*, 2020, pp. 6931–6936.
- [17] C. Shorten, T. M. Khoshgoftaar, and B. Furht, "Text data augmentation for deep learning," *Journal of big Data*, vol. 8, pp. 1–34, 2021.
- [18] M. Bayer, M.-A. Kaufhold, and C. Reuter, "A survey on data augmentation for text classification," *ACM Computing Surveys*, vol. 55, no. 7, pp. 1–39, 2022.
- [19] B. Min, H. Ross, E. Sulem, A. P. B. Veyseh, T. H. Nguyen, O. Sainz, E. Agirre, I. Heinz, and D. Roth, "Recent advances in natural language processing via large pre-trained language models: A survey," *arXiv preprint arXiv:2111.01243*, 2021.
- [20] Z. Liu, M. He, Z. Jiang, Z. Wu, H. Dai, L. Zhang, S. Luo, T. Han, X. Li, X. Jiang et al., "Survey on natural language processing in medical image analysis," *Zhong nan da xue xue bao. Yi xue ban= Journal of Central South University. Medical Sciences*, vol. 47, no. 8, pp. 981–993, 2022.
- [21] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Gray et al., "Training language models to follow instructions with human feedback," in *Advances in Neural Information Processing Systems*, 2022.
- [22] Y. Belinkov and Y. Bisk, "Synthetic and natural noise both break neural machine translation," *arXiv preprint arXiv:1711.02173*, 2017.
- [23] C. Coulombe, "Text Data Augmentation Made Simple By Leveraging NLP Cloud APIs," Dec. 2018. [Online]. Available: <http://arxiv.org/abs/1812.04718>
- [24] J. Wei and K. Zou, "EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 6382–6388. [Online]. Available: <https://aclanthology.org/D19-1670>
- [25] E. Pavlick, P. Rastogi, J. Ganitkevitch, B. Van Durme, and C. Callison-Burch, "Ppdb 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 2015, pp. 425–430.
- [26] T. Niu and M. Bansal, "Adversarial Over-Sensitivity and Over-Stability Strategies for Dialogue Models," in *Proceedings of the 22nd Conference on Computational Natural Language Learning*.

- Brussels, Belgium: Association for Computational Linguistics, 2018, pp. 486–496. [Online]. Available: <http://aclweb.org/anthology/K18-1047>
- [27] G. A. Miller, “Wordnet: a lexical database for english,” *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.
- [28] W. Y. Wang and D. Yang, “That’s so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using#petpeeve tweets,” in *Proceedings of the 2015 conference on empirical methods in natural language processing*, 2015, pp. 2557–2563.
- [29] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” *Advances in neural information processing systems*, vol. 26, 2013.
- [30] N. Mrkšić, D. Ó Séaghdha, B. Thomson, M. Gašić, L. M. Rojas-Barahona, P.-H. Su, D. Vandyke, T.-H. Wen, and S. Young, “Counter-fitting Word Vectors to Linguistic Constraints,” in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, California: Association for Computational Linguistics, Jun. 2016, pp. 142–148. [Online]. Available: <https://aclanthology.org/N16-1018>
- [31] M. Alzantot, Y. Sharma, A. Elgohary, B.-J. Ho, M. Srivastava, and K.-W. Chang, “Generating Natural Language Adversarial Examples,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, 2018, pp. 2890–2896. [Online]. Available: <http://aclweb.org/anthology/D18-1316>
- [32] S. Kobayashi, “Contextual Augmentation: Data Augmentation by Words with Paradigmatic Relations,” in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 452–457. [Online]. Available: <https://aclanthology.org/N18-2072>
- [33] V. Kumar, A. Choudhary, and E. Cho, “Data Augmentation Using Pre-trained Transformer Models,” *arXiv preprint arXiv:2003.02245*, 2020.
- [34] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” May 2019. [Online]. Available: <http://arxiv.org/abs/1810.04805>
- [35] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter,” *ArXiv*, vol. abs/1910.01108, 2019.
- [36] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “Roberta: A robustly optimized bert pretraining approach,” *arXiv preprint arXiv:1907.11692*, 2019.
- [37] R. Sennrich, B. Haddow, and A. Birch, “Improving Neural Machine Translation Models with Monolingual Data,” in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 86–96. [Online]. Available: <https://aclanthology.org/P16-1009>
- [38] V. Gangal, S. Y. Feng, M. Alikhani, T. Mitamura, and E. Hovy, “Nareor: The narrative reordering problem,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 10, 2022, pp. 10645–10653.
- [39] L. Fei-Fei, R. Fergus, and P. Perona, “One-shot learning of object categories,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 4, pp. 594–611, 2006.
- [40] Y. Ge, Y. Guo, Y.-C. Yang, M. A. Al-Garadi, and A. Sarker, “Few-shot learning for medical text: A systematic review,” *arXiv preprint arXiv:2204.14081*, 2022.
- [41] J. Wei, C. Huang, S. Vosoughi, Y. Cheng, and S. Xu, “Few-shot text classification with triplet networks, data augmentation, and curriculum learning,” in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2021, pp. 5493–5500.
- [42] T. Gao, A. Fisch, and D. Chen, “Making pre-trained language models better few-shot learners,” in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2021, pp. 3816–3830.
- [43] A. Antoniou, H. Edwards, and A. Storkey, “How to train your maml,” *arXiv preprint arXiv:1810.09502*, 2018.
- [44] C. Finn, P. Abbeel, and S. Levine, “Model-agnostic meta-learning for fast adaptation of deep networks,” in *International conference on machine learning*. PMLR, 2017, pp. 1126–1135.
- [45] X. Yao, J. Zhu, G. Huo, N. Xu, X. Liu, and C. Zhang, “Model-agnostic multi-stage loss optimization meta learning,” *International Journal of Machine Learning and Cybernetics*, vol. 12, no. 8, pp. 2349–2363, 2021.
- [46] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, “Improving language understanding by generative pre-training,” 2018.
- [47] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, “Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension,” *arXiv preprint arXiv:1910.13461*, 2019.
- [48] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, “Exploring the limits of transfer learning with a unified text-to-text transformer,” *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 5485–5551, 2020.
- [49] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann *et al.*, “Palm: Scaling language modeling with pathways,” *arXiv preprint arXiv:2204.02311*, 2022.
- [50] T. L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilić, D. Hesslow, R. Castagné, A. S. Luccioni, F. Yvon, M. Gallé *et al.*, “Bloom: A 176b-parameter open-access multilingual language model,” *arXiv preprint arXiv:2211.05100*, 2022.
- [51] S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. Diab, X. Li, X. V. Lin *et al.*, “Opt: Open pre-trained transformer language models,” *arXiv preprint arXiv:2205.01068*, 2022.
- [52] S. Longpre, L. Hou, T. Vu, A. Webson, H. W. Chung, Y. Tay, D. Zhou, Q. V. Le, B. Zoph, J. Wei *et al.*, “The flan collection: Designing data and methods for effective instruction tuning,” *arXiv preprint arXiv:2301.13688*, 2023.
- [53] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, and H. Poon, “Domain-specific language model pretraining for biomedical natural language processing,” *ACM Transactions on Computing for Healthcare (HEALTH)*, vol. 3, no. 1, pp. 1–23, 2021.
- [54] S. Rezayi, H. Dai, Z. Liu, Z. Wu, A. Hebbbar, A. H. Burns, L. Zhao, D. Zhu, Q. Li, W. Liu *et al.*, “Clinicalradiobert: Knowledge-infused few shot learning for clinical notes named entity recognition,” in *Machine Learning in Medical Imaging: 13th International Workshop, MLMI 2022, Held in Conjunction with MICCAI 2022, Singapore, September 18, 2022, Proceedings*. Springer, 2022, pp. 269–278.
- [55] C. Qin, A. Zhang, Z. Zhang, J. Chen, M. Yasunaga, and D. Yang, “Is chatgpt a general-purpose natural language processing task solver?” *arXiv preprint arXiv:2302.06476*, 2023.
- [56] W. Jiao, W. Wang, J.-t. Huang, X. Wang, and Z. Tu, “Is chatgpt a good translator? a preliminary study,” *arXiv preprint arXiv:2301.08745*, 2023.
- [57] Y. Bang, S. Cahyawijaya, N. Lee, W. Dai, D. Su, B. Wilie, H. Love-nia, Z. Ji, T. Yu, W. Chung *et al.*, “A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity,” *arXiv preprint arXiv:2302.04023*, 2023.
- [58] Y. Shen, L. Heacock, J. Elias, K. D. Hentel, B. Reig, G. Shih, and L. Moy, “Chatgpt and other large language models are double-edged swords,” p. 230163, 2023.
- [59] F. Antaki, S. Touma, D. Milad, J. El-Khoury, and R. Duval, “Evaluating the performance of chatgpt in ophthalmology: An analysis of its successes and shortcomings,” *medRxiv*, pp. 2023–01, 2023.
- [60] T. H. Kung, M. Cheatham, A. Medenilla, C. Sillos, L. De Leon, C. Elepaño, M. Madriaga, R. Aggabao, G. Diaz-Candido, J. Maningo *et al.*, “Performance of chatgpt on usml: Potential for ai-assisted medical education using large language models,” *PLOS Digital Health*, vol. 2, no. 2, p. e0000198, 2023.
- [61] J. V. Pavlik, “Collaborating with chatgpt: Considering the implications of generative artificial intelligence for journalism and media education,” *Journalism & Mass Communication Educator*, p. 10776958221149577, 2023.
- [62] D. Baidoo-Anu and L. Owusu Ansah, “Education in the era of generative artificial intelligence (ai): Understanding the potential benefits of chatgpt in promoting teaching and learning,” *Available at SSRN 4337484*, 2023.
- [63] S. Frieder, L. Pinchetti, R.-R. Griffiths, T. Salvatori, T. Lukasiewicz,

- P. C. Petersen, A. Chevalier, and J. Berner, "Mathematical capabilities of chatgpt," *arXiv preprint arXiv:2301.13867*, 2023.
- [64] M. Dowling and B. Lucey, "Chatgpt for (finance) research: The bananarama conjecture," *Finance Research Letters*, p. 103662, 2023.
- [65] E. A. van Dis, J. Bollen, W. Zuidema, R. van Rooij, and C. L. Bockting, "Chatgpt: five priorities for research," *Nature*, vol. 614, no. 7947, pp. 224–226, 2023.
- [66] R. W. McGee, "Is chat gpt biased against conservatives? an empirical study," *An Empirical Study (February 15, 2023)*, 2023.
- [67] A. Blum, "Breaking chatgpt with dangerous questions understanding how chatgpt prioritizes safety, context, and obedience," 2022.
- [68] H. Y. Jabotinsky and R. Sarel, "Co-authoring with an ai? ethical dilemmas and artificial intelligence," *Ethical Dilemmas and Artificial Intelligence (December 15, 2022)*, 2022.
- [69] T. Susnjak, "Chatgpt: The end of online exam integrity?" *arXiv preprint arXiv:2212.09292*, 2022.
- [70] M. Khalil and E. Er, "Will chatgpt get you caught? rethinking of plagiarism detection," *arXiv preprint arXiv:2302.04335*, 2023.
- [71] D. Castelvetti, "Are chatgpt and alphacode going to replace programmers?" *Nature*, 2022.
- [72] A. Zarifhonorvar, "Economics of chatgpt: A labor market view on the occupational impact of artificial intelligence," *Available at SSRN 4350925*, 2023.
- [73] F. Démoncourt and J. Y. Lee, "Pubmed 200k rct: a dataset for sequential sentence classification in medical abstracts," in *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 2017, pp. 308–313.
- [74] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.
- [75] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.
- [76] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.
- [77] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of NAACL-HLT*, 2019, pp. 4171–4186.
- [78] W. Liao, Z. Liu, H. Dai, Z. Wu, Y. Zhang, X. Huang, Y. Chen, X. Jiang, D. Zhu, T. Liu, S. Li, X. Li, and H. Cai, "Mask-guided bert for few shot text classification," *arXiv preprint arXiv:2302.10447*, 2023.
- [79] S. Rezayi, Z. Liu, Z. Wu, C. Dhakal, B. Ge, C. Zhen, T. Liu, and S. Li, "Agribert: Knowledge-infused agricultural language models for matching food and nutrition," *International Joint Conference on Artificial Intelligence, July 23-29, 2022, Vienna, Austria, 2022*.
- [80] E. Ma, "Nlp augmentation," <https://github.com/makcedward/nlpaug>, 2019.
- [81] J. Morris, E. Lifland, J. Y. Yoo, J. Grigsby, D. Jin, and Y. Qi, "Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2020, pp. 119–126.
- [82] N. Ng, K. Yee, A. Baevski, M. Ott, M. Auli, and S. Edunov, "Facebook fair's wmt19 news translation task submission," in *Proc. of WMT*, 2020.
- [83] L.-K. Huang, J. Huang, Y. Rong, Q. Yang, and Y. Wei, "Frustratingly easy transferability estimation," in *International Conference on Machine Learning*. PMLR, 2022, pp. 9201–9225.
- [84] J. Wang and Y. Dong, "Measurement of text similarity: a survey," *Information*, vol. 11, no. 9, p. 421, 2020.
- [85] T. M. Cover, *Elements of information theory*. John Wiley & Sons, 1999.
- [86] X. Cai, S. Liu, J. Han, L. Yang, Z. Liu, and T. Liu, "Chestxraybert: A pretrained language model for chest radiology report summarization," *IEEE Transactions on Multimedia*, pp. 1–1, 2021.
- [87] X. Cai, S. Liu, L. Yang, Y. Lu, J. Zhao, D. Shen, and T. Liu, "Covidsum: A linguistically enriched scibert-based summarization model for covid-19 scientific papers," *Journal of Biomedical Informatics*, vol. 127, p. 103999, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1532046422000156>
- [88] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, 2022.
- [89] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 684–10 695.
- [90] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," *arXiv preprint arXiv:2106.09685*, 2021.
- [91] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman, "Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation," *arXiv preprint arXiv:2208.12242*, 2022.
- [92] M. Kosinski, "Theory of mind may have spontaneously emerged in large language models," *arXiv preprint arXiv:2302.02083*, 2023.
- [93] X. Liu, M. Zhou, G. Shi, Y. Du, L. Zhao, Z. Wu, D. Liu, T. Liu, and X. Hu, "Coupling artificial neurons in bert and biological neurons in the human brain," in *Proceedings of the 37th AAAI Conference on Artificial Intelligence, AAAI, 2023*.