Data Generation for NLP Classification Dataset Augmentation: Using Existing LLMs to Improve Dataset Quality

Stanford CS224N Custom Project

Elliot Dauber
Department of Computer Science
Stanford University
dauber23@stanford.edu

Sahit Dendekuri
Department of Computer Science
Stanford University
sahit@stanford.edu

Abstract

In recent years, LLMs have made significant strides in their efficiency, complexity, and accuracy. However, the availability of high-quality training data remains a key challenge in developing domain-specific LLMs. In this paper, we investigate the effectiveness of synthetic data generation in augmenting the performance of language models. Our approach involves using existing LLMs to generate synthetic data that mimics the statistical properties and patterns of real data, similar to data augmentation techniques in computer vision models. Specifically, we evaluate the performance of a BERT classifier trained on varying ratios of synthetic to real data for the classification of music genres based on song lyrics. We aim to determine whether synthetic data can improve model performance and discuss the practical and ethical considerations associated with its use in NLP tasks. We also explore alternative data generation techniques that may enhance the effectiveness of training.

1 Key Information to include

- Mentor: Ansh Khurana
- External Collaborators (if you have any): N/A
- Sharing project: N/A

2 Introduction

In recent years, the use of NLP methods for training large language models has become widespread, with transformers dominating the industry and enabling the training of large models that are capable of generating text with impressive ability [1] While data quantity is often prioritized over quality in these training algorithms, finetuning these models requires high-quality labeled data to achieve optimal performance. Unfortunately, for domain-specific tasks, publicly available labeled datasets may not exist, requiring researchers to devote considerable effort to finding, cleaning, and labeling data. Moreover, the lack of sufficient data in some domains severely hampers progress in finetuning these large language models. In this paper, we explore the feasibility of using existing LLMs to augment datasets by leveraging their understanding of language to transform existing data into new, semantically similar data that can be integrated back into the original dataset.

Dataset augmentation techniques are a well-established approach in computer vision tasks to overcome overfitting. However, in natural language processing (NLP), these techniques are not as straightforward to implement. This is due to the complexity of natural language, which is not easily distilled into discrete representations [2]. While algorithms such as word2vec can be used for individual words or short phrases, applying these techniques at the sentence or document level is more challenging.

Nonetheless, large language models (LLMs) trained on vast amounts of data have some level of understanding of natural language that can be leveraged to generate semantically similar text [3] Our

contribution to the NLP data augmentation/generation conversation is a method that draws inspiration from computer vision techniques but employs LLMs to create synthetic data similar to existing data while introducing variation to improve model accuracy and mitigate overfitting. While it is a seemingly straightforward idea, we believe that it could have a profound impact on accelerating the development of domain-specific NLP models where large-scale data collection is infeasible or impractical.

In this paper, our focus is on the generation of data for natural language classification tasks for several reasons. Firstly, assessing the accuracy of the model will be easier, making it a more effective first analysis of the method's effectiveness. Secondly, we believe that classification tasks are more common in domain-specific NLP problems that suffer from a lack of data. Lastly, our main focus is on the use of dataset augmentation, and we have decided to devote most of our time to this problem, rather than delving into all the issues related to text generation.

This paper presents an investigation of the potential of LLMs for dataset augmentation and generation in the domain of natural language processing, specifically in the context of music genre classification. While we acknowledge that this domain already possesses a significant amount of data, we selected it based on personal interest and the potential for creativity in generating synthetic song lyrics using current LLMs. The impact of this choice will be analyzed further in subsequent sections, where we explore the broader implications and applicability of our methods in more practical scenarios.

3 Related Work

Data augmentation relates to techniques used to diversify training data for models, without having to necessarily collect more data. While data augmentation techniques are commonplace in CV, they are less commonplace in NLP given the more discrete input space of text [2]. Thus far, there has been a lot of research work done around discovering different data augmentation methods as well as work around determining its efficacy in various sparse-data scenarios.

Data Generation: The use of data augmentation methods in natural language processing (NLP) is a well-defined area of research. These methods can be broadly classified into rule-based, interpolation-based, and model-based techniques. Rule-based techniques involve perturbing tokens through the random insertion, deletion, and swapping of words or phrases, as in the case of the EDA method [4]. Interpolation-based methods, such as data mix-ups, involve the combination of different samples of labeled text to create new training data [5] Finally, model-based techniques leverage the power of existing language models, such as GPT, to generate synthetic data points based on real data, thus providing an effective means of data augmentation [3]

Applications: Data augmentation techniques have a variety of applications. One application is generating synthetic data from high-resource languages to improve NMT for low-resource languages [6] Another such application is improving few-shot learning methodologies by supplementing specific classes with synthetic data during the few-shot phase. Finally, a particularly interesting application of data augmentation is using synthetic data to mitigate bias in data sets [7]

For this paper, we aimed to evaluate not only the efficacy of data augmentation to improve classifier performance, but we also wanted to see if synthetic data can help adjust class imbalances in the real training data and thus mitigate classifier bias, as we feel this is especially important when data for underrepresented variables is not readily available. Our paper is a useful replication of the experiment conducted by Zhao et. all, as we feel more work needs to be done in evaluating whether data augmentation is a viable solution for reducing bias.

4 Approach

This paper presents an evaluation of the effectiveness of using synthetic training data to fine-tune a classifier model. Our study investigates whether using synthetic data can improve classifier accuracy in the presence of limited data, as well as whether it has the potential to reduce bias against underrepresented variables in a dataset. Our research consists of two key components: synthetic data generation and classifier fine-tuning on various mixtures of data. For the synthetic data generation component, we originally wrote a script that prompts GPT-3 to generate song lyrics by genre (Ex: "Generate rock song lyrics"). When we started generating lyrics this way, we found that the lyrics

didn't really resemble human lyrics, so we instead changed our script to use songs in the data set we are using to prompt GPT-3: "Write a song in the [insert genre] that is written in the same style as this song: [song lyrics from dataset]".

Our theory is that this type of data generation is analogous to data augmentation techniques in computer vision tasks such as flipping pictures or changing their saturation, etc – it is essentially a transformation on the original data, except that for NLP tasks it is much harder to "transform" a sentence or block of text mathematically. We hypothesize that this approach is analogous to data augmentation techniques in computer vision and can transform the original data. Our study employs Google's BERT transformer model as the classifier, and fine-tunes it on data mixtures with different proportions of synthetic and real data. We evaluate the accuracy of the classifier in identifying the genre of song lyrics. We selected four popular high-level genres and algorithmically cleaned and modified the dataset to fit our predetermined classes. Our study sheds light on the potential benefits of synthetic data generation for improving classifier accuracy in NLP tasks.

4.1 Data Cleaning

Prior to generating the data mixtures, we first cleaned our Hugging Face song lyrics data set to ensure the quality and relevance of the data. The dataset featured song lyrics from Genius, each labeled with multiple genres and sub-genres.

To prepare the data set for our experiments, we selected four genre categories that were generally orthogonal to each other in terms of subject matter and style: Gospel, Country, Rap, and Metal. We then isolated song lyrics from the original dataset that fit into these four categories and removed any null or multi-genre data points. This step was necessary to ensure that the data was consistent and meaningful for our experiments.

Once we had a clean data set, we randomly sampled enough songs from each of the four genres to satisfy a predetermined number of data points for each genre for the sake of our experiments. For example, we sampled 500 songs from the Rap genre.

Our rigorous data cleaning process ensured that we had a high-quality dataset for our experiments.

4.2 Data Mixtures

1. The first data mixture utilizes only real training data from the Hugging Face dataset. This will include no generated or augmented data whatsoever. The purpose of this data mixture is to establish a baseline level of performance, which we use to evaluate against performance with varying proportions of synthetic data.

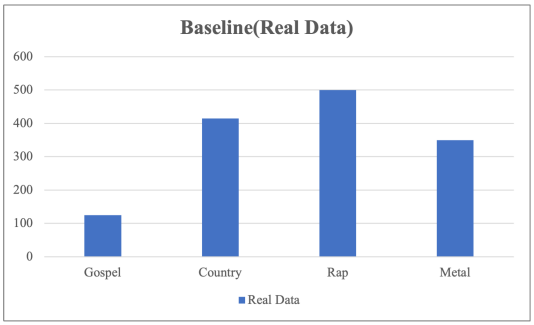

Figure 1: Dataset includes four genres in different proportions

2. The second data mixture will help us evaluate how effective synthetic data is in addressing a biased dataset, i.e. one without equal variable representation. As such, we will use a data set that has exactly equal proportions of the different music genres we are aiming to classify, with both real training data from the Hugging Face dataset and also synthetic data we've generated to address the varying proportions of the different music genres. For example, if the dataset has 100 examples of pop, 50 examples of hip-hop, and 20 examples of country, we will generate 50 examples of hip-hop and 80 examples of the country to bring every class to the same level. In general, if one class has so much more data than the other classes that it would be intractable to bring the other classes up to its level, it may be preferable to

meet somewhere in the middle (i.e. remove some of the dominating class' data). We are exploring this as well.

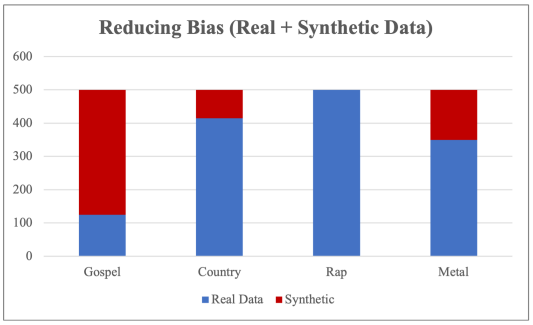

Figure 2: Dataset reduces variable bias, evening out genres with synthetic data

3. The third data mixture will involve using the same amount of real training data from the Hugging Face dataset as our baseline data mixture, however, it will also add synthetic data on top of this data such that the total lyric data maintains the same genre proportions as the baseline, just with more of it. The Hugging Face dataset, plus generated data in the same class proportions as the Hugging Face dataset. This will test if data generation can help just in the sheer amount of extra data that can be produced.

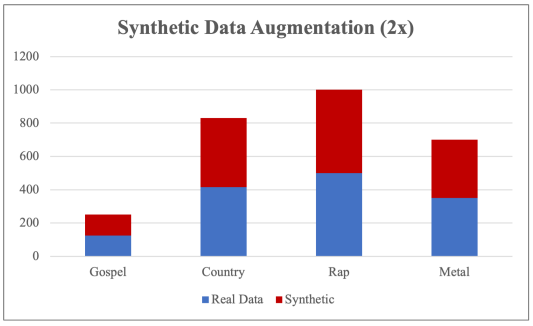

Figure 3: Total data is doubled using a combination of real and synthetic data

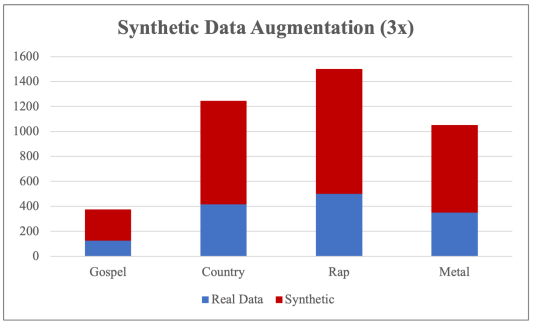

Figure 4: Total data is tripled using a combination of real and synthetic data

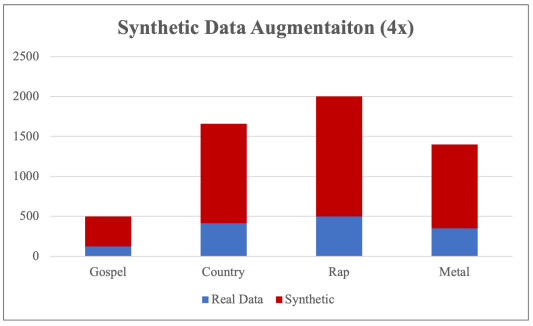

Figure 5: Total data is quadrupled using a combination of real and synthetic data

4. The fourth data mixture will be fully synthetically generated in the same proportions and amount as our baseline. This will allow us to test if synthetic data can fully replace real training data in fine-tuning a classifier.

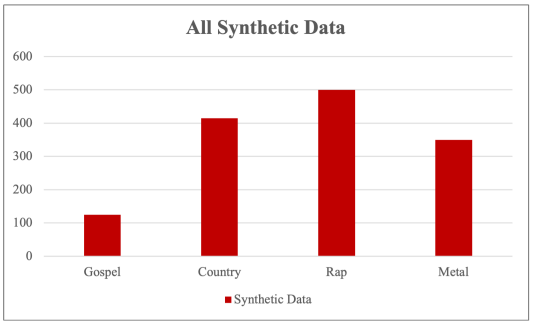

Figure 6: Fully synthetic data in same amounts as baseline

5. The fifth data mixture combines the principles of both the second and fourth data mixture in that it uses only synthetic data and also maintains equivalent genre proportions to see if synthetic data can address both bias and also lack of good training data simultaneously.

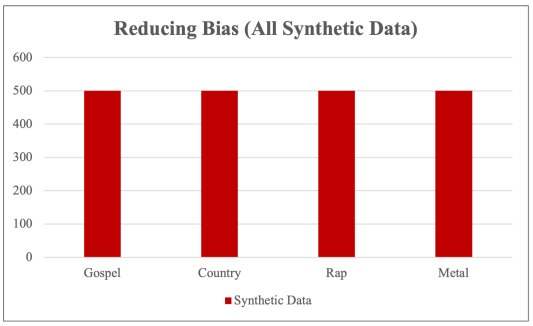

Figure 7: Fully synthetic data with even genre proportions

4.3 Data Generation

To obtain a starting point for data generation, we sourced real data from the "genius-lyrics" dataset provided by Bruno Kreiner on HuggingFace. This dataset consists of lyrics scraped from Genius Lyrics, a reputable music lyric service, and is labeled with genre information. Although the data quality is high, the genre information is represented in a list format containing specific sub-genres, rather than a simple overarching genre. To address this, we selected four high-level genres as our target classes and algorithmically mapped each song in the dataset to one of these four classes. We also removed songs that did not fit into any of our predetermined classes. We will elaborate further on our choice of genres and the cleaning process in the subsequent sections of this paper.

Once we had our base dataset, we used OpenAI's Davinci model API to generate text based on random song lyrics from our dataset. Originally, we had prompted Davinci with the prompt "Generate a song in the *metal* genre", but the data got very repetitive and didn't resemble real song lyrics.

We then realized we could use the existing "real" data to inform the generation of new data, and changed the prompt to "Write a song in the *country* genre, that is written in the same style as the song with these lyrics: *real lyrics here*" This gave a huge increase in the quality of the lyrics. With this new prompting strategy, we got lyrics such as:

Real Song Lyrics	Synthetic Song Lyrics
Well, I got my first truck, when I was three *Drove a hundred thousand miles on my knees* *Hauled marbles and rocks, and thought twice before* *I hauled a Barbie Doll bed for the girl next door* *She tried to pay me with a kiss and I began to understand* *There's just something women like about a Pickup Man*	*I was born in a truck town, where you had to earn your stripes* *Growin' up I was runnin' around, in my daddy's old beater truck* *I'd head out for open roads, coastin' through the Texas night* *With the windows rolled down, music on loud, livin' my life*

Table 1: Comparison of real and synthetic song lyrics.

4.4 Noise for Data Generation

When using Open AI's Davinci model to generate our synthetic song lyrics, one key decision we had to make was our input for the temperature (noise) parameter, which controls the degree of randomness or unpredictability in the generated output. To evaluate which value to use, we determined the METEOR score, a metric used to evaluate the quality of machine translation output when compared to the reference text, for each of the possible noise inputs. We did this by generating a synthetic song lyric from a reference lyric at each of the different noise values and found that 0.9 had the lowest METEOR score. Our aim with the synthetic data was to have song lyrics that loosely resemble the original one while maintaining the genre identity so that the training data would be more diverse and produce a higher classification accuracy. As such, we settled on 0.9 as our temperature parameter because the synthetic data output was most in line with our criteria.

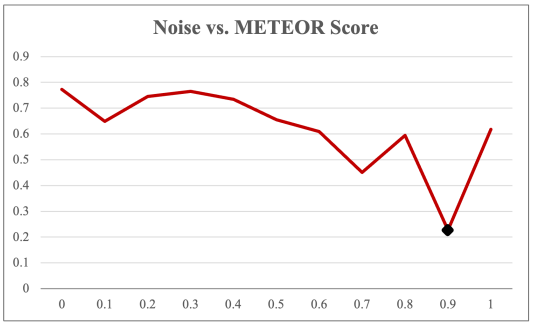

Figure 8: 0.9 was the optimal temperature parameter for Davinci

4.5 Classifier

We fine-tuned Google's BERT Transformer to train and evaluate the different datasets we tested. The reason we selected BERT is that it's bidirectional and can read a text, in this case, song lyrics, all at once. This translates to the model better being able to understand the context of both to the left and right of a given token, which is especially useful in the context of songs, as song lyrics require interpretation and are context-heavy, which means they can have entirely different meanings when evaluated in context versus not in context.

4.6 Training

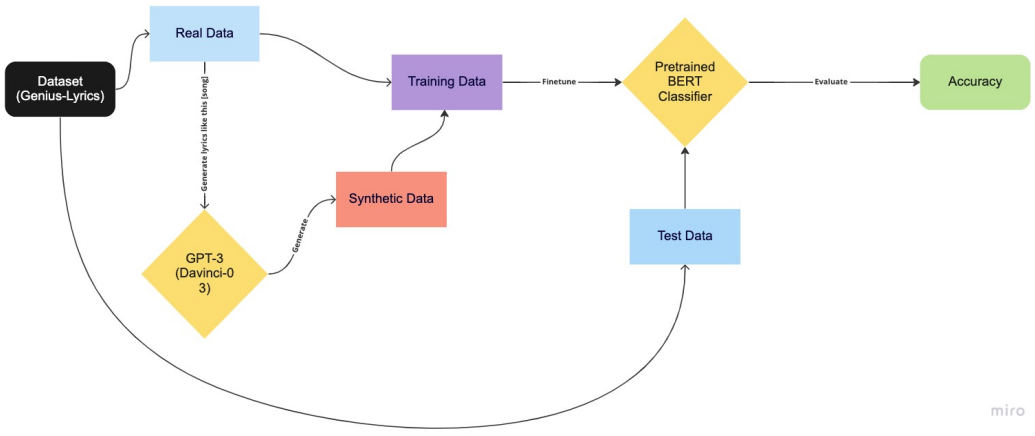

Figure 9: Data generation, training, and evaluation flowchart

Our training approach involved two key components.

The first component involved generating synthetic data. To do this, we separated out our hugging face data set into training data and test data. Then, we fed the training data into GPT-3's Davinci model with the following prompt: "Write a song in the [insert genre] that is written in the same style

as this song: [song lyrics in training data]". Once we generated our synthetic data, we then combined it with the real data in various data mixtures as described above to form our training data set.

The second component involved feeding our training data into our pre-trained BERT Classifier to fine-tune its parameters. Once the classifier was fine-tuned, we evaluated its accuracy on the test data we separated out earlier.

Model	Purpose	Specs
Text-Davinci-003	Data Generation	Temperature: 0.9, Tokens: 750-1024
BERT	Classifier	Learning Rate: 5e-5, Batch Size: 64, Epochs: 200

Table 2: Models and their specifications

5 Experiments

Our experiment structure was driven by the various data mixtures we defined in the approach section. We assembled seven sets of data, which featured varying combinations and proportions of synthetic and real data. Then, we fine tuned BERT on these different data mixtures. Finally, we evaluated the accuracy of BERT's classification on our test data to determine how the accuracy of the different data mixtures compares to our baseline of all real training data.

5.1 Data

The data set we are using for this project is "genius-lyrics" from Bruno Kreiner on Hugging Face. The data set is a well-labeled set of song lyrics from Genius that includes the musical genre that the song corresponds to. We are using this dataset to evaluate how effective our BERT classifier can identify what genre a set of song lyrics corresponds to, specifically when fine-tuned on different data mixtures. We are limiting our classification task to four genres (metal, gospel, country, rap) that are mostly orthogonal to each other in terms of the types of lyrics used.

5.2 Evaluation

To evaluate the classification accuracy of BERT when finetuned on different data mixtures, we will be using accuracy, a scoring system in binary classification: (True Positives + True Negatives) / (True Positives + True Negatives + False Positives + False Negatives), i.e. accuracy score function from sklearn. We feel that this method is the most clear-cut way to evaluate the efficacy of the model as there are only two outcomes to the classification, either the model is correct or incorrect.

5.3 Results

Experiment	Data Description	Accuracy	Improvement
Baseline	Uses only real training data from Hugging Face, without any generated or augmented data.	74.38%	(0.0%)
Reducing Bias (Synthetic + Real Data)	Uses equal proportions of different music genres, with both real and synthetic data to address bias.	81.4%	(7.02%)
Synthetic Data Augmentation (2x)	Uses additional synthetic data (2x total), maintaining the same genre proportions as the baseline.	81.23%	(6.85%)
Synthetic Data Augmentation (3x)	Uses additional synthetic data (3x total), maintaining the same genre proportions as the baseline.	81.25%	(6.87%)
Synthetic Data Augmentation (4x)	Uses additional synthetic data (4x total), maintaining the same genre proportions as the baseline.	87.39%	(13.01%)
All Synthetic Data	Fully synthetically generated in the same proportions and amount as the baseline.	87.68%	(13.3%)
Evened Out Synthetic Data	Uses synthetic data to address both bias and lack of good training data, maintaining equivalent genre proportions.	91.96%	(17.58%)

Table 3: Accuracy results for different datasets

6 Analysis

6.1 Reducing Bias (Synthetic + Real Data):

The results align with our initial hypothesis that implementing synthetic data to address class imbalances in real data significantly enhances classification accuracy beyond the baseline. This effect is possibly attributed to the synthetic data's ability to bolster classification accuracy within the underrepresented music genres present in the initial data set.

6.2 Synthetic Data Augmentation:

Overall, the results demonstrate that utilizing synthetic data augmentation universally enhances classification accuracy beyond the baseline. Notably, the 4x data augmentation approach yields a significant improvement compared to both the 2x and 3x data mixtures, as well as the baseline. Since the synthetic data is generated from the real data, it appears that a sufficient amount of it is necessary to expand the classification scope of the classifier, particularly to classify lyrics that lie on the genre's fringes. This implies that a specific ratio or threshold of synthetic data to real data must be met to achieve more substantial improvements in classification performance.

6.3 All Synthetic Data:

The results suggest that employing solely synthetic data can result in significant accuracy improvements compared to the baseline. This is likely attributed to the synthetic data's increased level of noise, allowing for a wider spectrum of genre lyrics to be covered. While utilizing only synthetic data in place of real data may not be a feasible option in practical applications, the outcomes of our investigation demonstrate the efficacy of synthetic data in enhancing classification accuracy.

6.4 Evened Out Synthetic Data:

The results demonstrate that this data mixture yields a significant boost in classification accuracy relative to the baseline. This method not only builds upon the substantial improvement observed with the exclusively synthetic data mixture but also mitigates class imbalances present in the original dataset. We speculate that this combination of factors is responsible for the additional 4% enhancement in performance.

7 Conclusion

This research project aims to investigate the effectiveness of using synthetic data for improving classification accuracy. Class imbalance and limited availability of training data can hinder the accuracy of machine learning models. Our project demonstrates how synthetic data can mitigate these issues by creating more diverse and abundant training data.

Our experiments were performed on a classification task involving song lyrics. We used a limited number of genre categories and training data to evaluate the impact of synthetic data. Our results showed that using synthetic data increased the classification accuracy by addressing class imbalance and augmenting the training data. We utilized Language Models (LLMs) to generate synthetic data that closely resembled real data, which resulted in further accuracy improvements.

However, our study has some limitations. Our focus on a narrow classification task may limit the generalizability of our findings to other tasks. Additionally, we only evaluated one method and model for generating synthetic data, which may not represent the best possible quality of synthetic data.

For future research, we suggest exploring different methods of synthetic data generation, such as GANs or MCMC, to compare their efficacy with LLMs. We also recommend investigating the effect of synthetic data on classification accuracy when larger datasets and more categories are used. Our research demonstrates the potential of synthetic data for improving classification accuracy and encourages further exploration of its applications in machine learning.

References

[1] Niki Parmar Jakob Uszkoreit Llion Jones Aidan N. Gomez Lukasz Kaiser Illia Polosukhin Ashish Vaswani, Noam Shazeer. Attention is all you need. In *arxiv*, 2017.

[2] Jason Wei Sarath Chandar Soroush Vosoughi Teruko Mitamura Eduard Hovy Steven Y. Feng, Varun Gangal. A survey of data augmentation approaches for nlp. In *arXiv*, 2021.

[3] Rewon Child David Luan Dario Amodei Ilya Sutskever Alec Radford, Jeffrey Wu. Langauge models are unsupervised multitask learners. In *arXiv*, 2018.

[4] Kai Zou Jason Wei. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. In *arXiv*, 2019.

[5] Akilesh Badrinaaraayanan Vikas Verma1 Sarath Chandar Mojtaba Faramarzi, Mohammad Amini. Patchup: A feature-space block-level regularization technique for convolutional neural networks. In *arXiv*, 2020.

[6] Antonios Anastasopoulos Mengzhou Xia, Xiang Kong and Graham Neubig. Generalized data augmentation for low-resource translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

[7] Mark Yatskar Vicente Ordonez Kai-Wei Chang Jieyu Zhao, Tianlu Wang. Gender bias in coreference resolution: Evaluation and debiasing methods. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 2018.