International Journal of Data Science and Advanced Analytics (ISSN: 2563-4429)

DATA AUGMENTATION FOR NATURAL LANGUAGE PROCESSING

Parul Gupta[1], Maha Mahmood[2]
[1]IT Department, Tata Consultancy Services, India
[2]Iraqi Prime Minister's Office, Iraq
[1]viveikh@gmail.com , [2]mahamahmood1988@gmail.com

Corresponding author email: viveikh@gmail.com

Abstract Recent years have seen a rise in the use of data augmentation approaches in natural language processing (NLP) to create more trustworthy models. Data augmentation has recently received a lot of interest in NLP due to new aims, more work being done in lower source domains, and the popularity of large-scale neural networks, which need a lot of training data. Despite this recent development, there hasn't been much research done in this area; this may be because the linguistic data presents some challenges. In this paper we compared four data augmentation (easy data augmentations (EDA), backtranslation, Mix-up and generative models like GPT-2 and BERT) approaches on two datasets for the NLP tasks of sentiment classification and question classification and used accuracy, precision, recall and f1- scores as evaluation metrics. We showed how not only accuracy, but other evaluation metrics are also required to choose the best model especially when the dataset is imbalance. We also show that these data augmentation approaches perform well only in low-data regime and the evaluation metrics for these augmentation techniques starts to get hurt when the training data is increased. Further we also concluded how backtranslation augmentation method performance depends on the language used for translation. Based on the findings, we made several recommendations for potential future work for the researchers to work on in the future.

***Keywords*—Natural Language Processing, Data Augmentation, Easy Data Augmentation, Machine Learning, Performance.**

1. Introduction

Machine learning models can be trained effectively and accurately if we have more data. Data Augmentation (DA) is a method or a technique for accumulating more data without gathering it [1]. To increase or accumulate more training data, Automatic Data Augmentation is frequently used in Computer Vision. Techniques such as cropping, flipping, and colour jittering are utilised in computer vision model training since they don't alter the image's semantics. However, the challenge arises when we try to use same generic methods in NLP [2]. It is a challenge to use generic methods for text modification because it is difficult to use such methods on text without changing the meaning or the sense of the sentence and universal data augmentation solutions in NLP have not been researched or studied properly or in depth. An ideal Data Augmentation strategy is to simply adopt and to enhance model performance because it seeks to offer an alternative to gathering more data [3]. It needs to provide techniques which are both easy to use and increase the performance efficiently. Data augmentation provides with rule-based techniques which are easier to use and implement on data but only provides marginal increase in performance whereas model-based techniques that are provided by Data augmentation show a significant increase in performance, but they are difficult to use and costlier. Additionally, the distribution of the augmented data should not be excessively similar or dissimilar from the original. Because these samples are not typical of the given domain, training on them may result in increased overfitting or subpar performance. The goal of effective DA strategies should be balance. Despite the issues and limitations there can be seen increased interest in Data augmentation as you can see in Figure 1.1 Day by day more new techniques and models are being tested for data augmentation [4] [5]. There are challenges in text data augmentation but despite these challenges new techniques are coming up for NLP tasks [6].

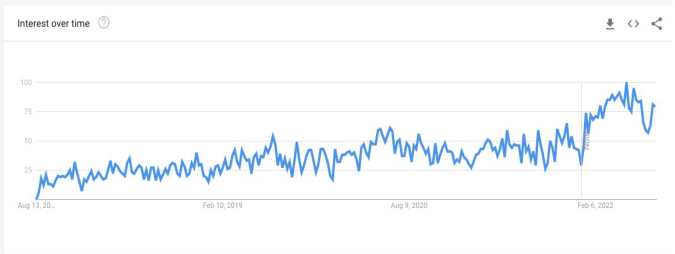

Figure 1.1: Google trends score on search of term "Data Augmentation" for last five years

The NLP community has conducted a significant amount of research on learning methods for a variety of NLP tasks. Data augmentation produces additional data by transforming existing data points through transformations developed based on prior understanding of the problem's structure [7][8]. Various data augmentation methods are being used currently that use low resource models, transformer models etc. for various NLP tasks/jobs. But, according to us overall it does seem that little is known about the precise mechanisms by which DA operates. The existing research on this subject is primarily superficial and rarely examines the underlying theories and ideas. We need more research on different datasets with different sizes and we need more data augmentation experiments done for different NLP tasks to get more efficient techniques and information regarding the mechanisms of data augmentation for text [9]. The major goal of this research is to propose an empirical study on the approaches used for data creation or data augmentation

for NLP tasks and to provide an analysis on such approachesto simplify text data augmentation for new researchers.
The objectives for this study were established as follows, in accordance with its goal:

- To conduct experiments to compare the effectiveness of various augmentation approaches.
- To assess how the accuracy is affected for each augmentation technique if the training data per class is increased.
- To recommend the best augmentation method for each text classification task on basis of evaluation metrics.
- To check if the performance of backtranslation augmentation technique depends on the translated language.

2. Related Research Work

Work done in the past has suggested various techniques for data augmentation in NLP. One of the papers provided the technique of using pool of four easy data augmentation operation: synonym replacement, random insertion, random swap, and random deletion on classification tasks and smaller dataset .This technique address NLP's inferior use of standardised data augmentation compared to computer vision by providing set of simple operations .The limitation of this technique is that gains in performance is less when training with entire datasets, the average performance gain for was less than 1%. Another technique in a popular study translated one language to another and backtranslated in to the first language. It converted English language sentences to another language and back to English this study used French as the other language to generate new data [10]. This method aids in automatically increasing the amount of training data for a variety of language-based tasks, including the one in which we are interested. Another research has employed various pre-trained transformer model types for data augmentation, including pre-trained seq2seq model, autoencoder, and auto-regressive. In order to create more data, a method known as mixup is employed in computer vision [11]. This method has been empirically explored and applied in one work to help understand mixup in NLP. A new mixup transformer model was introduced which uses BERT model and applies mixup strategy as a new dynamic data augmentation technique. There are some techniques that are not used often because the implementation cost is high than the relative performance gain like synonym replacement using predictive language models or smoothing using data noising. Recent developments in text generation models make it possible to handle circumstances where there is a lack of data in a creative way [12]. Although it may appear contradictory to improve text categorisation in these scenarios using deep learning techniques, pretrained models are presenting fresh approaches to this task.
In recent years several surveys are done that explores different techniques of data augmentation in NLP. There are surveys that focuses on specific task approaches for data augmentation to provide a review that is comprehensive. They summaries the literature in an organized way to give a thorough and unified assessment of data augmentation for NLP. Prior to discussing the main methodologically representative options, they introduce and motivate data augmentation for NLP [13]. They then describe methods that are employed in common NLP tasks and applications. They end by describing present issues and potential future research avenues. There are surveys giving an empirical study over different data augmentation methods for supervised and un-semi-supervised settings. They offer an empirical assessment of recent progress on data augmentation for NLP in the restricted labelled data context with tests on eleven datasets including single-sentence tasks., inference tasks, paraphrase tasks, and news/topics classification [14]. We summarise the available techniques including hidden-space augmentations, sentence-level augmentations, adversarial augmentations, and token-level augmentations. Including papers that briefly covers data augmentation as one of several techniques and give broader overview of techniques in low resource scenarios for NLP [15]. Numerous low-resource natural language processing techniques are reviewed. After discussing the many facets of data accessibility, they give a systematic overview of methods for supporting learning when there is a lack of training data. The purpose of their survey is to clarify how different strategies differ in their prerequisites since doing so is crucial to choose a strategy appropriate for a certain low-resource environment [16].
In contrast to survey text data augmentation our work focuses on analysing and comparing performance of data augmentation or data generation techniques that we will test on two different NLP task: Sentiment classification and Question classification. This paper would provide insights and would also be a beginners guide to data augmentation approaches in NLP that will give a basic overview on text data augmentation techniques [17]. To the best of my knowledge, no other paper has experimented and compared and analysed all four data augmentation (EDA, Backtranslation, Mixup and Pre-trained transformer model) techniques together in NLP task for question and sentiment classification. By offering an empirical assessment of several augmentation methods on two benchmark datasets, we also concentrate on their applicability to learning from limited data, giving future research on data augmentation selections some direction [18]. Only two datasets, which are used for the NLP tasks of sentiment classification and question categorization, will be used for experimentation in this study. Four different augmentation strategies, including both low-resource and pre-trained model techniques, would be used. More NLP classification problems can be used in the study. On the classification task, further augmentation techniques can be applied to determine which method is more effective [19]. This study can act as a benchmark for other investigations testing data augmentation methods on diverse datasets. We use models and methods in this paper that are simple to implement using PyTorch. Other researchers who have access to better facilities can test out more expensive and computationally intensive strategies [20].

3. Dataset selected

In our studies, we try to simulate situations where there is a lack of training data which is the case in real life. As a result, we choose a very tiny subset (10 examples from each label class) of the available training data and dev data at random from each dataset listed below to build an initial training set. We would be studying the performance of the data augmentation techniques for the tasks of sentiment classification and question classification [21]. For this research, we will be using two benchmark datasets:

•SST-2(Standford Sentiment Treebank): A corpus with fully annotated parse trees, the Stanford Sentiment Treebank, enables a thorough examination of the compositional consequences of sentiment in language. The corpus, which is made up of 11,855 single sentences taken from movie reviews with three labels positive, negative and neutral, is based on the dataset first presented. We will be using revised binary classification version of SST dataset which is known as SST-2 dataset. SST-2 dataset is a sentiment classification dataset which consists of movie reviews with binary labels of negative and positive applied on sentence level. In Figure 1.2 sample of the dataset is shown where it is labelled with 0 (negative) and 1(positive) [22].

```
there is a freedom to watching stunts that are this crude , this fast paced and this insane     1
if the tuxedo actually were a suit , it would fit chan like a 99 bargain basement special       0
as quiet , patient and tenacious as mr lopez himself , who approaches his difficult , endless work with remarkable serenity and
final verdict you 've seen it all before        0
blue crush follows the formula , but throws in too many conflicts to keep the story compelling  0
you get a sense of good intentions derailed by a failure to seek and strike just the right tone 0
a slick , engrossing melodrama  1
a wretched movie that reduces the second world war to one man 's quest to find an old flame     0
```

```
DESC:manner How did serfdom develop in and then leave Russia ?
ENTY:cremat What films featured the character Popeye Doyle ?
DESC:manner How can I find a list of celebrities ' real names ?
ENTY:animal What fowl grabs the spotlight after the Chinese Year of the Monkey ?
ABBR:exp What is the full form of .com ?
HUM:ind What contemptible scoundrel stole the cork from my lunch ?
HUM:gr What team did baseball 's St. Louis Browns become ?
HUM:title What is the oldest profession ?
DESC:def What are liver enzymes ?
HUM:ind Name the scar-faced bounty hunter of The Old West .
NUM:date When was Ozzy Osbourne born ?
```

Figure 1.2: Sample of SST-2 dataset and TREC dataset

•Text REtrieval Conference (TREC) dataset which is a question classification dataset with six labels Abb, Desc, Enty, Hum, Loc and Nym. The dataset includes 500 questions for the test set and 5500 labelled questions in the training set. The average sentence length is 10, and the vocabulary size is 8700. Data are gathered from four sources: the test set, which consists of 500 questions from TREC 10, 894 questions from TREC 8 and TREC 9, and 4,500 English questions provided by USC and about 500 manually constructed questions for a few rare classes. The sample of the dataset is provided in Figure 1.2, the dataset is worked on their paper.

For SST-2 we will be using dataset provided at https://github.com/clairett/pytorch-sentiment-classification and for TREC will be using dataset provided at https://www.tensorflow.org/datasets/catalog/trec

4. Materials and Methods

4.1 Experimental setup

In this research we have planned to start by applying four different data augmentation techniques on two datasets for NLP task of text classification (Figure 1.3). Then, we will fine be tuning the models using the datasets as we will be using small training subset from them. We will randomly be selecting 5 to 10 training subsets from both the datasets. After implementing the techniques on dataset, we will analyse the performance of each technique and model.

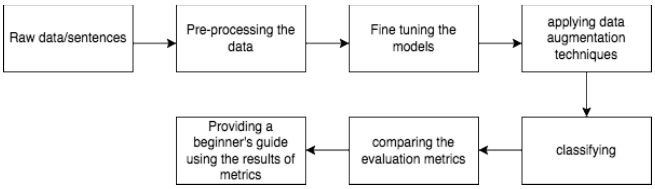

Figure 1.3 Flowchart of the experimental setup

We will evaluate and compare the results for all the techniques and the dataset using evaluation metrics. We will discuss which classification works better for which of the data augmentation techniques and compare each evaluation metrics with one another. At last, we will be providing a summary on all the techniques used in the research to provide a brief guide to people who arenew in the field of data augmentation in NLP.

4.2 Proposed Techniques

We will be using four papers as our baseline for four techniques to use on the datasets:

Easy Data augmentation (EDA)

This straightforward yet effective data augmentation method consists of four steps/operations which I will be using they are known as random deletion(RD) where each word is removed at random at a probability p from the sentence , synonym replacement(SR) involves selecting n randomly chosen words in a sentence that are not the end word and randomly substituting them with their synonyms, **random swap(RS)** two words are randomly selected in a sentence and their position is swapped and **random insertion(RI)** wherea random word in a sentence which is not the end word is selected and its random synonym is inserted at a random position. The example of the EDA operations can be seen in **Figure 1.5**.The baseline paper used both CNN and RNN as classifiers to evaluate the performance of EDA operations above operations. As seen in the base paperCNN **(convolution neural network)** performed better than RNN with the operations since sentiment is typically determined by a few key phrases, CNN are better suited for classification tasks like sentiment classification, whereas RNNs are better suited for sequence modelling tasks like language modelling, machine translation, or image captioning, which call for flexible modelling of context dependencies [23] [24]. RNNs typically excel at foretelling what will happen next in a sequence,

whereas CNNs can be trained to categorise a sentence or a paragraph. For our experiment we used BERT classifier as this model really boost the performance for NLP and shown us better result while experimentation [25].

Operation	Sentence
None	A sad, superior human comedy played out on the back roads of life.
SR	A ***lamentable***, superior human comedy played out on the ***backward*** road of life.
RI	A sad, superior human comedy played out on ***funniness*** the back roads of life.
RS	A sad, superior human comedy played out on ***roads*** back ***the*** of life.
RD	A sad, superior human out on the roads of life.

Figure 1.4 EDA operations on a sentence

Back Translation: In the base paper back translation was introduced. Using this method, the first language is translated into another language before being returned to the first language. The workflow of this method can be seen in Figure 1.5

Figure 1.5: Backtranslation workflow

•The sequence modelling toolkit Fairseq(-py) enables academics and developers to train unique models for tasks including translation, summarization, language modelling, and other text production. We will be using the language models provided by this toolkit to implement our experiment. We will be using the wmt-19 translation models. Links used to download the models we will using can be seen in Table 1.1 We will be performing backtranslation experiment/technique twice for two language (English - Russian) and (English - German). We will be using model 'wmt19.en-ru. single_model' to convert English(en) language sentences to Russian(ru) and model 'wmt19.ru-en. single_model' to translate it back to English(en). Then we will be experimenting again using model 'wmt19.en-de.joined- dict.single_model' English(en) language sentences to German(de) and model wmt19.de-en.joined-dict.single_model' to convert it back to English(en) [26].

Table 1.1:WMT-19 models and their url's

Model	• Url
wmt19.en-ru.single_model	https://dl.fbaipublicfiles.com/fairseq/models/wmt19.en-ru.single_model.tar.gz
wmt19.ru-en.single_model	https://dl.fbaipublicfiles. com/fairseq/models/wmt19.ru-en.single_model.tar.gz
wmt19.en-de.joined-dict.single_model	https://dl.fbaipublicfiles.com/fairseq/models/ wmt19.en-de.joined-dict.single_model.tar.gz
wmt19.de-en.joined-dict.single_model	https://dl.fbaipublicfiles. com/fairseq/models/wmt19.de-en.joined-dict.single_model.tar.gz

Pre-trained language models: First, we will be prepending our training data with labels after that we will be fine tuning the pre-trained language models using pytorch . For **autoencoder** we will be using **BERT** model It is a bidirectional transformer that was previously trained using both the next sentence prediction feature and a masked language modelling objective. We will be using the prepend method for BERT where will be prepending label to each sequence in the training data without adding it to the vocabulary of the model [27] .We will be using model provided by Hugging face https://huggingface.co/docs/transformers/model_doc/bert for the implementation and for **auto-regressive** we will be using **GPT-2** .It is adept at anticipating the following token in a sequence for implementation since it is trained on causal language modelling (CLM) objectives .We will be using model provided by Hugging face https://huggingface.co/docs/transformers/model_doc/gpt2.

Mixup: There are two methods mentioned in the baseline paper one is wordmixup and other is sentence mixup. We will be using sentence mixup method in this paper .In sentence mixup two different sentences are used and they are zero- padded to the same length. After passing their word embeddings through the LSTM or CNN encoder, we take the final hidden state and turn it into a sentence embedding [29]. These embeddings are mixed together in a specific ratio before being forwarded to the last classification layer. Based on both labels of the original phrases in the specified proportion, the cross-entropy loss is determined. Sentence mixup is basically performed before the softmax [30]. CNN and LSTM are used as the classification models in the baseline paper, but we will be using BERT (based-uncased) classification model for this research.

5. Evaluation metrics

The classifiers must be tested after they have been trained. The evaluation metrics used to compare the outputs of the techniques or classifiers is accuracy, precision, recall, F1 scores.

Accuracy: It is the most basic model evaluation parameter for classification models. It is the proportion of labels that were successfully predicted out of all labels. Only when your classification has an equal distribution of classes then accuracy a useful metric [31].

Precision: It is determined by dividing the total number of positive predictions by the percentage of accurately predicted true positives. It basically measures the likelihood that a predicted "Yes" is a "Yes". This metric is useful when the class has an imbalance [32].

Recall: It is described as the proportion of Positive samples that have been accurately identified as Positive to total of the actual Positive samples. This metric is useful when the class has an imbalance [33].

F1- scores: It is the precision and recall harmonic mean. There will occasionally be a trade-off between recall and precision. In these scenarios, the F-measure will decrease. When the precision and recall are both high, it will be high. Since this metric depends on precision and recall which are useful metric when class has an imbalance F1-scores tend to be an important metric [34].

The above metrics will be used to compare the modals or the techniques with each other. We will also be comparing these metrics results with each other to see which metrics gives the best result.

6. Models Evaluation Outputs

In the Table 1 .2 scores for 15 experiments of data augmentation techniques on SST-2 dataset for sentiment classification where the number of examples per class was ten in the training data .We see that the Backtranslation for en-ru has outperformed all other techniques including pre-trained models like GPT-2 and BERT-prepend for all four evaluation metrics whereas Backtranslation for en-de has not performed better than the pre-trained models on basis of accuracy but still has better f1-score than Bert-prepend. EDA although easy to implement than other techniques didn't perform as well as other models as it can be seen in the table that it only surpasses 'No data augmentation' on basis of accuracy. Mixup also is an easy to implement technique can be preferred over EDA on basis of all the metrics performance and also surpasses 'No data augmentation' on basis of all performance metrics unlike EDA. Bert-prepend has a decent accuracy but fails to perform on basis of F1-score. GPT-2 model has surpassed Backtranslation (en-de) model on basis of all the metrics. Overall, Backtranslation (en-ru) surpassed the performance in comparison to other techniques. For SST-2 backtranslation (en-ru) would be a good data augmentation choice in case of accuracy, precision, recall and f1-scores.GPT-2 can also be used for sentiment classification since SST-2 dataset is not an imbalance dataset accuracy can still be considered as a good measure to choose the augmentation model.

Table 1.2: Mean evaluation metrics for 15 experiments on SST-2 (10 examples per class)

Evaluation Metrics SST-2 (10 examples per class) →	Accuracy	Precision	Recall	F1-score
No Data Aug	51.09	54.07	47.67	50.66
EDA	54.2	51.37	44.36	47.60
Backtranslation(en-ru)	**57.9**	**54.22**	**53.42**	**53.81**
Backtranslation(en-de)	56.02	52.39	48.49	50.31
Mixup(sentence)	55.15	52.46	48.18	50.22
GPT-2	57.78	53.48	50.53	51.96
Bert-prepend	56.36	52.30	37.98	44

In the Table 1.2, we can see the average of accuracy, precision, recall, and F1-scores for 15 experiments for data augmentation techniques on TREC dataset for sentiment classification where the number of examples per class was ten. Mixup has the best accuracy in all the data augmentation techniques but Backtranslation (en-ru) surpasses Mixup and other techniques on basis of recall and F1-score and Backtranslation(en-de) surpasses Mixup and other techniques on basis of precision. Bert-prepend surpassed Backtranslation (en-de) on accuracy but still lags on basis of remaining evaluation metrics. recall and F1-score make for a better evaluation metric than accuracy. Both the Backtranslation techniques and Mixup technique has surpassed other techniques on basis of F1-score with a significant difference.

Table 1.3: Mean evaluation metrics for 15 experiments on TREC (10 example per class)

Evaluation Metrics TREC (10 examples per class)→	Accuracy	Precision	Recall	F1-score
No Data Aug	50.97	61.06	49.93	54.93
EDA	53.11	63.94	44.30	52.33
Backtranslation(en-ru)	59.77	62.62	**61.36**	**61.98**
Backtranslation(en-de)	58.6	**64.68**	58.07	61.19
Mixup(sentence)	**60.12**	64.24	56.26	59.98
GPT-2	54.28	64.04	45.12	52.94
Bert-prepend	58.93	57.59	54.09	55.73

When we compare Table 1.2 and Table 1.3 we observe how data augmentation techniques works differently for sentiment classification and question classification task specially when your dataset has imbalance class label and is in low data regime. Still Backtranslation can be considered a good data augmentation model for both datasets. In SST-2 Backtranslation(en-ru) surpasses all augmentation on basis of all performance metrics and in TREC where F1-score is a better choice for model selection there also Backtranslation(en-ru) surpasses all the other data augmentation methods. Since the mixup strategy outperforms other models in terms of accuracy and also performs significantly well in terms of F1-score, it may also be regarded as a useful technique for the TREC dataset.

Now we discuss the output of data augmentation models when the number of examples per class is increased from 10 to 50. In the Table 1.3, we can see the average of accuracy, precision, recall, and F1-scores for 15 experiments for data augmentation techniques on SST-2 dataset for sentiment classification where the number of examples per class was 50. Accuracy of all the techniques have been surpassed by the accuracy when no data augmentation technique is applied on the dataset. It means as we increase the number of examples per class in the training data the accuracy gets hurt. According to one theory, this can be because fine-tuning large pre- trained transformers on tasks provides very little benefit. It is surprising to see that in this scenario how EDA outperforms GPT-2 on basis of accuracy. Backtranslation (en-ru) model has surpassed other techniques and 'No Data aug' recall and F1-score.

Table 1.4: Mean evaluation metrics for 15 experiments on SST-2 (50 examples per class)

Evaluation Metrics SST-2 (50 examples per class)→	Accuracy	Precision	Recall	F1-score
No Data Aug	**79.84**	82.1	77.53	79.74
EDA	77.20	80.16	72.52	76.14
Backtranslation(en-ru)	79.10	80.90	**78.66**	**79.76**
Backtranslation(en-de)	78.74	81.13	75.26	78.08
Mixup(sentence)	78.15	81.58	77.12	79.28
GPT-2	74.60	80.96	74.61	77.65
Bert-prepend	78.06	**82.21**	73.49	77.60

In the Table 1.5, we can see the average of accuracy, precision, recall, and F1-scores for 15 experiments for data augmentation techniques on TREC dataset for sentiment classification where the number of examples per class was 50. recall and F1-score make for a better evaluation metric than accuracy. Accuracy of

all the techniques have been surpassed by the accuracy when no data augmentation technique is applied on the dataset. It means as we increase the number of examples per class in the training data the accuracy gets hurt. Evaluation metrics where no data augmentation technique was applied performed better than all the metrics when augmentation technique was applied to the data. It means that for TREC dataset the performance of all augmentation technique only got hurt when the examples per class were increased on training data.

Table 1.5: Mean evaluation metrics for 15 experiments on TREC(50 examples per class)

Evaluation Metrics TREC (50 examples per class)→	Accuracy	Precision	Recall	F1-score
No Data Aug	**72.26**	**74.26**	**79.35**	**76.72**
EDA	55.49	58.43	45.06	50.88
Backtranslation(en-ru)	59.52	51.50	42.71	46.69
Backtranslation(en-de)	63.18	64.89	49.90	56.41
Mixup (sentence)	64.76	62.16	47.45	53.81
GPT-2	53.08	59.94	34.77	44.01
Bert-prepend	61.98	47.39	47.63	47.50

From Table 1.4 and Table 1.5 we note that different data augmentation methods give variable results for the sentiment classification and question classification tasks, particularly when the dataset has an unbalanced class label. For both the datasets when the number of examples per class increased to 50 the performance of all the data augmentation technique got hurt. In SST- 2 data we see that Backtranslation performance better than 'No data augmentation' on basis of recall and f1-score but that to not with much significant difference. Even for Precision Bert- prepend surpasses 'No data augmentation' but not with much significant difference. For accuracy metrics none of the data augmentation technique could surpass the baseline accuracy when no data augmentation is applied. In TREC dataset we see that none of the augmentation technique performed in any of the performance metrics and had a significant difference from the baseline metrics.

7. Conclusions

For our first objective we had to compare the effectiveness of various data augmentation approaches and we found that for SST-2 dataset when the training data has 10 number of examples per class for all four performance measures, backtranslation(en-ru) outperformed all other methods, even pre-trained models like GPT-2 and BERT- prepend. Backtranslation(en-de) did not outperform pre-trained models in terms of accuracy, but it still outperformed Bert-prepend in terms of f1-score. Despite being more straightforward to use than other strategies, EDA didn't perform as well as other models and it only outperforms "No data augmentation" in terms of accuracy. Based on all performance criteria, Mixup is a simple technique that can be favored over EDA. In contrast to EDA, it also outperforms "No data augmentation" on all metrics. Bert-prepend performs poorly based on F1-score despite having a respectable accuracy. Based on all measures, the GPT-2 model has surpassed the Backtranslation (en- de) model. In terms of accuracy, precision, recall, and f1-scores, backtranslation (en-ru) would be a good data augmentation option for SST-2. Since SST-2 dataset is not an imbalance dataset, GPT-2 may still be used for sentiment classification and accuracy can still be regarded as an acceptable criterion for selecting the augmentation model. For TREC dataset which has an imbalance dataset Mixup has the highest accuracy of all the data augmentation strategies, while Backtranslation (en-ru) and Backtranslation

(en-de) both outperform Mixup and other procedures in terms of recall and F1-score and precision. Bert-prepend outperformed Backtranslation (en–de) in terms of accuracy, but it still falls short when compared to the other evaluation measures. EDA performed worse than other tactics despite being easier to use than them, it only outperformed "No data augmentation "metrics on basis of accuracy and precision. Pre-trained models GPT-2 and Bert-prepend had significant performance for all evaluation metrics but got out performed by other data augmentation techniques. On the basis of F1-score, the Backtranslation and Mixup approaches have both significantly outperformed other strategies.

- For our next objective of assessing how the accuracy is affected for each augmentation technique if the training data per class is increased. For this assessing this objective we ran the experiments when the number of examples per class for training data was ten and then we again ran the experiments and increased the number of examples per class for training data to fifty. The findings of these tests lead to the conclusion that data augmentation approaches function better in a low data regime, i.e., they performed significantly better when training data consisted of ten samples per class. When number of examples per class were increased to 50 in SST-2 dataset data augmentation techniques barely made a significant difference on basis of precision, recall and f1-score and as for accuracy it got worse for data augmentation technique than when no augmentation technique was applied. Whereas in the case of TREC dataset none of the data augmentation technique performed better than the baseline metric where no augmentation was applied. We can conclude that data augmentation perform better in a low data regime and the performance of data augmentation techniques gets hurt when the training data per class is increased.
- For our next objective we had to recommend the best augmentation method for each text classification task on basis of evaluation metrics we found that when your dataset contains imbalanced class labels and is in a low data environment, how data augmentation approaches behave differently for the sentiment classification and question classification tasks. However, for both datasets, backtranslation technqie can be regarded as a good data augmentation model. Backtranslation(en-ru) outperforms all other techniques of data augmentation in SST-2 based on all performance metrics, and it also outperforms all other methods of data augmentation in TREC where F1-score is a better option for model selection. The mixup strategy may also be regarded as a good method for the TREC dataset because it outperforms other models in terms of accuracy and also performs noticeably well in terms of F1-score. So, we can conclude that overall Backtranslation is a better choice for sentiment classification over other techniques

and Backtranslation and Mixup technique both can be regarded as a better choice for question classification over other data augmentation techniques.

• For our last objective we had to check if the performance of backtranslation augmentation technique depends on the translation language. For this objective we experimented using two models Backtranslation for English-Russian language and Backtranslation for English-German language. The results of the backtranslation models can be seen in Table 6.1. It can be seen that for SST-2 where training data had ten examples per class Backtranslation(en-ru) model outperformed Backtranslation (en- de) model on basis of all evaluation metrics. Similarly, for TREC dataset where training data had ten examples per class Backtranslation(en-ru) model outperformed Backtranslation(en-de) model on basis of all evaluation metrics except for precision. Since TREC is an imbalance dataset and Backtranslation(en-ru) model has better F1- score it is safe to say it performed better overall. In SST-2 where training data had 50 examples per class it can be seen that Backtranslation (en-ru) model outperformed Backtranslation(en-de) model on basis of all evaluation metrics except for precision. It can be seen that for TREC dataset where training data had 50 examples per class Backtranslation(en-de) model outperformes Backtranslation(en-ru) model on basis of all evaluation metrics with a significant difference.

Table 1.6: Comparing Backtranslation(en-ru) and Backtranslation(en-de) models

Model	Accuracy	Precision	Recall	F1-score
Backtranslation(en-ru) SST-2 (10 examples per class)	**57.9**	**54.22**	**53.42**	**53.81**
Backtranslation(en-de) SST-2 (10 examples per class)	56.02	52.39	48.49	50.31
Backtranslation(en-ru) TREC (10 examples per class)	**59.77**	62.62	**61.36**	**61.98**
Backtranslation(en-de) TREC (10 examples per class)	58.6	**64.68**	58.07	61.19
Backtranslation(en-ru) SST-2 (50 examples per class)	**79.10**	80.90	**78.66**	**79.76**
Backtranslation(en-de) SST-2 (50 examples per class)	78.74	**81.13**	75.26	78.08
Backtranslation(en-ru) TREC (50 examples per class)	59.52	51.50	42.71	46.69
Backtranslation(en-de) TREC (50 examples per class)	**63.18**	**64.89**	**49.90**	**56.41**

References

[1] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving Language Understanding by Generative Pre-Training," 2018. [Online]. Available: https://gluebenchmark.com/leaderboard.

[2] A. Anaby-Tavor, B. Carmeli, E. Goldbraich, A. Kantor, G. Kour, S. Shlomov, N. Tepper, and N. Zwerdling, "Do Not Have Enough Data? Deep Learning to the Rescue!" 2019. [Online]. Available: www.aaai.org.

[3] V. Barriere and A. Balahur, "Improving Sentiment Analysis over non-English Tweets using Multilingual Transformers and Automatic Translation for Data-Augmentation," in Proc. 28th Int. Conf. Computational Linguistics, Stroudsburg, PA, USA, 2020, pp. 266-271.

[4] D.R. Beddiar, M.S. Jahan, and M. Oussalah, "Data expansion using back translation and paraphrasing for hate speech detection," Online Social Networks Media, vol. 24, p. 100153, 2021.

[5] D.E. Cahyani and I. Patasik, "Performance comparison of TF-IDF and Word2Vec models for emotion text classification," Bulletin of Electrical Engineering and Informatics, vol. 105, pp. 2780-2788, 2021.

[6] J. Chen, D. Tam, C. Raffel, M. Bansal, and D. Yang, "An Empirical Survey of Data Augmentation for Limited Data Learning in NLP," 2021.

[7] C. Coulombe, "Text data augmentation made simple by leveraging nlp cloud apis," arXiv preprint arXiv:1812.04718, 2018.

[8] G. Daval-Frerot and Y. Weis, "WMD at SemEval-2020 Tasks 7 and 11: Assessing Humor and Propaganda Using Unsupervised Data Augmentation," in Proc. Fourteenth Workshop on Semantic Evaluation, Stroudsburg, PA, USA, 2020, pp. 1865-1874.

[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," 2018.

[10] H.-T. Duong and T.-A. Nguyen-Thi, "A review: preprocessing techniques and data augmentation for sentiment analysis," Computational Social Networks, vol. 81, p. 1, 2021.

[11] S.Y. Feng, V. Gangal, J. Wei, S. Chandar, S. Vosoughi, T. Mitamura, and E. Hovy, "A Survey of Data Augmentation Approaches for NLP," 2021.

[12] S. Garg and G. Ramakrishnan, "BAE: BERT-based Adversarial Examples for Text Classification," 2020.

[13] H. Guo, Y. Mao, and R. Zhang, "Augmenting Data with Mixup for Sentence Classification: An Empirical Study," 2019.

[14] M.A. Hedderich, L. Lange, H. Adel, J. Strötgen, and D. Klakow, "A Survey on Recent Approaches for Natural Language Processing in Low-Resource Scenarios," 2020.

[15] E. Hovy, L. Gerber, U. Hermjakob, C.-Y. Lin, and D. Ravichandran, "Toward semantics-based answer pinpointing," in Proc. 1st Int. Conf. Human Language Technology Research, 2001.

[16] Z. Hu, B. Tan, R.R. Salakhutdinov, T.M. Mitchell, and E.P. Xing, "Learning Data Manipulation for Augmentation and Weighting," in Advances in Neural Information Processing Systems, H. Wallach, H. Larochelle, A. Beygelzimer, F. d

Alché-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper/2019/file/671f0311e2754fcdd37f70a8550379bc-Paper.pdf.

[17] A. Jain, P.R. Samala, D. Mittal, P. Jyoti, and M. Singh, "SpliceOut: A Simple and Efficient Audio Augmentation Method," 2021.

[18] Awad, W.K., Mahdi, E.T., & Rashid, M.N. (2022). Features Extraction of Fingerprints Based on Bat Algorithms. International Journal on Technical and Physical Problems of Engineering, 14(4), 274–279.

[19] S. Kobayashi, "Contextual Augmentation: Data Augmentation by Words with Paradigmatic Relations," 2018.

[20] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M.D. Plumbley, "PANNs: Large-Scale Pretrained Audio Neural Networks for Audio Pattern Recognition," IEEE/ACM Trans. Audio, Speech, Lang. Process., vol. 28, pp. 2880-2894, 2020.

[21] A. Rajpurkar, J. Lee, R. Liang, and P. Liang, "SQuAD 2.0: The Stanford Question Answering Dataset," arXiv preprint arXiv:1806.03822, 2018.

[22] J. Kowsari, H. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, "Text Classification Algorithms: A Survey," Information, vol. 104, p. 150, 2019.

[23] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "ImageNet classification with deep convolutional neural networks," Commun. ACM, vol. 60, no. 6, pp. 84-90, 2017.

[24] V. Kumar, A. Choudhary, and E. Cho, "Data Augmentation using Pre-trained Transformer Models," 2020.

[25] 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," 2019.

[26] B. Li, Y. Hou, and W. Che, "Data augmentation approaches in natural language processing: A survey," AI Open, vol. 3, pp. 71-90, 2022.

[27] X. Li and D. Roth, "Learning Question Classifiers," in COLING 2002: The 19th Int. Conf. Computational Linguistics, 2002. [Online]. Available: https://aclanthology.org/C02-1150.

[28] H. Liu, L. Cui, J. Liu, and Y. Zhang, "Natural Language Inference in Context - Investigating Contextual Reasoning over Long Texts," Proc. AAAI Conf. Artif. Intell., vol. 35, no. 1, pp. 13388-13396, 2021.

[29] P. Liu, X. Wang, C. Xiang, and W. Meng, "A Survey of Text Data Augmentation," in 2020 Int. Conf. Comput. Commun. Netw. Security (CCNS), IEEE, 2020, pp. 191-195.

[30] S. Liu, K. Lee, and I. Lee, "Document-level multi-topic sentiment classification of Email data with BiLSTM and data augmentation," Knowledge-Based Systems, vol. 197, p. 105918, 2020.

[31] S. Longpre, Y. Lu, Z. Tu, and C. DuBois, "An Exploration of Data Augmentation and Sampling Techniques for Domain-Agnostic Question Answering," in Proc. 2nd Workshop Machine Reading for Question Answering, Stroudsburg, PA, USA, 2019, pp. 220-227.

[32] S. Longpre, Y. Wang, and C. DuBois, "How Effective is Task-Agnostic Data Augmentation for Pretrained Transformers?" in Findings of the Assoc. Comput. Linguistics: EMNLP 2020, Stroudsburg, PA, USA, 2020, pp. 4401-4411.

[33] D. Lowell, B. Howard, Z.C. Lipton, and B. Wallace, "Unsupervised Data Augmentation for Biomedical Entity Recognition," in Proc. 11th Int. Workshop on Health Text Mining and Information Analysis, Stroudsburg, PA, USA, 2020, pp. 82-93.

[34] Mahdi, E.T., Awad, W.K., Rasheed, M.M., & Mahdi, A.T. (2023). Proposed Security System for Cities Based on Animal Recognition Using IoT and Clouds. In Proceedings - International Conference on Developments in eSystems Engineering (DeSE) (pp. 834–839).• Torfi, A., Shirvani, R.A., Keneshloo, Y., Tavaf, N. and Fox, E.A., (2020) Natural Language Processing Advancements By Deep Learning: A Survey.