Assessing the Eligibility of Backtranslated Samples Based on Semantic Similarity for the Paraphrase Identification Task

Jean-Philippe Corbeil and **Hadi Abdi Ghavidel**
Polytechnique Montreal
{jean-philippe.corbeil,hadi.abdi-ghavidel}@polymtl.ca

Abstract

In the domain of natural language augmentation, the eligibility of generated samples remains not well understood. To gather insights around this eligibility issue, we apply a transformer-based similarity calculation within the BET framework based on backtranslation, in the context of automated paraphrase detection. While providing a rigorous statistical foundation to BET, we push their results by analyzing statistically the impacts of the level of qualification, and several sample sizes. We conducted a vast amount of experiments on the MRPC corpus using six pre-trained models: BERT, XLNet, Albert, RoBERTa, Electra, and DeBerta. We show that our method improves significantly these "base" models while using only a fraction of the corpus. Our results suggest that using some of those smaller pre-trained models, namely RoBERTa base and Electra base, helps us reach F1 scores very close to their large counterparts, as reported on the GLUE benchmark. On top of acting as a regularizer, the proposed method is efficient in dealing with data scarcity with improvements of around 3% in F1 score for most pre-trained models, and more than 7.5% in the case of Electra.

1 Introduction

Natural language processing (NLP) tasks require sufficiently large datasets to achieve the maximum robustness of the trained models. Low sizes of data pose the risks of hindering the models' convergence during the training process, which leads to less accurate predictions (e.g. classification) or generations (e.g. translations). On the other hand, the provision of high-quality labelled data is often very expensive both in terms of money and time. As a result, NLP scientists seek alternative methods to tackle this issue. One of the solutions is the application of data augmentation techniques. These methods help considerably to alleviate insufficiencies regarding the quantity of labelled data and the expertise to annotate the data. These augmentation techniques are also proved to induce a regularization effect during the training of NLP models to avoid overfitting, most notably, on surface cues.

In this paper, we examine the impact of adding a post-processing stage after applying such data augmentation technique to assess the eligibility of the generated samples. We intend to run our analysis in automated paraphrase identification. In this regard, we increase the size of the paraphrase data through a backtranslation method called BET (Corbeil and Abdi Ghavidel, 2020). In particular, we conduct the following experiments:

- We take randomly several samples of the original train set and augment them with backtranslation.
- After augmenting the textual data, we assess the eligibility by applying a similarity filter. We report the results for three criteria: 0.8, 0.9, and 0.95.
- We examine six pre-trained transformer models: BERT, XLNet, RoBERTa, ALBERT, Electra, and DeBerta.
- We run ten times each experiment randomizing the random seed to measure the averaged metrics and their p-values.

The remainder of this paper is structured as follows. In section 2, we describe the previous works in natural language augmentation. In section 3, we explain our methodology in terms of the dataset and our overall pipeline. Next, we illustrate and discuss the results. Finally, we summarize our findings and talk about the possible future research avenues.

Proceedings of Recent Advances in Natural Language Processing, pages 301–308
Sep 1–3, 2021.
https://doi.org/10.26615/978-954-452-072-4_035

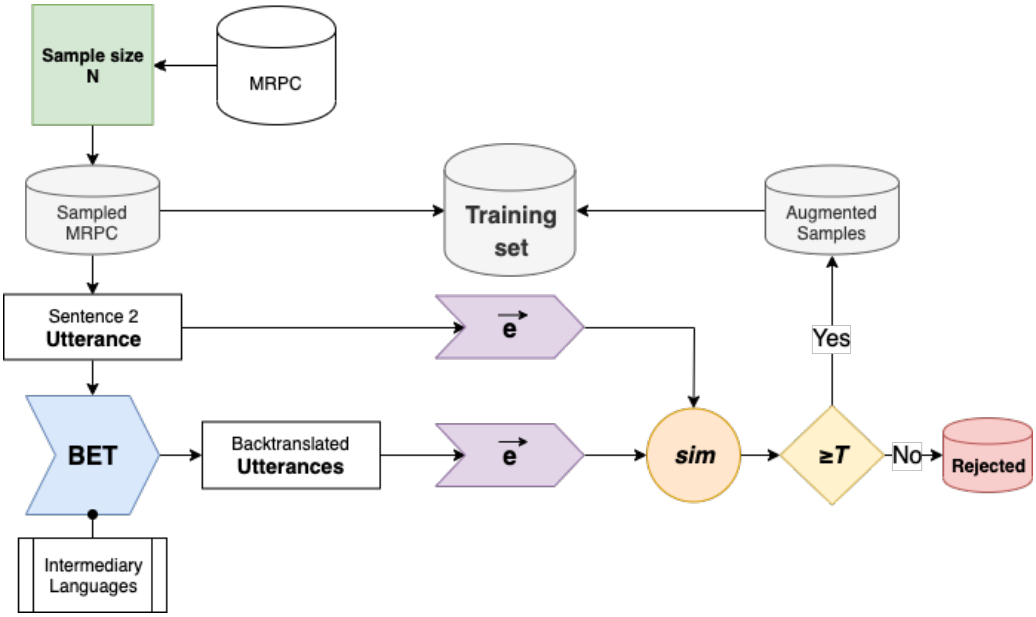

Figure 1: Experimentation pipeline schema to generate the training set from a sampled version of the MRPC — N randomly selected samples for each experiment — on top of which we add the eligible backtranslated examples. Using *sentence-transformers* $\vec{e}$ to encode the utterances as vectors, we estimate the qualification given a threshold T by measuring the cosine similarity *sim* between the generated sentences and the original ones. The utterance named *Sentence 2* corresponds to the column name inside the MRPC dataset, which is identified as the paraphrase.

2 Related Work

Data augmentation has been intensively explored in computer vision given its straightforward geometrical nature, especially image processing. According to Feng et al. (2021), the popular techniques in this field are cropping, flipping, and colour jittering. From a natural language processing standpoint, many authors noted that the natural language augmentation methods (NLA) either attempt to preserve the meaning and structure after the data augmentation process (Corbeil and Abdi Ghavidel, 2020; Tong et al., 2019; Coulombe, 2018; Sennrich et al., 2016; Anaby-Tavor et al., 2019; Radford et al., 2019) or to modify the tokens without taking into account the overall structure of the language (Wei and Zou, 2019; Coulombe, 2018). Feng et al. (2021) classified the techniques into the following categories:

- *Rule-Based techniques*: In these techniques, the original examples are changed (rewritten) based on a set of pre-defined rules. For instance, Wei and Zou (2019) applied random insertion, deletion, and swap on the tokens of the sentences.
- *Example interpolation techniques*: These techniques, also called *mixed sample data augmentation*, either interpolate the feature vectors (Zhang et al., 2017) or fuse the original examples into pairs (Ghiasi et al., 2020).
- *Model-Based techniques*: This set of techniques are concentrated on training models to generate diverse examples out of the original counterparts. Paraphrase generation (Sennrich et al., 2016) is a widely-known example of such techniques.

To the best of our knowledge, none of the papers in the aforementioned categories has analyzed the effect of a post-processing stage so far. Only Coulombe (2018) and Corbeil and Abdi Ghavidel (2020) highlighted the necessity for filtering out the backtranslation outputs to assess the data augmentation validity, without conducting any specific experiment to support the claim.

In the current paper, we closely set our work on the BET framework proposed by (Corbeil and Abdi Ghavidel, 2020), on top of which we enrich the meaning preserving aspect with a semantic similarity stage. Their original approach uses a model-based technique by applying backtranslation on ten intermediary languages to obtain a soft data augmentation. Thus, they generate ten times the amount of original data. Then, they analyzed the

resulting improvements on the paraphrase detection task as external validation. They tested various pre-trained models: BERT (Vaswani et al., 2017), XLNet (Yang et al., 2019), RoBERTa (Liu et al., 2019) and ALBERT (Lan et al., 2019). In this work, we carry out the experiments with the same pre-trained models adding the most recent ones: Electra (Clark et al., 2020) and DeBerta (He et al., 2020). There are still other closely related works such as (Shakeel et al., 2020), but the authors used neural network architectures such as LSTMs and CNNs through exploiting a set of hand-crafted features on the MRPC, Quora and SemEval datasets.

3 Methodology

3.1 Dataset

In this paper, we focus our experiments on the MRPC[1] corpus. This paraphrase corpus is included in the GLUE benchmark (Wang et al., 2019). It consists of a pair of sentences (sentence and paraphrase), which are pulled from online news sources. Overall, 4076 pairs were allocated to the train set and 1725 to the test set. We further split the MRPC train set into a smaller train set (90%, 3,668 pairs) and a validation set (10%, 408 pairs).

3.2 Data Augmentation Pipeline

As illustrated in Figure 1, our pipeline includes a backtranslation process based on BET using the *Google Translate* API and a filtering process using the sentence-transformers bi-encoder approach (Reimers and Gurevych, 2019).

On the basis of BET, we selected ten languages for the backtranslation procedure. These intermediary languages are: Chinese (*zh*), Spanish (*es*), Arabic (*ar*), Japanese (*ja*), Telugu (*te*), Javanese (*jv*), Korean (*ko*), Vietnamese (*vi*), Turkish (*tr*) and Yoruba (*yo*). In this regard, we augment only the paraphrases (e.g. the column *Sentence 2*) through backtranslating them into English from one of the aforementioned languages.

Our filtering module is mainly based on the sentence-transformers bi-encoder approach (Reimers and Gurevych, 2019). It is built to compute a unique sentence representation by pooling all the transformer's contextual word embeddings — applying the mean. It is optimized under cosine loss in a Siamese neural network fashion. We choose the *stsb-distilroberta-v2* model, which is a lightweight version. Formally, we note it as a function $\vec{e}(s)$ with s being the sentence to encode into a sentence embedding. Then, we calculate the cosine similarity (see equation 1) between the original sample and the backtranslated one. Finally, we opt-out the ones which are below various thresholds $T \in \{0.95, 0.9, 0.8\}$.

$$sim(s_1, s_2) = \frac{\vec{e}(s_1) \cdot \vec{e}(s_2)}{||\vec{e}(s_1)|| \cdot ||\vec{e}(s_2)||} \quad (1)$$

We show that different thresholds T influence drastically the outcome of our transformer-based paraphrase identifiers. We can approximate the effect of the semantic filtering as a paraphrase verification $para(\cdot, \cdot)$ like in equation 2. We hypothesize that, by adding this filtering stage, we can reinforce the preservation of meaning into BET up to some specific threshold T.

$$para(s_1, s_2) \approx \begin{cases} 1 & if\ sim(s_1, s_2) \geq T \\ 0 & else \end{cases} \quad (2)$$

3.3 Adjusting the Thresholds for Understanding the Similarities

In Figure 2, we present the histograms representing the distributions of similarities between the original sentence and the backtranslated ones. We displayed one histogram for each intermediary language. Giving the proximity of our setup with the original BET setup, we observe that the amount of generated examples with a similarity above 0.95 — as it is sorted in Figure 2 — correlates with the results reported by original BET experiments (Corbeil and Abdi Ghavidel, 2020). For instance, the authors mentioned that Spanish (*es*) and Vietnamese (*vi*) are among the best intermediary languages to use with BET to achieve the most gain on the performances. From our observations, we conclude that looking at similarity is a better way to analyze the impact of intermediary languages on backtranslation.

Based on those distributions, we also set the three similarity thresholds used in our experiments. We selected 0.8 because it conserves a majority of the generated data while filtering outliers. Afterwards, we chose 0.9 which is a compromise between quantity and quality. Finally, 0.95 is the strictest threshold keeping only the most similar examples. We won't extend our analysis to a threshold of 0 — equivalent to the original BET — since

[1] Microsoft Research Paraphrase Corpus

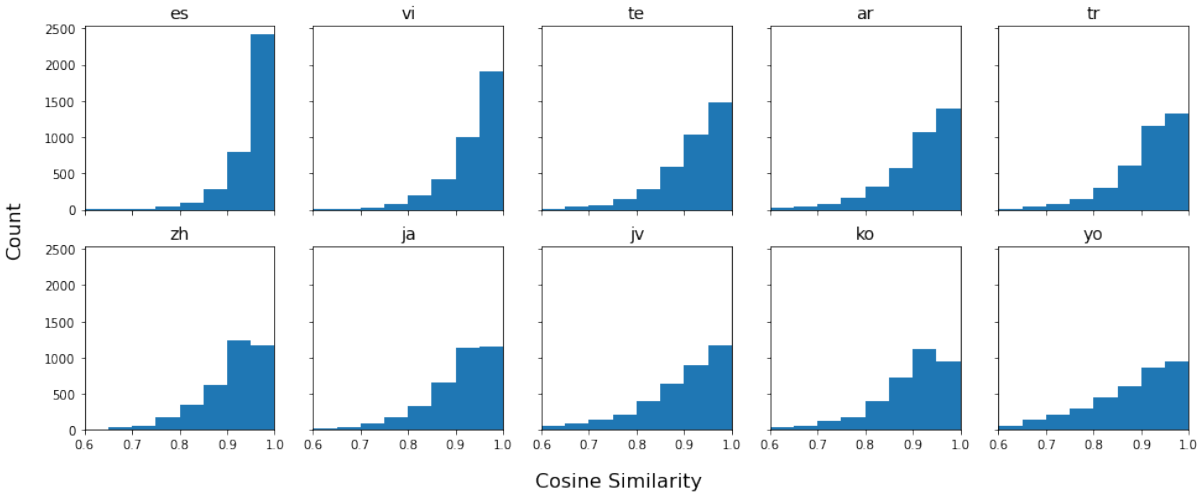

Figure 2: Similarity distributions for all intermediary languages sorted from left to right (up and down) by the amount of samples with a similarity above 0.95. The abbreviations are: Spanish (*es*), Chinese (*zh*), Arabic (*ar*), Japanese (*ja*), Telugu (*te*), Javanese (*jv*), Korean (*ko*), Vietnamese (*vi*), Turkish (*tr*) and Yoruba (*yo*)

0.8 encompasses most of the data and the rest should be only outliers.

Considering the full MRPC corpus, we further analyzed the total amount of eligible samples after applying the different similarity thresholds T in the bar chart of Figure 3. We can see that the threshold of 0.8 retains most of the generated data. For a threshold of 0.9, a majority of samples are still qualified for the training of the model. The 0.95 threshold drops less than two-thirds of the data taking only the most similar examples to the original sentence. By observing the results of the experiments in the next section, we can conclude about which of the quantity criterion ($T = 0.8$) or the quality criterion ($T = 0.95$) is better to determine the eligibility of a backtranslated text. We externally assessed this qualification by measuring the performances achieved by the models on the paraphrase detection task.

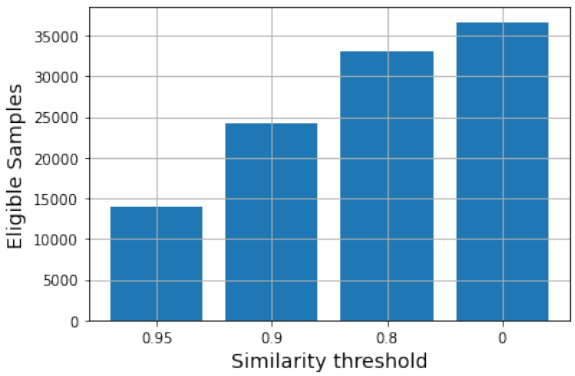

Figure 3: The total amount of eligible samples per similarity threshold. 0 corresponds to no filtering.

4 Results and Discussion

As we mentioned in section 1, we evaluated our natural language augmentation approach on BERT (Devlin et al., 2018), XLNet (Yang et al., 2019), RoBERTa (Liu et al., 2019), ALBERT (Lan et al., 2019), Electra (Clark et al., 2020) and DeBerta (He et al., 2020). We performed our experiments iteratively beginning with the sampling of only 100 random examples from the original MRPC dataset up to reaching the whole trainset. We selected ten sample sizes to cover low-data regime situations (100 - 1,000) and high-data regime situations (1,000 - 3,668). We leveraged the HuggingFace[2] *transformers* library and the *sentence-transformers* library for all our fine-tuning and filtering experiments. We fixed the training configuration to well-known hyperparameters for this task based on HuggingFace's recommendations. We left a granular optimization of the hyperparameters for future works. Thus, our experimental setup is as follows:

- Batch size: 32
- Learning rate: 3e-5
- Number of runs per experiment (random seeds were randomized at each run): 10
- Number of different experiments: 240
- Sample sizes: [100, 250, 500, 750, 1000, 1500, 2000, 2500, 3000, 3668]

[2]https://huggingface.co/

- FP16 mode

Overall, we conduct 240 unique experiments. For each of these, we report the average of 10 runs in Figures 4, 7, and 8. Our evaluation metrics are respectively: F-1 score, precision, and recall. We mainly focus on the average of F-1 scores since it is the metric used for the GLUE benchmark and in the literature. Nonetheless, we also inspect the precision and recall — the components of the harmonic mean used to compute the F-1 score — to gain a thorough understanding of our method.

In Figure 4, we illustrate the F-1 scores for all six models across all the sampling sizes. We display four curves: baseline (plain MRPC without BET), BET filtered 0.95, BET filtered 0.9, and BET filtered 0.8. We also added the F1 scores (dashed black lines) reported by their original authors for the corresponding large models. As a first general observation, we observe that all the baseline curves have approximately an S-shaped trend, in which sharp variations occur. In contrast, the BET filtered lines are smooth logarithmic-like growth, mostly all above the baseline curve. We further note that the higher we fix the similarity threshold, the bigger the gains we have. Some models like RoBERTa and DeBerta have gained between 0.04 and 0.08 in the sample size region between 500 and 1,500 samples.

We directly provided in Figure 6 the F-1 scores gain G in percent computed by comparing the BET filtered 0.95 to the baseline. We used the equation 3. We first note that most models are around 3% gain in the data scarcity region. For the Electra base model, we observe a maximum peak of 7.6% gain in F-1 score for a 750 sample size. The second highest peak is reached by Albert with nearly 5% between 500 samples and 1,500 samples. When we look at the sizes near the full dataset, we can't discern a clear portrait, despite less gain overall. We measured around 1% for RoBERTa and Electra, which lead in absolute to results near their large equivalent. Two cases are slightly below the 0% — BERT and Albert. XLNet and Deberta seem to be in between those extremums. Therefore, the trend is moving from high gains in the low-data region (250 to 2,000 samples) to lower gains at higher sample sizes (3,000 to 3,668 samples).

$$G = 100 \cdot \frac{F1_{augmented} - F1_{baseline}}{F1_{baseline}} \quad (3)$$

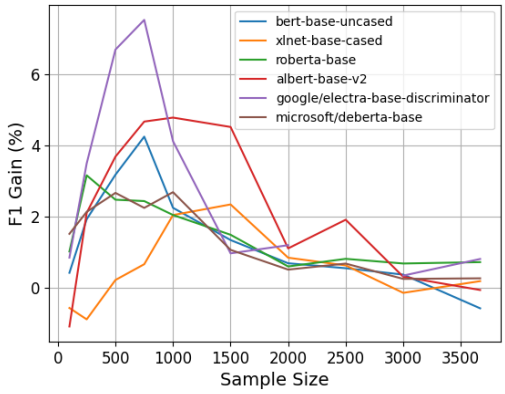

Figure 6: F-1 score gain in percent by comparing the F1-score averages between the *BET filtered 0.95* against the *Baseline*.

We also checked the p-values from the *Student's T-test* between all the augmented F-1 scores and the baseline ones, in Figure 5. In statistics, we are usually advised for a minimum of about 35 runs to benefit from the law of large numbers. Given the long training times of transformer models and the number of configurations set by our methodology, we limited our experiments to ten runs. However, we note it is already twice the usual five runs used in the literature with these models. The resulting comparison using the p-values is therefore limited. Yet, we observe that in the case of the *BET filtered 0.95* mostly all F-1 scores are strongly significant below a p-value of 0.05 (dashed black line). However, we note less significant results at 100 samples, and some at high sample sizes. Those regions, as well as the *BET filtered 0.8* and *BET filtered 0.9*, would require further runs to conclude statistically the *T-test*. We finally highlight that, in the case of RoBERTa and Electra in the large sample size region (from 3,000 to 3,668), the results reaching near their large counterparts are significant.

To have an idea of the underlying influences behind the reported F-1 scores, we also provide the values for precision and recall in Figures 7 and 8.

In Figure 7, we observe in many cases a reduction in precision. However, in the low sample sizes — below 1,500 samples —, we notice gains in precision between 0.03 and 0.15. Furthermore, we report that the lower the similarity filter is(0.8 and 0.9), the more we tend to degrade the precision compared to the baseline. We remark that all the *BET filtered 0.95* curves are surpassing the baseline precision. We mentioned as a first hypothesis that a higher threshold on the similarity scores would

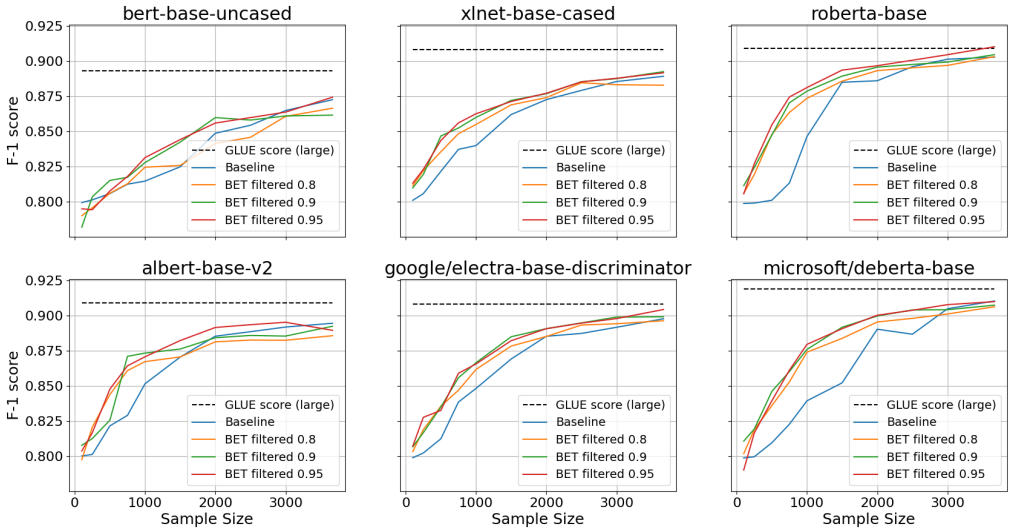

Figure 4: F-1 score curves for all experiments, where each point is the average of ten experiments. The dashed black lines are the GLUE benchmark scores reported for the large models. The model names are the ones used by the HuggingFace *Model Hub*.

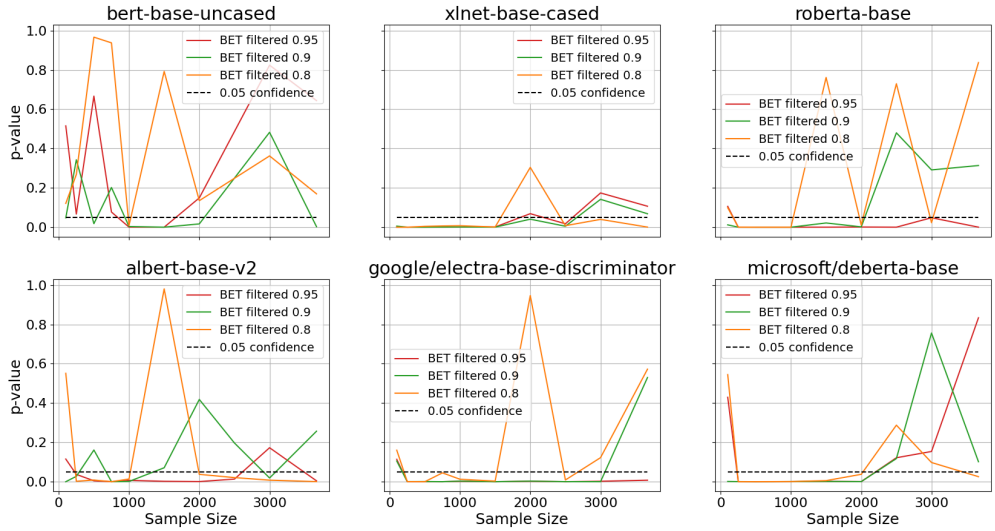

Figure 5: P-values of the F1-score curves augmented by the *BET* framework against the *Baseline* curve.

induce a higher quality of the generated samples — leading logically to a rise in precision. Therefore, we confirm the validity of this hypothesis based on its impact on the precision curves.

In Figure 8, we show the sensitivity curves, on which we denote two observations. First, as expected generally with data augmentation in NLP, we note an overall gain in recall when applying BET from a couple of percent up to 0.05. We observe that this gain tends to lower as the similarity threshold gets higher, but remains above the baseline. The pre-trained models that benefit the most in terms of sensitivity are respectively BERT, DeBerta, Albert and RoBERTa. XLNet and Electra obtained very low improvements on the sensitivity metric. When looking below a 1,000 sample sizes, we notice a drastic drop in recall from the baseline to any of the BET curves. Nonetheless, we rationalize that the models tend to declare a paraphrase too often. We conclude that this issue is solved by applying any backtranslation.

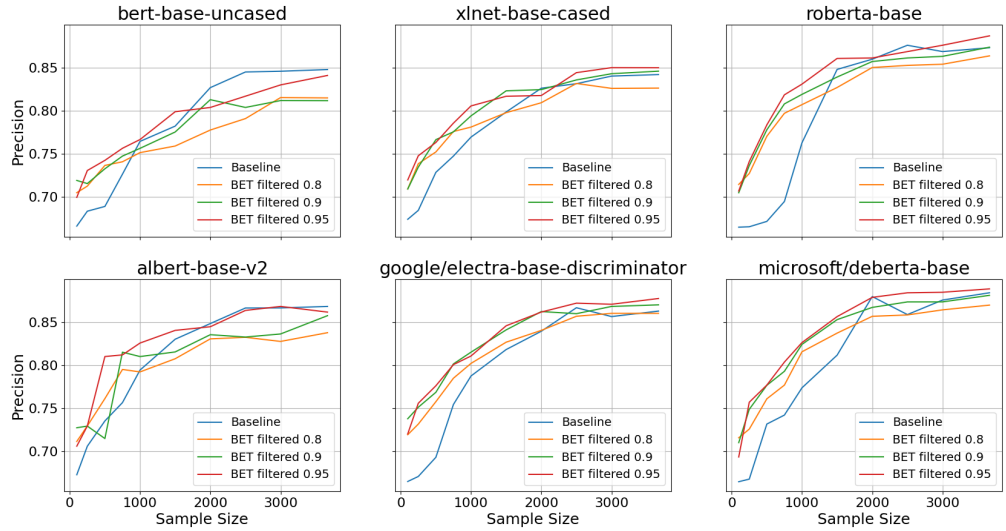

Figure 7: Precision curves for all experiments, where each point is the average of ten experiments.

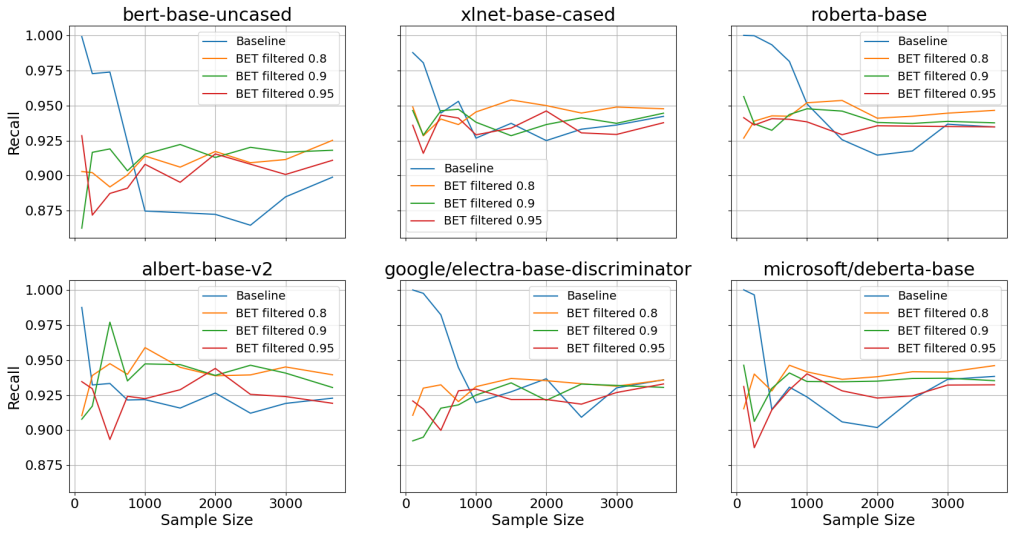

Figure 8: Recall curves for all experiments, where each point is the average of ten experiments.

5 Conclusion and Future Work

In this paper, we described a method based on back-translation which is followed by a filtering stage to keep the most eligible examples. We increased the F-1 scores on the automatic paraphrase detection task by up to 7.6% compared to the baseline using only a fraction of the original dataset. Furthermore, we demonstrated that this approach limits the gain in recall while avoiding degrading the precision, which results in the best F-1 scores. With the augmentation of the full dataset using RoBERTa base and Electra base, we achieved results that are close to the reported GLUE benchmark scores, while the original authors were using their corresponding large versions. In conclusion, pre-trained transformer models have very good transfer-learning capabilities, but they still largely benefit from the support of high-quality natural language augmentation, both to enrich very small datasets and to alleviate the overfit on surface cues.

In future work, we will extend this work to the other paraphrase corpus as well as to the other NLP tasks such as multi-class classification.

References

Ateret Anaby-Tavor, Boaz Carmeli, Esther Goldbraich, Amir Kantor, George Kour, Segev Shlomov, Naama Tepper, and Naama Zwerdling. 2019. Not enough data? deep learning to the rescue! *arXiv preprint arXiv:1911.03118*.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Jean-Philippe Corbeil and Hadi Abdi Ghavidel. 2020. Bet: A backtranslation approach for easy data augmentation in transformer-based paraphrase identification context. *arXiv preprint arXiv:1907.12412*.

Claude Coulombe. 2018. Text data augmentation made simple by leveraging nlp cloud apis. *arXiv preprint arXiv:1812.04718*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Steven Y. Feng, Varun Gangal, Jason Weiy, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy1. 2021. A survey of data augmentation approaches for nlp. *arXiv:2105.03075*.

Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D. Cubuk, Quoc V. Le, and Barret Zoph. 2020. Simple copy-paste is a strong data augmentation method for instance segmentation. *arXiv:2012.07177*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. 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)*, pages 86–96.

Muhammad Haroon Shakeel, Asim Karim, and Imdadullah Khan. 2020. A multi-cascaded model with data augmentation for enhanced paraphrase detection in short texts. *Information Processing & Management*, 57(3):102204.

Yixuan Tong, Liang Liang, Boyan Liu, Shanshan Jiang, and Bin Dong. 2019. Supervised neural machine translation based on data augmentation and improved training\& inference process. In *Proceedings of the 6th Workshop on Asian Translation*, pages 147–151.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding*. In the Proceedings of ICLR.

Jason W Wei and Kai Zou. 2019. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. *arXiv preprint arXiv:1901.11196*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5754–5764.

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. 2017. mixup: Beyond empirical risk minimization. *arXiv:1710.09412*.