Mr-Fosdick at SemEval-2023 Task 5: Comparing Dataset Expansion Techniques for Non-Transformer and Transformer Models: Improving Model Performance through Data Augmentation

Christian Falkenberg and **Erik Schönwälder** and **Tom Rietzke**
Chris-Andris Görner and **Robert Walther** and **Julius Gonsior** and **Anja Reusch**
Technische Universität Dresden
first.last@mailbox.tu-dresden.de

Abstract

In supervised learning, a significant amount of data is essential. To achieve this, we generated and evaluated datasets based on a provided dataset using transformer and non-transformer models. By utilizing these generated datasets during the training of new models, we attain a higher balanced accuracy during validation compared to using only the original dataset.

1 Introduction

Titles of posts are increasingly utilizing exaggerations and other persuasive techniques to entice clicks. Such titles like the following are known as clickbait:

'Stranger Knocked Angrily on Her Door – What He Left Made Her Run to the Phone'

The Clickbait Challenge, hosted on Tira (Fröbe et al., 2023b), is a shared task that aims to address and satisfy the curiosity generated by this phenomenon. The challenge comprises two subtasks that utilize the post titles featuring clickbait and their corresponding articles as input. In the first task, three types of spoilers are to be distinguished, while the second task involves retrieving the spoiler from the article. For a detailed description of the tasks, please refer to the Overview Paper (Fröbe et al., 2023a) or the official Clickbait Challenge website (Clickbait).

We concentrated on optimising the given dataset to provide better training data for the models we created. In the following, we will describe our approaches to obtaining new datasets. We then evaluate our generated datasets and compare them to the original datasets. Our approaches for task 1, clickbait classification, using a non-transformer and a transformer-based approach are then described. After that we present our model for task 2 and show how we retrieve the spoilers for the clickbait titles.

2 Expanding the Dataset

2.1 Dataset Generation

The original dataset provided to us (Hagen et al., 2022) contained 3200 examples of training data. All examples consisted of a clickbait post, the article text, the spoiler type and the corresponding spoiler. Table 1 presents an overview of all utilized and created datasets, including their corresponding source file sizes.

2.1.1 QA Dataset

Our initial approach was to use existing data and format it for our purposes. For example, question answering datasets (QA) are appropriate for the task at hand. In QA datasets, the question can be understood as PostText and the answer as a spoiler. The length of the spoiler then in turn determines the spoiler type. To create our own suitable **MixQA** (MixQA Dataset) dataset, several QA datasets [1] were combined. The largest and best known dataset in this series is the SQuAD2.0 (Rajpurkar et al., 2018) dataset.

2.1.2 Pegasus Dataset

Data generation through paraphrasing is a process where new, diverse and meaningful samples are generated from existing data by changing the wording or sentence structure.
One approach we tried to generate new data was Pegasus (Zhang et al., 2019), which is a transformer-based language generation model developed by Google. We used a pretrained Pegasus model (Pegasus Paraphrasing) fine-tuned on the paraphrasing task. The model is given a post title and generates a different but semantically similar title based on that. 16000 samples were generated, which together form the **Pegasus** (Pegasus Dataset) dataset.

[1] (Joshi et al., 2017) (Su et al., 2016) (Talmor and Berant, 2018) (Rajpurkar et al., 2018)

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 88–93

Dataset	Description	Quantity	Source
Original	Provided train split without validation split	3200	Hagen et al. 2022
MixQA	Combination of different QA datasets	264699	MixQA Dataset
Pegasus	Paraphrasing with Pegasus	16000	Pegasus Dataset
Parrot	Paraphrasing with Parrot	6652	Parrot Dataset
Q_Gen	Clickbait generation with question generation model	6321	Q-Gen Dataset
GPT	GPT reformulation	2324	GPT Dataset
GPT243_S	GPT reformulation & manual selection	3434	GPT234_S Dataset

Table 1: Overview of the datasets. MixQA and GPT243_S files contain the original dataset.

2.1.3 T5 Parrot Dataset

Like Pegasus, T5 (Text-to-Text Transfer Transformer) is also a deep learning based natural language processing model developed by Google Research. Through its text-to-text architecture, T5 can be easily fine-tuned for specific tasks without having to adjust the input or output layers. The Parrot model (Damodaran, 2021) we used is based on T5 and fine-tuned for paraphrasing short sentences with less than 32 tokens. These around 6600 samples give us the **Parrot** (Parrot Dataset) dataset.

2.1.4 Clickbait Generation Dataset

In the given training dataset, 84% of samples contained a so called "humanSpoiler". These are spoilers that were not extracted but handwritten. We tried using them to generate completely new titles from the paragraphs and spoilers with a question generation model. Those kind of models take in a context and the corresponding answer and generate a question from that. With this model (Romero, 2021) we were able to generate new clickbait titles based on the context and the spoiler. These around 6300 generated samples form the **Q_Gen** (Q-Gen Dataset) dataset.

2.1.5 GPT Dataset

Another approach for generating data was the utilization of GPT-3 (Brown et al., 2020), one of the most powerful language models available. More specifically, OpenAI's Babbage GPT-3 model was used with the following prompt: *'Reformulate the following sentence as a clickbait headline:'*. The dataset generated was named **GPT** (GPT Dataset). A sample of 1000 data points were examined and selected by human reviewers. The re-formulated post title would be selected if it keeps most of the features of the original (numbers, key nouns etc.) and still points in the general direction of the desired answer. This resulted in a final dataset of 234 instances, named **GPT234_S** (GPT234_S Dataset).

2.2 Dataset Evaluation

Our aim was to identify and provide the final model with the best dataset.

To evaluate the performance of the various datasets for transformers, the DeBERTa-Large (He et al., 2021) model was fine-tuned using a single output layer. The results are presented in Table 2 and visualized in Appendix 2, which shows the difference from the baseline of the datasets. Due to the class imbalance present in all datasets, we focused on balanced accuracy as it provides a more accurate representation of the model's performance.

The results indicate that excessive amounts of artificially generated or augmented data can be detrimental to the model accuracy, as shown by the GPT dataset where a limited number of data points had a positive effect, but as the number of data points increased, the effect became negative.

This suggests that the transformer approach is highly susceptible to changes in the dataset composition, even when generated data constitutes only a small portion of the complete training data. This implies that it is crucial to be very selective while incorporating generated data and to carefully consider the proportion of generated data in comparison to the original data. Overall, the GPT and Parrot datasets performed well, with GPT achieving the best results and being further improved through the manually selected GPT234_S dataset.

Notably, a lot of improvement was observed without the need for human post-selection.

2.3 Outliers and Uncertainty

After initial analysis, the provided dataset appeared to have outliers that we aimed to remove. Outliers have a correlation to uncertainty and can often be identified with uncertainty quantification meth-

Model	Quantity	Median Δ%	Mean Δ%	StdDev%	Runs
GPT234_S	234	2.1304	1.146	3.6152	20
GPT	234	1.2790	0.6050	3.2725	20
GPT	500	0.8823	0.1905	3.1006	10
GPT	1000	-4.7053	-4.7645	3.3851	10
GPT	2000	-6.8159	-6.5558	2.0544	10
MixQA	500	0.0032	0.1071	3.1783	10
Parrot	500	0.8302	0.6798	1.6347	10
Pegasus	500	0.3311	-0.0095	3.1144	10
Q_Gen	500	-0.7557	-1.4822	2.8471	10

Table 2: Balanced accuracy difference from baseline (42 Runs) for all datasets. Quantity signifies the amount of new data utilized in addition to the provided train split.

ods (Gonsior et al., 2022). We decided to use an Ensemble with the same model architecture as the dataset evaluation, trained on the provided validation data only, to evaluate the various training data. Uncertainty was measured by the logits deviation of each data point produced by the Ensemble. The uncertainty was normalised on the provided train split and visualised with the other datasets in Figure 1.

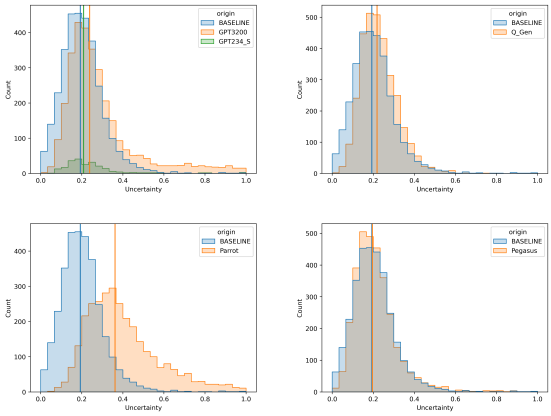

Figure 1: Normalized Uncertainty distributions.

The data with the largest uncertainty, i.e. the suspected potential outliers, were removed and not used for training. This approach had no positive effect on the provided training split. There was either no negative effect of the outliers or they were not correctly detected by the uncertainty detection. Nevertheless, the uncertainty distributions showed how different the datasets are from the original and that similar datasets to the original like GPT234_S and more distant ones like Parrot can both work.

3 System Overview and Results

3.1 Task 1 Random Forest

We initially avoided using a transformer model and chose a Random Forest using the framework scikit-learn (Pedregosa et al., 2011). The extracted features for the Random Forest can be divided into 4 groups: general features, the TfIdf matrix of the PostText and the targetTitle, information from the Natural language processing with SpaCy (Montani et al., 2022) and regex patterns.

The general features were values like the largest number in the postText or the number of words. The Natural Language Processing features were based on the tagger labels, the parser labels and the Named Entity Recognition labels. We counted the number of tokens with the respective labels as well as the number of combinations of the labels. We also extracted the tense of the verbs as a feature. With the regex patterns we extracted word combinations like a number followed by "things" or “will ... you” as a feature.

With all these features we train 3 Complement Naive Bayes Classifiers (Rennie et al., 2003), each of which classifies a post into phrase or not, passage or not and multi or not. Each input feature is scaled by its maximum absolute value. The results of these 3 classifiers also serve as input for the forest.

We feed the 18011 features as input to a Random Forest Classifier. The Random Forest consists of 100 decision trees and measures the quality of a split using the Gini impurity.

We utilized this method to evaluate our datasets by training 100 Random Forests for each dataset and determining the average balanced accuracy score. By exclusively utilizing the provided train split, we achieved an average balanced accuracy of 58.6% during validation. We observed that in-

corporating the provided train split, Pegasus, Parrot, and GPT243_S datasets resulted in an improved average validation balanced accuracy score of 59.5%. On the validation dataset, our most successful model obtained a balanced accuracy of 61.6%, while on the test dataset, it achieved a balanced accuracy of 57.8%.

Interesting to note is that compared to the transformer approach, the Random Forest model exhibited significantly more robustness when dealing with new data and did not exhibit negative effects from the incorporation of a large amount of new data.

Moreover, Random Forest has a notable advantage in terms of transparency, as it enables us to extract feature importance or output the tree structure. In addition, compared to transformer-based approaches, Random Forest is significantly less computationally expensive.

3.2 Task 1 Transformer

For our final transformer model two dropout layers, one normalization layer and two linear layers were added to the pretrained DeBERTa-Large (He et al., 2021) model. As a result of the dataset evaluation, this model employed our best performing GPT234_S dataset. By utilizing the provided train split alone, we were able to attain a validation accuracy of 73.25%. After incorporating the GPT234_S dataset, we achieved a better validation accuracy of 75.125%. Our dataset contributed to achieving a balanced test accuracy of 69.1% in the end.

3.3 Task 2 Transformer

Deepset's deberta-v3-large-squad2 (Deepset, 2022) model was fine-tuned using the extractive Question Answering Trainer architecture developed by HuggingFace (Wolf et al., 2020) with a slight modification in the post-processing hyper-parameter, extending the maximum acceptable answer length. In the example code available on Colab provided by HuggingFace, the value of max_answer_length was modified from 30 to 50 in the postprocess_qa_predictions function. The aim, similar to task 1, was to enhance results by experimenting with different datasets. However, the evaluation presented in Table 3 revealed that all datasets tested had a negative impact on overall performance, including those that were successful in task 1 such as GPT234_S and Parrot. As a result, the final model was trained on the original provided dataset without using the tag attribute, leading to an improvement in validation BLEU-score from the given transformer task 2 baseline of 0.382 to 0.448. The positive effect arises only from the Deepset transformer and the hyper-parameter settings. The reason for lower performance with data augmentation is uncertain and requires further investigation. One potential explanation is that the strategies used may not be appropriate for task 2, resulting in poor quality training data due to their dissimilarity from the original clickbait.

Model	Quantity	BLEUΔ	Runs
GPT234_S	234	-1.72	3
Parrot	500	-1.68	1
Pegasus	500	-1,76	1
Q_Gen	500	-2.23	1

Table 3: BLEU-score difference from baseline.

4 Conclusion

Throughout this project, we have experimented with various methods to retrieve spoilers from clickbait titles and post texts in order to close the curiosity gap. These methods included a non-transformer based Random Forest, a custom DeBERTa-based model for task 1, and another transformer model for task 2. However, our primary focus was not on the individual models themselves, but on generating datasets to enhance model robustness and improve the accuracy of our results. By utilizing datasets generated from converting existing question-answering datasets, paraphrasing, question generation, and GPT reformulation, we were able to demonstrate significant improvements in the proposed approaches for task 1. Unfortunately, for unknown reasons, we did not observe any improvements in task 2 with the generated data. Instead, our task 2 results improved through fine-tuning hyperparameters in the transformer-based approach we described.

We demonstrated that the transformer approaches are extremely sensitive to even small amounts of newly generated data, whereas the Random Forest model could handle large amounts of new data with ease while still reaping benefits. Furthermore, our uncertainty analysis demonstrated that both more conservative and imaginative approaches to data generation can prove to be effective.

We are of the opinion that our results can be further enhanced by integrating more diverse data

generation techniques and better post-selection of the generated data, perhaps even through automatic means.

Limitations

First, we acknowledge that our dataset expansion efforts focused exclusively on a single dataset, which may have inherent biases due to its limited data sources. Although some of our expansion techniques yielded improved model performance on this dataset, it remains unclear whether these gains will generalize to other datasets. Moreover, the size of the expanded dataset was determined empirically and was only tested on this one dataset. It is possible that additional data augmentation could further improve model performance or that certain expansion methods are particularly suited to this dataset. For example, it is conceivable that GPT3 may be better suited for clickbait augmentation due to its training on internet data. Additionally, we cannot account for the differential performance between the two tasks employed in this study. While task 1 classification performed well, task 2 retrieval did not, and the reasons for this discrepancy are unclear. Future studies should examine the suitability of our dataset expansion techniques on a variety of sources, and consider the composition of a representative final training set for specific tasks.

Ethics Statement

In this paper, we augmented the original dataset with the intention of improving the performance of the models trained on it. The use of these models is outside of our control, but our purpose for augmenting the data was solely to enhance model performance and did not have any malicious intent. Our trained models were only used to address two tasks. Spoiling clickbait or news posts in general may discourage users from engaging with or seeking out diverse perspectives and sources of information. While some posts have short and definitive answers, some are ambivalent and need context, which can get lost in the retrieved spoilers. The datasets generated and used in this paper may contain inaccuracies or harmful content that have not been reviewed by humans. Training models on this data may result in incorrect understanding and incorrect outputs.

Acknowledgements

The authors gratefully acknowledge the GWK support for funding this project by providing computing time through the Center for Information Services and HPC (ZIH) at TU Dresden.

References

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Clickbait. 2023. Clickbait challenge at semeval 2023 - clickbait spoiling. Accessed 31-01-2023.

Prithiviraj Damodaran. 2021. Parrot: Paraphrase generation for nlu.

Deepset. 2022. Deberta-v3-large-squad2. Accessed 31-01-2023.

Maik Fröbe, Tim Gollub, Benno Stein, Matthias Hagen, and Martin Potthast. 2023a. SemEval-2023 Task 5: Clickbait Spoiling. In *17th International Workshop on Semantic Evaluation (SemEval-2023)*.

Maik Fröbe, Matti Wiegmann, Nikolay Kolyada, Bastian Grahm, Theresa Elstner, Frank Loebe, Matthias Hagen, Benno Stein, and Martin Potthast. 2023b. Continuous Integration for Reproducible Shared Tasks with TIRA.io. In *Advances in Information Retrieval. 45th European Conference on IR Research (ECIR 2023)*, Lecture Notes in Computer Science, Berlin Heidelberg New York. Springer.

Julius Gonsior, Christian Falkenberg, Silvio Magino, Anja Reusch, Maik Thiele, and Wolfgang Lehner. 2022. To softmax, or not to softmax: that is the question when applying active learning for transformer models.

GPT Dataset. 2023. Gpt dataset. Accessed 22-02-2023.

GPT234_S Dataset. 2023. Gpt234_s dataset. Accessed 18-02-2023.

Matthias Hagen, Maik Fröbe, Artur Jurk, and Martin Potthast. 2022. Clickbait Spoiling via Question Answering and Passage Retrieval. In *60th Annual Meeting of the Association for Computational Linguistics (ACL 2022)*, pages 7025–7036. Association for Computational Linguistics.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

HuggingFace. n.d. Question answering with hugging face transformers. https://colab.research.google.com/github/huggingface/notebooks/blob/main/examples/question_answering.ipynb. Accessed 18-02-2023.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada. Association for Computational Linguistics.

MixQA Dataset. 2023. Mixqa dataset. Accessed 18-02-2023.

Ines Montani, Matthew Honnibal, Matthew Honnibal, Sofie Van Landeghem, Adriane Boyd, Henning Peters, Paul O'Leary McCann, jim geovedi, Jim O'Regan, Maxim Samsonov, Duygu Altinok, György Orosz, Daniël de Kok, Søren Lind Kristiansen, Raphaël Bournhonesque, Madeesh Kannan, Lj Miranda, Peter Baumgartner, Edward, Explosion Bot, Richard Hudson, Roman, Leander Fiedler, Raphael Mitsch, Ryn Daniels, Grégory Howard, Wannaphong Phatthiyaphaibun, Yohei Tamura, and Sam Bozek. 2022. explosion/spaCy: v3.4.3.

Parrot Dataset. 2023. Parrot dataset. Accessed 18-02-2023.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pegasus Dataset. 2023. Pegasus dataset. Accessed 18-02-2023.

Pegasus Paraphrasing. 2021. Pegasus fine-tuned for paraphrasing. Accessed 31-01-2023.

Q-Gen Dataset. 2023. Q-gen dataset. Accessed 18-02-2023.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad.

Jason D. M. Rennie, Lawrence Shih, Jaime Teevan, and David R. Karger. 2003. Tackling the poor assumptions of naive bayes text classifiers. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, ICML'03, page 616–623. AAAI Press.

Manuel Romero. 2021. T5 (base) fine-tuned on squad for qg via ap. https://huggingface.co/mrm8488/t5-base-finetuned-question-generation-ap.

Yu Su, Huan Sun, Brian Sadler, Mudhakar Srivatsa, Izzeddin Gür, Zenghui Yan, and Xifeng Yan. 2016. On generating characteristic-rich question sets for QA evaluation. In *Conference on Empirical Methods in Natural Language Processing*, Austin, Texas, USA. Association for Computational Linguistics.

Alon Talmor and Jonathan Berant. 2018. The web as a knowledge-base for answering complex questions. In *North American Chapter of the Association for Computational Linguistics*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2019. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization.

A Appendix

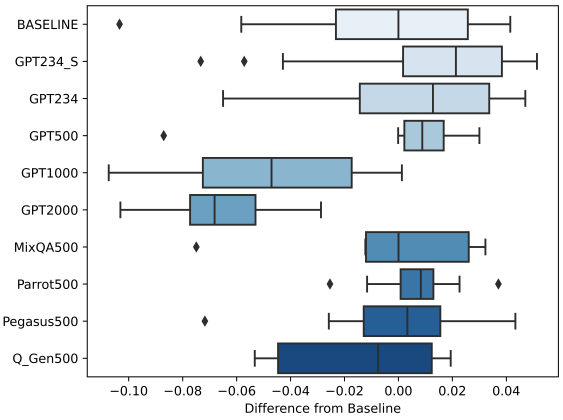

Figure 2: Boxplot visualization of Table 2.