Gallagher at SemEval-2023 Task 5: Tackling Clickbait with Seq2Seq Models

Tugay Bilgis, Nimet Beyza Bozdag, Steven Bethard
University of Arizona
{tbilgis, nbbozdag, bethard}@arizona.edu

Abstract

This paper presents the systems and approaches of the Gallagher team for the SemEval-2023 Task 5: Clickbait Spoiling. We propose a method to classify the type of spoiler (phrase, passage, multi) and a question-answering method to generate spoilers that satisfy the curiosity caused by clickbait posts. We experiment with the state-of-the-art Seq2Seq model T5. To identify the spoiler types we used a fine-tuned T5 classifier (Subtask 1). A mixture of T5 and Flan-T5 was used to generate the spoilers for clickbait posts (Subtask 2). Our system officially ranks first in generating phrase type spoilers in Subtask 2, and achieves the highest precision score for passage type spoilers in Subtask 1.

1 Introduction

The goal of SemEval-2023 Task 5 is to spoil clickbait. Clickbait posts are texts that arouse curiosity not by providing informative summaries of articles, but by purposefully teasing and leaving out key information from an article to advertise a web page's content. The aim of this shared task is to generate short texts (spoilers) that satisfy the curiosity induced by clickbait posts and provide more informative summaries of the linked articles. Although the first subtask may seem unnecessary, our results suggest that it could be highly helpful for the main purpose of spoiler generation. As clickbait posts are getting more common every day, this task holds more importance than ever. This task is only aimed towards English clickbait posts (Fröbe et al., 2023a).

Our strategy for this shared task was to approach it as a question-answering problem. Our system uses the state-of-the-art sequence-to-sequence (Seq2Seq) model T5 (Raffel et al., 2019). We experiment with different variants of T5, some of which were previously fine-tuned for question-answering tasks. We implement a different model for each spoiler category type due to the contrast between the different types of output sequences.

Our system achieves competitive results in both subtasks compared to the baseline from the shared task organizers and to the other teams. It has the highest precision for classifying the passage type spoilers in the spoiler classification subtask, and ranks as the top submission for phrase type spoilers in the generation subtask. However, our system struggles with generating longer spoilers such as phrase and multi type spoilers.

We submitted our system as a docker image on the TIRA platform (Fröbe et al., 2023b) to increase reproducibility. We release our code [1] and the model checkpoints [2].

2 Background

2.1 Related Work

A limited number of studies have directly focused on the task of clickbait spoiling. The most extensive work comes from Hagen et al. (2022), the organizers of this shared task. In their study, they approach clickbait spoiling as a question-answering and passage retrieval task. They use the question-answering method for phrase spoilers and the passage retrieval method for passage spoilers. The passage retrieval method is a relaxed question-answering method, where it allows the answer to be a longer sequence of text. They experiment with a variety of Transformer models. However, their work does not include spoiler generation for multi type spoilers as they acknowledge that their methods would not work well with multi type spoilers.

The work of Heiervang (2022) focused on spoiling clickbait posts that they collected from the Reddit forum "Saved You a Click", shortened as SYAC. They experiment by fine-tuning the T5 and the UnifiedQA (Khashabi et al., 2020) models on their

[1] https://www.github.com/tbilgis23/clickbait-spoiling

[2] https://www.huggingface.co/Tugay

Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 1650–1655
July 13-14, 2023 ©2023 Association for Computational Linguistics

	Phrase	Passage	Multi	Total
Train	1367	1274	559	3200
Validation	335	322	143	800

Table 1: Distribution of spoiler types in the train and validation set

Reddit SYAC dataset and introduced a new method called “Title Answering”, which achieves an overall BLEU-4 score of 0.1746 on the test set.

2.2 Data

The data provided is split into three sets; train, validation, and test. There are 3200 posts in the train set, 800 in the validation set, and 1000 in the test set, for a total of 5000 clickbait posts. These clickbait posts are collected from five different social media accounts and were manually spoiled (Hagen et al., 2022). The test set is hidden from the participants and only used by the organizers to evaluate the systems. Spoilers are categorized into 3 types: short phrase spoilers, longer passage spoilers, and multiple non-consecutive pieces of text. Table 1 provides the distribution of the spoiler types in the train and validation set.

The data for the shared task is provided in JSON format, where each clickbait post is a JSON object and has several fields describing the clickbait post. Descriptions of those fields can be found on the task description website[3]. The `tags` field contains the output for the spoiler classification subtask and the `spoiler` field contains the output for the spoiler generation subtask. These output fields are not provided in the test set.

3 System Overview

We used the T5 model (Raffel et al., 2019) for both the spoiler classification and spoiler generation tasks. T5 is a state-of-the-art Seq2Seq model and has shown successful performance for a variety of NLP tasks. It is a text-to-text framework, where the input is fed as a text and a target text is generated for the output.

Although T5 works for a variety of tasks out-of-the-box, as it was trained on a large corpus with a mixture of supervised and unsupervised tasks, we did additional training to fine-tune it for the specific needs of the clickbait tasks. We also experimented with variants of T5 and models that were fine-tuned for specific tasks such as question answering.

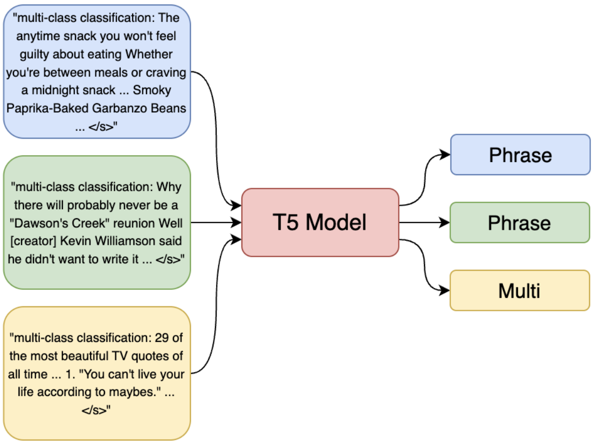
"multi-class classification: The anytime snack you won't feel guilty about eating Whether you're between meals or craving a midnight snack ... Smoky Paprika-Baked Garbanzo Beans ... </s>"
"multi-class classification: Why there will probably never be a "Dawson's Creek" reunion Well [creator] Kevin Williamson said he didn't want to write it ... </s>"
"multi-class classification: 29 of the most beautiful TV quotes of all time ... 1. "You can't live your life according to maybes." ... </s>"
T5 Model
Phrase
Phrase
Multi

Figure 1: Visual representation of our system for Subtask 1

3.1 Subtask 1: Spoiler Classification

The task of spoiler classification is a first step towards the main goal of spoiler generation, allowing different approaches to be used in the generation of different spoiler types. We experimented with both T5 and LongT5 (Guo et al., 2021), as some of the passages exceeded the maximum sequence length of 512 and thus could potentially lose information needed to classify the spoiler type. We take advantage of the larger checkpoints of T5 and LongT5 with 770 Billion parameters (T5$_{LARGE}$ and LongT5$_{LARGE}$).

For both models, we follow the approach of Hagen et al. (2022) of feeding the clickbait post and the article of the post as the input. Additionally, we add to each input a prefix of `multi-class classification:`, as T5 performs well with task prefixes (Raffel et al., 2019), and a suffix of `</s>`, the string ending token. Figure 1 illustrates our approach for this subtask.

A potential challenge in this classification task is data imbalance. As seen in Table 1, the number of multi type spoilers is less than half that of phrase or passage type spoilers. However, the multi type spoilers usually contain a quantitative element in the clickbait post and in the article (as seen in Figure 1), which potentially could make it easier to classify the multi type spoilers.

3.2 Subtask 2: Spoiler Generation

Spoiler generation was the main focus of the shared task. We approach this as a question-answering

[3] https://pan.webis.de/semeval23/pan23-web/clickbait-challenge.html

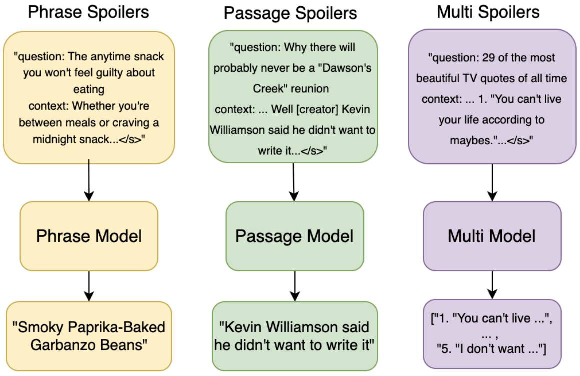
Phrase Spoilers
Passage Spoilers
Multi Spoilers
"question: The anytime snack you won't feel guilty about eating
context: Whether you're between meals or craving a midnight snack...</s>"
"question: Why there will probably never be a "Dawson's Creek" reunion
context: ... Well [creator] Kevin Williamson said he didn't want to write it...</s>"
"question: 29 of the most beautiful TV quotes of all time
context: ... 1. "You can't live your life according to maybes."...</s>"
Phrase Model
Passage Model
Multi Model
"Smoky Paprika-Baked Garbanzo Beans"
"Kevin Williamson said he didn't want to write it"
["1. "You can't live ...",
... ,
"5. "I don't want ..."]

Figure 2: Visual representation of our system for Subtask 2

problem, similar to the approach of Hagen et al. (2022). Each type of spoiler differs in its structure, therefore we implemented different models for each. Along with the T5 model, we explore several other scaled and fine-tuned T5 models for each spoiler type.

For all the spoiler types, we give the clickbait post and the article as the input, in a format similar to the SQUAD dataset (Rajpurkar et al., 2016), where the clickbait post has the `question:` prefix, the article has the `context:` prefix, and `</s>`, the string ending token, follows the article. We give the corresponding spoilers as the output. Figure 2 illustrates our approach to this subtask.

3.2.1 Phrase Spoiler Generation

Phrase type spoilers are spoilers for clickbait posts that can be spoiled in either a single word or a couple of words that occur together in the article. Hence, we approach it as an extractive question-answering problem.

We experimented with fine-tuning the T5$_{BASE}$, T5$_{LARGE}$, UnifiedQA$_{LARGE}$, Flan-T5$_{LARGE}$ (Chung et al., 2022), and LongT5$_{LARGE}$ models. UnifiedQA is a T5 model that has been trained on various question-answering formats, combined into a single model.

The input is fed in the SQUAD format for T5, Flan-T5, and LongT5 models without any additional pre-processing of the data. However, for the UnifiedQA model, we follow the recommended pre-processing step for the input by giving no prefixes and separating the clickbait post and the article with the `\n` separator (Khashabi et al., 2020).

3.2.2 Passage Spoiler Generation

Passage type spoilers are slightly more complex than phrase spoilers, as the spoilers are full sentences rather than words. We follow a similar approach to phrase spoiler generation and treat it as an extractive question-answering problem. However, we use a separate model from the model of the previous section since fine-tuning with both phrase and passage spoilers could confuse the model.

We experimented with fine-tuning the T5$_{LARGE}$, UnifiedQA$_{LARGE}$, Flan-T5$_{LARGE}$, and LongT5$_{LARGE}$ models. The input was given in the SQUAD format for T5, Flan-T5, and LongT5. For the UnifiedQA model, the special pre-processing described in Section 3.2.1 was applied to the input.

3.2.3 Multi Spoiler Generation

Multi type spoilers are spoilers that spoil multiple non-consecutive texts from the clickbait post's article. It is more complex than the phrase or passage spoiler generation as it requires the model to pay attention to multiple parts of the article rather than a single part. The output should be the first five spoilers if there are more than 5 spoilers in the context, as the organizers assert that if 5 can be found correctly, others would be too. We approach this problem as a question-answering problem mixed with summarization for the multi-span extraction. This was the most challenging part of this shared task, as there hasn't been a lot of previous work done in this area and the data for multi spoiler type was limited.

We only experimented with T5$_{LARGE}$ and UnifiedQA$_{LARGE}$ models for the multi spoiler generation due to time and resource constraints. The input format was the same as other spoiler types: SQUAD format for the T5 model and the special preprocessing described in Section 3.2.1 was applied to the input of UnifiedQA.

4 Experimental Setup

We used the provided training data only to train our models, and the provided validation split only to evaluate the models. We did not use the validation split to further train our models. No preprocessing is done to the shared task data other than adding the subtask-specific prefixes mentioned in Sections 3.1 and 3.2. We used the Huggingface Transformers library (version 4.6.0) to train our models.

We used the AdamW optimizer with an initial learning rate of $5 \times e^{-5}$ and with a weight decay of 0.01 during the training for all models. For the spoiler classification subtask, we trained our models for 10 epochs and with a batch size of 8,

Model	Balanced Accuracy
T5$_{LARGE}$	0.75
LongT5$_{LARGE}$	0.41
Fröbe et al. (2023a)	0.734

Table 2: Results on the validation split for the clickbait classification subtask.

and for the spoiler generation subtask, we trained our models for 5 epochs and with a batch size of 4. The maximum token length is 512 for all models except for the experiments with LongT5. We use a token length of 1024 with the LongT5 experiments.

After each epoch, we evaluated the output generated on the validation data. We used the `sklearn.metrics.balanced_accuracy_score` from the scikit-learn library (version 1.2.1) to evaluate the balanced accuracy score for the spoiler classification subtask and to choose the best-performing model. For the spoiler generation task, we evaluated the output by the BLEU-4 score using the `Evaluate` library (version 0.4.0) from Huggingface to choose the best-performing spoiler generation model for each spoiler category. We saved the model checkpoints after each epoch. We ran these experiments on LambdaLabs cloud instances, with an NVIDIA A100 GPU.

5 Results

5.1 Results on Validation

Table 2 shows our results on the validation split for the spoiler classification subtask. We only report the epoch that got the highest balanced accuracy score. The table also contains the baseline balanced accuracy score provided by the task organizers to show how our model compares. As can be seen, T5$_{LARGE}$ outperforms LongT5$_{LARGE}$ and gets a score that beats the baseline score of 0.734 (Fröbe et al., 2023a).

Table 3 shows our results on the validation split for the spoiler generation task. The models with the best BLEU-4 scores are highlighted for each spoiler category. We report the highest BLEU-4 score achieved in an epoch for each model. T5$_{LARGE}$ achieves the highest score for the phrase and multi spoiler types, while Flan-T5 achieves the best-performing results for the passage spoilers. Across all models, the BLEU-4 scores are the highest for the phrase type spoilers, followed by the passage type spoilers, and lastly the multi type spoilers.

Subtask	Model	BLEU-4
Phrase	T5$_{BASE}$	33.44
Phrase	T5$_{LARGE}$	**56.35**
Phrase	UNIFIEDQA$_{LARGE}$	50.74
Phrase	Flan-T5$_{LARGE}$	48.52
Phrase	LongT5$_{LARGE}$	39.17
Passage	T5$_{LARGE}$	20.74
Passage	UNIFIEDQA$_{LARGE}$	20.33
Passage	Flan-T5$_{LARGE}$	**21.36**
Passage	LongT5$_{LARGE}$	12.05
Multi	T5$_{LARGE}$	**8.94**
Multi	UNIFIEDQA$_{LARGE}$	5.93

Table 3: Results on the validation split for the spoiler generation subtask.

The results on the validation set guided our selection of models for our final system submission. We chose the fine-tuned T5$_{LARGE}$ as our model for the spoiler classification task since it performed the best, with the checkpoint that achieved the reported result. Similarly, we chose the T5$_{LARGE}$ for the phrase and multi type spoiler generation, and Flan-T5 for the passage spoiler generation.

5.2 Results on Test

The results on the test set come from our software submission on TIRA (Fröbe et al., 2023b). Table 4 shows the detailed results for the spoiler classification subtask. Our model achieves a balanced accuracy of 0.72 on the test set, which is lower than the score achieved on the validation set and than the baseline score provided for the evaluation set. However, our model gets the highest precision score for the passage type spoilers among all the submissions.

Detailed results for the spoiler generation task are provided in Table 5. Among all categories of spoilers, we achieve a BLEU-4 score of 0.41. The BLEU-4 score among all spoiler categories is

Accuracy	**Phrase**			**Passage**			**Multi**		
	Pr.	Rec.	F1	Pr.	Rec.	F1	Pr.	Rec.	F1
0.72	0.73	0.77	0.75	0.76	0.69	0.72	0.66	0.70	0.68

Table 4: Overview of the effectiveness in spoiler type prediction (Subtask 1 at SemEval 2023 Task 5) measured as balanced accuracy over all three spoiler types and precision (Pr.), recall (Rec.), and F1 score (F1) for phrase, passage, and multi spoilers on the test set.

All			Phrase			Passage			Multi		
BL4	BSc.	MET	BL4	BSc.	MET	BL4	BSc.	MET	BL4	BSc.	MET
0.41	0.92	0.44	0.69	0.96	0.71	0.24	0.90	0.42	0.12	0.88	0.35

Table 5: Overview of the effectiveness in spoiler generation (subtask 2 at SemEval 2023 Task 5) measured as BLEU-4 (BL4), BERTScore (BSc.) and METEOR (MET) overall clickbait posts respectively those requiring phrase, passage, or multi spoilers on the test set.

higher than the baseline score of 0.382 (Fröbe et al., 2023a). Our system achieves the highest scores for phrase spoilers among all other submissions. Similar to the results of the evaluation test, our system performs the best in phrase type spoilers, followed by the passage type spoilers, and finally the multi type spoilers.

5.3 Error Analysis

We investigated a sample of wrong predictions for Subtask 1 and wrongfully generated spoilers for Subtask 2 on the validation set.

For Subtask 1, the most common error is between classifying phrase and passage spoilers. The clickbait posts and articles for passage and phrase type spoilers can be very close in structure, especially since the data contains some short passage type spoilers and some longer phrase type spoilers that are similar to each other. Thus, our system is confused in those instances. One other common error is in the prediction of multi type spoilers. Our system reliably identifies the multi spoilers that have quantitative elements or indicators in the clickbait post or in the article but commonly misclassifies the multi posts and articles that don't have quantitative elements or indicators.

The most common error pattern for Subtask 2 is incomplete spoilers for phrase and multi spoilers. For passage spoilers, our system often identifies where the spoiler is in the article but struggles with extracting the whole passage as the spoiler. Similarly, our system struggles with the same issue for multi type spoiler, but additionally, it often has a difficulty in generating spoilers that are spread out across the text compared to the multi spoilers that are close together.

6 Conclusion

We present a question-answering framework to spoil clickbait posts. We show that the state-of-the-art Seq2Seq model T5 and its variants perform well in classifying the spoiler type and in generating the spoilers when fine-tuned on clickbait posts. We found that spoiling clickbait posts is easier for phrase type spoilers and gets progressively more difficult when the spoiler is longer, especially if it requires multi-span extraction like the multi type spoilers.

In future work, we plan to improve the performance of the spoiler generation for passage and multi type spoilers. We observed that a more complicated approach is needed for these longer spoiler types rather than a simple extractive question-answering method.

References

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

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.

Mandy Guo, Joshua Ainslie, David Uthus, Santiago Ontanon, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2021. Longt5: Efficient text-to-text transformer for long sequences.

Matthias Hagen, Maik Fröbe, Artur Jurk, and Martin Potthast. 2022. Clickbait Spoiling via Question Answering and Passage Retrieval. In *60th Annual Meet-*

ing of the Association for Computational Linguistics (ACL 2022), pages 7025–7036. Association for Computational Linguistics.

Markus Sverdvik Heiervang. 2022. Abstractive title answering for clickbait content. Master's thesis, University of Oslo, Department of Informatics.

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. Unifiedqa: Crossing format boundaries with a single qa system.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. *arXiv e-prints*, page arXiv:1606.05250.