Mitigating Clickbait: An Approach to Spoiler Generation Using Multitask Learning

Sayantan Pal , Souvik Das , Rohini K. Srihari
State University of New York at Buffalo
Department of Computer Science and Engineering
spal5, souvikda, rohini@buffalo.edu

Abstract

This study introduces 'clickbait spoiling', a novel technique designed to detect, categorize, and generate spoilers as succinct text responses, countering the curiosity induced by clickbait content. By leveraging a multi-task learning framework, our model's generalization capabilities are significantly enhanced, effectively addressing the pervasive issue of clickbait. The crux of our research lies in generating appropriate spoilers, be it a phrase, an extended passage, or multiple, depending on the spoiler type required. Our methodology integrates two crucial techniques: a refined spoiler categorization method and a modified version of the Question Answering (QA) mechanism, incorporated within a multi-task learning paradigm for optimized spoiler extraction from context. Notably, we have included fine-tuning methods for models capable of handling longer sequences to accommodate the generation of extended spoilers. This research highlights the potential of sophisticated text processing techniques in tackling the omnipresent issue of clickbait, promising an enhanced user experience in the digital realm.

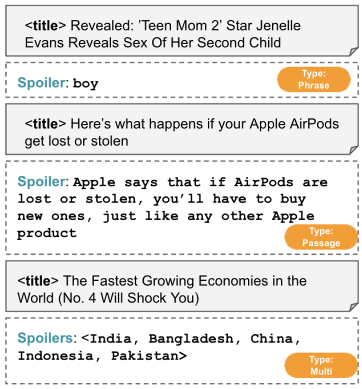

Figure 1: Examples of different categories of clickbait spoilers in Webis (Hagen et al., 2022) dataset.

1 Introduction

In today's fast-paced digital world, 'clickbait'—enticing but often misleading headlines designed to generate clicks have become a common trend.(Potthast et al., 2016a; Sterz et al., 2023). While these headlines draw users in, they often fail to deliver their promises, leading to user dissatisfaction (D. Molina et al., 2021). Consequently, 'clickbait spoiling'—providing succinct, truthful summaries (or 'spoilers') of clickbait content—has emerged as a powerful strategy to counter this trend (Hagen et al., 2022).

However, detecting, categorizing, and generating these spoilers involves sophisticated Natural Language Processing (NLP) techniques (Kurenkov et al., 2022). Furthermore, the type of required spoiler can vary from a short phrase to an extended passage or multiple snippets from the document, adding another layer of complexity to the problem (Roberts et al., 2019). Examples illustrating each of these types of spoilers are presented in Figure 1. Hagen et al. (2022) has proposed a two-step methodology to tackle clickbait. Although, Hagen et al. approach is a crucial contribution, it has enough room for improvement. Our research aims to extend their work by exploring ways to identify passage and multi-spoilers and enhance the user experience by providing accurate spoilers to clickbait content, mitigating the negative impacts of such misleading headlines (Chakraborty et al., 2016).

Contribution: In this study, we introduce a simple and novel methodology to solve the clickbait spoiling task using multi-task learning(MTL) framework,a method that enables the simultaneous learning of multiple related tasks to improve performance across them as outlined by Liu et al. (Liu et al., 2019). This allows a more comprehensive representation of the problem space, potentially enhancing its ability to generate accurate and diverse spoilers from different aspects of the same content. Specifically, we show that using **Spoiler generation in MTL setting can increase the overall generation quality**. Furthermore, we show **fine-**

tuning LongT5 model (Roberts et al., 2019) has displayed superior performance, achieving 60% higher BLEU-4 over our multi-task approach.

2 Background: Clickbait Spoiling

Clickbait spoiling has garnered significant attention in the research community, primarily driven by the increasing prevalence of clickbait headlines(Hurst, 2016) in the digital media landscape (Hovy et al., 2013). The misleading headlines (Wei and Wan, 2017) often lead to user dissatisfaction due to the discrepancy between the promised and delivered content (Potthast et al., 2016b). It has prompted the emergence of clickbait spoiling, a strategy aimed at providing users with honest, succinct summary generation techniques(Pal et al., 2021) to generate spoilers of the clickbait content, thus fostering a more satisfying user experience (Hagen et al., 2022).

2.1 Existing Approaches

Potthast et al. (2016c) and Agrawal (2016) have explored clickbait detection using Classical Machine Learning (ML) and Deep Learning (DL) approaches.Hagen et al. (2022) delves into more challenging clickbait spoiling in a unique two-step approach, which involves classifying spoiler types by finetuning BERT(Devlin et al., 2019) based models and treating the spoiler generation task as a question-answering or passage retrieval(Karpukhin et al., 2020) challenge. Lastly, it comprehensively evaluates leading techniques for spoiler-type classification and passage retrieval. The paper suggests that spoiler-type classification can be beneficial but not crucial and indicates an unaddressed area of research for generating multi-part spoilers.

2.2 Data

We used the Webis Clickbait(Hagen et al., 2022) Spoiling Corpus 2022[1] to conduct our experiments, it is a collection of 5,000 spoiled clickbait posts gathered from social media platforms like Facebook, Reddit, and Twitter. This corpus aims to assist in clickbait spoiling(Kurenkov et al., 2022), a process that involves creating a brief piece of text that satiates the curiosity generated by a clickbait post. The corpus is divided into 3,200 training samples, 800 validation samples, and 1,000 testing samples, which consist of phrases, passages, and multi-spoilers.

[1]Data: https://webis.de/data.html?q=clickbait

3 Our Approach

We address the challenge of mitigating clickbait by a two-step process. The first task is to classify the spoilers into phrases, passages, or multi, followed by the second task of generating contextually accurate and informative spoilers through a multi-task learning framework.

3.1 Clickbait Spoiler Classification

The primary goal of our spoiler classification approach is to analyze a document and accurately categorize spoilers based on their length and context, identifying them as short phrases, longer passages, or spanning multiple sections. In our spoiler classification approach, we leverage both the paragraphs (context) and the title text (question), which are concatenated and formatted as: `<[title_text] [SEP] [context]>` as the input, with the golden spoiler class labels (0, 1, 2) indicating phrase, passage, and multi, respectively. To process this input, we employ a 512-token sequence, applying necessary padding and truncation to maintain a consistent input size. A BERT-based encoder(Devlin et al., 2019) transforms this input into dense vector representations, mainly focusing on the hidden representation of the `[CLS]` token, encapsulating global information from the input. This representation is then passed through a linear layer followed by a softmax activation function to perform the classification. Cross-entropy loss given in equation 1 optimizes the model parameters, ensuring accurate and reliable spoiler detection.

$$L(y, \hat{y}) = -\sum_{i=1}^{C} y_i \log(\hat{y}_i) \tag{1}$$

3.2 Clickbait Spoiler Generation

Our objective is to perform Question Answering to generate the spoilers from the documents. In our multi-task learning framework for spoiler generation, we process paragraphs $\boldsymbol{P} \in \mathbb{R}^{d\times P}$, title text $\boldsymbol{Q} \in \mathbb{R}^{d\times Q}$, and golden spoilers $\boldsymbol{A} \in \mathbb{R}^{d\times A}$ as inputs, aiming to generate spoilers by Question Answering. Here, d represents the dimensionality of the embeddings, and P, Q, and A represent the lengths of these sequences. Our multi-task framework consists of the following tasks:

Task 1: Spoiler Generation with Auxiliary Question In this task, we guide the spoiler generation using auxiliary questions generated using a

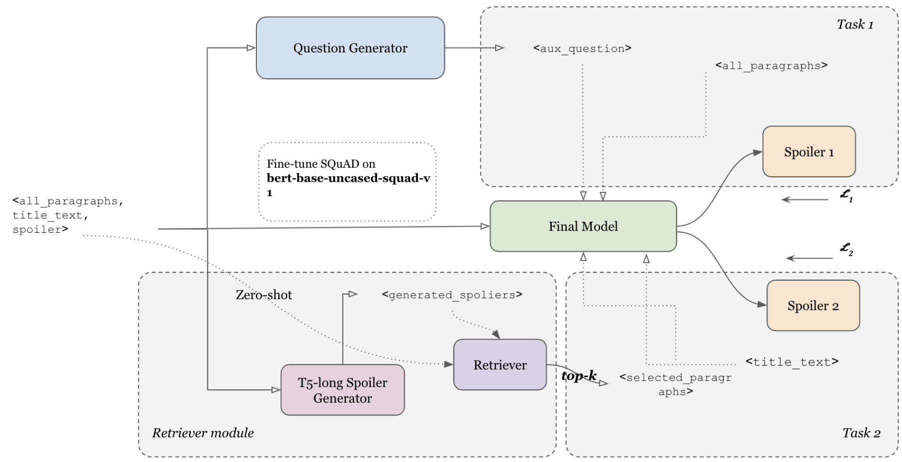

Figure 2: Overview of the Multitask Learning Learning Spoiler Generation Model

question generator[2]. The task can be formulated as:

$$\boldsymbol{S}_1 = f_{\text{QA}}(\boldsymbol{P}, \boldsymbol{Q}_{\text{aux}}, \boldsymbol{A}) \tag{2}$$

where $\boldsymbol{S}_1$ is the generated spoiler, $\boldsymbol{Q}_{\text{aux}}$ is the auxiliary question generated from the context $\boldsymbol{P}$, and f_{QA} is the final QA model[3].

Task 2: Spoiler Generation with Title Text In this task, the document's title text is treated as a question. To increase the precision of the process, we integrate a BM25[4] retriever module to extract the top k paragraphs from the original content, thereby narrowing the context for spoiler generation. Details in Appendix C

$$\boldsymbol{S}_2 = \begin{cases} f_{\text{QA}}(\boldsymbol{P}, \boldsymbol{Q}, \boldsymbol{A}), & \text{if } \boldsymbol{P}_{\text{top\_k}} = \boldsymbol{P} \\ f_{\text{QA}}(\boldsymbol{P}_{\text{top\_k}}, \boldsymbol{Q}, \boldsymbol{A}), & \text{if } \boldsymbol{P}_{\text{top\_k}} \subset \boldsymbol{P} \end{cases} \tag{3}$$

Equation 3 indicates an alternative method for calculating $\boldsymbol{S}_2$, where either the entire paragraph set $\boldsymbol{P}$ or the reduced paragraph set $\boldsymbol{P}_{\text{top\_k}}$ can be used for spoiler generation. The choice between these two depends on the specific use case and the desired balance between computational efficiency and accuracy.

We start by tokenizing both the original and generated questions ($\mathcal{Q}_{\text{orig}}$ and $\mathcal{Q}_{\text{gen}}$) alongside the contextual paragraphs ($\mathcal{P}$), utilizing a tokenizer τ that processes the text into token sequences while adhering to a maximum length of 384 tokens and ensuring proper padding and truncation. This results in sequences of token IDs and attention masks, as well as offset mappings ($\mathcal{T}_{\text{orig}}, \mathcal{O}_{\text{orig}}, \mathcal{T}_{\text{gen}}, \mathcal{O}_{\text{gen}}$). Concurrently, the answers ($\mathcal{A}$) are processed to ascertain their start and end positions within the tokenized context, translating character-level positions into token-level indices ($\text{start}_{\text{token}}, \text{end}_{\text{token}}$). Details in Appendix A

The final spoiler $\boldsymbol{S}$ is selected based on the combination of results from the main and auxiliary tasks, where the output spans with the lowest combined validation loss across both tasks are chosen as the final spoiler prediction

To optimize the parameters of our model, we use a loss function that combines the losses from both tasks. The total loss $\mathcal{L}$ is calculated as:

$$\mathcal{L} = \mathcal{L}_2 + \alpha \cdot \mathcal{L}_1 \tag{4}$$

where $\mathcal{L}_1$ and $\mathcal{L}_2$ are the cross-entropy (same as Eq. 1) losses for Task 1 and Task 2, respectively, and α is a hyperparameter. It is experimentally determined from the validation set. In our case, α value of 0.5 yielded optimal results. Implementation details in Appendix B

4 Results and Analysis

4.1 Spoiler Classification Task

We experimented with spoiler classification within a multi-class setting and rigorously tested many neural models, including DistillBERT(Sanh et al., 2019), RoBERTa(Zhuang et al., 2021), and Longformer(Beltagy et al., 2020) reported in Table 1. We found our RoBERTa-Large model surpassed the state-of-the-art result of accuracy 73.63% (Hagen et al., 2022), albeit by a small margin and it has been statistically analyzed (Appendix E).

[2] valhalla/t5-base-qg-hl

[3] we initialize our final QA model using the checkpoints of: https://huggingface.co/csarron/bert-base-uncased-squad-v1, subsequently fine-tuned on Webis data.

[4] https://pypi.org/project/rank-bm25/

Model Name	Eval Acc	Test Acc	Eval F1	Test F1
DistilBERT	67.8	67.7	67.7	66.2
Longformer	68.75	68.43	67.56	66.46
RoBERTa	71.8	71.46	70.3	70.26
RoBERTa-Large	**73.56**	**75**	**72.59**	**73.74**

Table 1: Comparison of model performances for Spoiler Classification Task

Model	BLEU-4	METEOR	BERT Sc.
BERT-b-u (v)	56.96	47.44	76
BERT-b-u (t)	52.79	59.16	84
BERT-b-u (MTL) (v)	47.69	40.12	67
RoBERTa-b (v)	64.11	53.92	79
RoBERTa-b (t)	61.26	53.82	78
Our RoBERTa-l (v)	73.36	61.38	84
Hagen et al. RoBERTa-l (v)(n=97)*	79.47	78.61	84.04
Our RoBERTa-l (t)	**68.35**	60.79	**83**
Hagen et al. RoBERTa-l (t)	65.70	66.15	74.81

Table 2: Phrase Spoiler Generation Results——Here the letter b stands for the base, l stands for large, and c/u denotes cased or uncased, (v) means validation samples, (t) means test samples, Total phrase spoiler in validation samples were 335, test samples were 423. * represents Hagen et al. used a subset of 97 Phrase spoilers out of 335 to report the scores.

4.2 Spoiler Generation Task

4.2.1 MTL vs. Non-MTL

In this section, we dive into the intricacies of spoiler generation and the implications of the methodologies employed in the study. The quality of generated spoilers is evaluated based on several metrics(Appendix F).

From table 2,3 we observe Roberta-Large achieved the highest BLEU-4 score for phrase spoilers among BERT-based models, while LongT5 significantly outperformed others in passage spoiler generation despite a higher computation cost (Details in Appendix D).

4.2.2 Case Study: Does context reduction help in MTL?

Our examination of 335 passage spoilers from a test dataset showed no notable improvements initially, prompting an investigation into how context size affects Multitask Learning (MTL) and non-MTL setups, as detailed in Table 3. Utilizing the retriever component revealed a distinct advantage for MTL setups, especially with the RoBERTa base model, pointing towards intriguing future research possibilities on context reduction and retriever models.

Model	BLEU-4	METEOR	BERT Score
BERT-b-u (v)	21.11	22.72	52.68
BERT-b-u (t)	17.09*	23.65*(335)	53.62(335)
BERT-b-u (MTL) (v)	14.28	13.21	44
BERT-b-u (MTL) (t)	**17.78***	22.56*(335)	53*(335)
RoBERTa-b (v)	26.73	27.97	58
RoBERTa-b (t)	21.81*	29.72*(335)	58*(335)
RoBERTa-b (MTL) (v)	**26.86**	**28.14**	54
RoBERTa-b (MTL) (t)	**22.59***	**32.67*(335)**	**60(335)**
RoBERTa-l (v)	30.67	32.78	58
RoBERTa-l (t)	26.52	36.66(335)	63(335)
RoBERTa-l (MTL) (v)	29.71	30.99	56
RoBERTa-l (MTL) (t)	26.65	36.42(335)	63(335)
Hagen et al. (Best)	31.44	46.06	51.06
LongT5 (v) (our best)	**88.72**	**90.29**	**97.98**
LongT5 (t) (our best)	**90.10**	**90.81**	**98.17**

Table 3: Passage Spoiler Generation Results——Here the letter b stands for the base, l stands for large, and c/u denotes cased or uncased, (v) means validation samples, (t) means test samples, * means the setting includes reduction of context. Total passage spoilers in validation samples were 322, and test samples were 403, and Numbers in the brackets indicate a subset of size n (random sample)

Model	BLEU-4	METEOR	BERT Score
Hagen et al.	-	-	-
LongT5	81.55	85.39	96.67

Table 4: Multi Spoiler Generation Results

4.2.3 Multi Spoilers

We used fine-tuned LongT5 to generate multi-spoilers and found exceptionally good results in spoiler generation quality, reported in Table 4.

5 Conclusion and Future Work

In this work, we have notably advanced the field of clickbait spoiling, particularly excelling in scenarios involving reduced context. Our research lays a solid groundwork for finding the optimal parameters for context reduction and innovative context generation techniques, ensuring better performance in spoiler generation tasks. Furthermore, the study underscores the superiority of LongT5-based models over their BERT-based counterparts in generating extended spoilers. The sensitivity of our MTL model's performance to the alpha hyperparameter, initially a limitation, opens a direction for developing adaptive strategies to fine-tune the balance between primary and auxiliary tasks, optimizing the model's efficiency. The future also holds the potential for rigorous statistical analysis to strengthen the validity of the performance gains attributed to Multitask Learning.

References

Amol Agrawal. 2016. Clickbait detection using deep learning. In *2016 2nd International Conference on Next Generation Computing Technologies (NGCT)*, pages 268–272.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *ArXiv*, abs/2004.05150.

Abhijnan Chakraborty, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly. 2016. Stop clickbait: Detecting and preventing clickbaits in online news media. In *Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '16, page 9–16. IEEE Press.

Maria D. Molina, S. Shyam Sundar, Md Main Uddin Rony, Naeemul Hassan, Thai Le, and Dongwon Lee. 2021. Does clickbait actually attract more clicks? three clickbait studies you must read. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA. Association for Computing Machinery.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Matthias Hagen, Maik Fröbe, Artur Jurk, and Martin Potthast. 2022. Clickbait spoiling via question answering and passage retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7025–7036, Dublin, Ireland. Association for Computational Linguistics.

Dirk Hovy, Taylor Berg-Kirkpatrick, Ashish Vaswani, and Eduard Hovy. 2013. Learning whom to trust with MACE. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1120–1130, Atlanta, Georgia. Association for Computational Linguistics.

Nathan Hurst. 2016. *To clickbait or not to clickbait?: an examination of clickbait headline effects on source credibility*. Ph.D. thesis, University of Missouri–Columbia.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Andrey Kurenkov, TA Mentor, Yian Zhang, and Oliver C. Johnson. 2022. Saved you a click: Automatically answering clickbait titles. *ArXiv*, abs/2212.08196.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496, Florence, Italy. Association for Computational Linguistics.

Sayantan Pal, Maiga Chang, and Maria Fernandez Iriarte. 2021. Summary generation using natural language processing techniques and cosine similarity. In *International Conference on Intelligent Systems Design and Applications*.

Martin Potthast, Sebastian Köpsel, Benno Stein, and Matthias Hagen. 2016a. Clickbait detection. In *European Conference on Information Retrieval*.

Martin Potthast, Sebastian Köpsel, Benno Stein, and Matthias Hagen. 2016b. Clickbait detection. In *Advances in Information Retrieval: 38th European Conference on IR Research, ECIR 2016, Padua, Italy, March 20–23, 2016. Proceedings 38*, pages 810–817. Springer.

Martin Potthast, Sebastian Köpsel, Benno Stein, and Matthias Hagen. 2016c. Clickbait detection. In *Advances in Information Retrieval*, pages 810–817, Cham. Springer International Publishing.

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

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

Hannah Sterz, Leonard Bongard, Tobias Werner, Clifton Poth, and Martin Hentschel. 2023. ML mob at SemEval-2023 task 5: "breaking news: Our semi-supervised and multi-task learning approach spoils clickbait". In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 1818–1823, Toronto, Canada. Association for Computational Linguistics.

Wei Wei and Xiaojun Wan. 2017. Learning to identify ambiguous and misleading news headlines. In *International Joint Conference on Artificial Intelligence*.

Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. 2021. A robustly optimized BERT pre-training approach with post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1218–1227, Huhhot, China. Chinese Information Processing Society of China.