Clickbait Detection and Spoiler Generation

Ponnampalam Pirapuraj
dept. Computer Science and Enginnering
Indian Institute of Technology Hyderabad
cs23resch16001@iith.ac.in

Souvik Sarkar
dept. Computer Science and Enginnering
Indian Institute of Technology Hyderabad
cs23mtech02001@iith.ac.in

***Abstract*—Clickbait articles are a widespread problem on the internet. One of the solutions put forward to solve the issue is the generation of spoilers, which are brief texts that negate clickbait by offering information that satisfies the curiosity causing it.In our approach we not only generating spoiler to kill the curiosity of the read but also detecting whether a post is clickbait or a non clickbait.In this approach we it using question answering strategy, where we used T5 and BeRT models.**

I. Introduction

Clickbait is a type of text on social media that is specifically designed to exploit reader's curiosity and force them into clicking on a linked website.This clickbaits are often very misleading in nature, that's why it is often considered harmful .So it is crucial to address this issue.

Generating Soplier is not a straight forward thing because there can three types of spoiler.Three categories exist for spoilers: passage spoilers (a few sentences at most), phrase spoilers (a single word or short phrase), and multi-part spoilers (spoiler scattered throughout the article) [1]. In order to solve this problem we can do clickbait detection where we try to find out whether an article or post is clickbait or not. After detection if we find that the post is a clickbait we can generate a spoiler to that post. In our approach we used the question answering methodology [1] which is a very well known and effective strategy to determine whether a post is clickbait or not.we use various T5 models to convert the title into a question , BERT[1] model to answer the question. At the end we used S-BERT[2] to check the similarity between the given title and the generated title.

type	clickbait	spoiler
phrase	You're missing this major way to save money	promotional code
passage	Scientists unearth big surprise near celebrated pyramids	remains of a bustling port and barracks for sailors or troops
multi	This is what REALLY happens when you don't brush your teeth	Bad breath, Coronary heart disease, Bleeding gums, (...)

Fig. 1. Types of spoilers [1]

[1]https://huggingface.co/docs/transformers/en/model_doc/bert
[2]https://huggingface.co/sentence-transformers

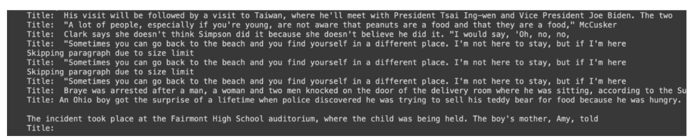
II. Methodology

As there are two tasks assigned in the project, which are Clickbaits detection and Spoiler generation, the methodology section discusses the two tasks separately. The section A discusses the Clickbaits detection and dataset generation for detection task and the B discusses the Spoiler generation.

A. Clickbaits detection

Clickbaits detection is an initial and crucial task in any clickbait related task. We introduce a novel approach for detecting whether the given postText is the clickbaits or not. The figure 2 explains the approach, and the figure 3 explains the modified version of the approach because of some imprecise results. For distinguishing purposes, we call the approach and modified approach as approach 1 and approach 2 respectively. First, we tried approach1 with the GPT-2 model, and we received a big paragraph as title, then we changed the model into T5. The result of GPT-2 is mentioned in figure 2

As illustrated in the figure 3, a title is generated from



```
Title:  His visit will be followed by a visit to Taiwan, where he'll meet with President Tsai Ing-wen and Vice President Joe Biden. The two
Title:  "A lot of people, especially if you're young, are not aware that peanuts are a food and that they are a food," McCusker
Title:  Clark says she doesn't think Simpson did it because she doesn't believe he did it. "I would say, 'Oh, no, no,
Title:  "Sometimes you can go back to the beach and you find yourself in a different place. I'm not here to stay, but if I'm here
Skipping paragraph due to size limit
Title:  "Sometimes you can go back to the beach and you find yourself in a different place. I'm not here to stay, but if I'm here
Skipping paragraph due to size limit
Title:  "Sometimes you can go back to the beach and you find yourself in a different place. I'm not here to stay, but if I'm here
Title:  Braye was arrested after a man, a woman and two men knocked on the door of the delivery room where he was sitting, according to the Su
Title: An Ohio boy got the surprise of a lifetime when police discovered he was trying to sell his teddy bear for food because he was hungry.

The incident took place at the Fairmont High School auditorium, where the child was being held. The boy's mother, Amy, told
Title:
```

Fig. 2. The output of GPT-2 model in title generation

the article of the postText using the JulesBelveze/t5-small-headline-generator model. Now, we have postText on one hand and generated title on the other hand. Finally, we take the embedding of the PostText and generated title, and find the cosine similarity between them using the S-BERT model. Our hypothesis is that, PostText is the headline of the article which stimulates the visitor to click the article, and the generated title from the article which conveys the exactly available information inside the article. If the cosine similarity is high for both 'T' and 'T'', then the postText is not a clickbait, if the cosine similarity is very low for both, then the posttext is clickbait. After implementing the approach1, we evaluated the developed system and received some imprecise results. We used the Webis-Clickbait-22 [1] dataset for initial testing, which includes full of clickbaits. But, the approach resulted in so many postTexts are not clickbaits as wrong. The figure 5 includes some results of this approach, in which the score is around 0.9 which means the approach detected as some

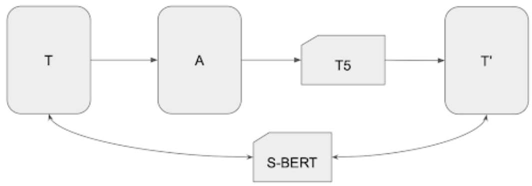


Fig. 3. Overview of proposed approach for detecting clickbaits, where 'T' is postText, 'A' is article and 'T''is generated title

postText are not clickbait while they are clickbait. The reason for getting the highest score is that the both generated title and the postText are semantically similar. The two phrases

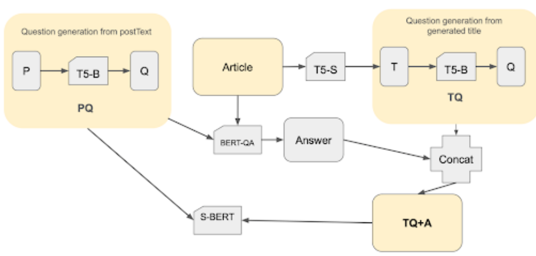


Fig. 4. The overview of approach 2, where, PQ is question format of postText

Generated Title: How much you should tip your hairdresser

The postText: Here's how much you should be tipping your hairdresser

Score: 0.90563005

Generated Title: Do bird fart?

The postText: Bird fart that can pass gas

Score: 0.86571044

Fig. 5. Sample output of the proposed approach

included in the figure 5 are clickbaits from Webis-Clickbait-22, the generated text and the postText should be different, but both are semantically similar here. We need to make them different, therefore, we decided to add some string to the generated text for making them different. Finally, we have decided to modify the approach1 as illustrated in figure 4 to approach2.

First, we convert both postText and generated title into question form (PQ and TQ) using the valhalla/t5-base-qg-hl model then, we derive the answer for the question of postText using the bert-large-uncased-hole-word-masking-fine tuned-squad model[3], and then, we add the derived answer with the question of generated title (TQ+A). Finally, we find the similarity score using the S-BERT model.

B. The dataset creation for the clickbaits detection

The Webis-Clickbait-22 includes full of clickbaits data, we need a data set which should include both clickbaits and no-clickbaits data in it for detection task. Therefore, we generated our own dataset using a research article published by A Chakraborty, et al. [4], the sample dataset of them is included in the figure (a) in figure 6. We have utilized the headlines from their dataset for downloading the corresponding articles using the web crawling process. The sample of our generated data set is included in figure (b) in figure 6. Our dataset includes 100 records, in which 50 records are clickbaits and 50 records are non-clickbaits.

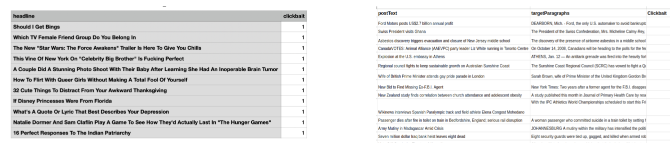


Fig. 6. The dataset samples used for clickbaits detection. (a). includes a dataset sample from [], (b). includes our own generated dataset sample [1], [2]

C. Spoiler generation

The second task of the project is spoiler generation. For generating multi-part spoilers, we apply the prompting engineering technique. We tried with open AI GPT-3 and Anthropic Claude[4] models by using API requests. We faced RAteLimitErro in both models, because of their request limit per day. And then, we tried with GPT-2 and T5-base[5] models, we received imprecise results. Finally, we tried with the Flan-t5-xl model, and achieved better results.

As we discussed earlier we tried with simple prompt and chain of thought prompt, but simple prompt worked fine. The figure 7 showing the prompts tried in this task, uncommented one gave better results.

```
#base_prompt = "Extract very important point only from the below paragraph:\n"
base_prompt = "Extract spoiler points from the below paragraph:\n "
#base_prompt = "Extract spoilers from the below paragraph:\n"

#base_prompt = "Extract very important point only from the below paragraph:\n"
base_prompt = """
Extract spoiler points from the below paragraph:\n
Chain of Thought:
    1. Read through the article text carefully to understand the overall context and story.
    2. Identify sentences or phrases that reveal important plot points, twists, or critical information about the story.
    3. Extract those sentences or phrases and store them as potential spoilers.
    4. Review the potential spoilers to ensure they are indeed spoilers and not just general information.
    5. Organize and present the extracted spoilers in a clear and concise manner.
"""
#base_prompt = "Extract spoilers from the below paragraph:\n"
```

Fig. 7. The prompt used to generate multi-part spoiler

III. Result

A. Result of clickbait detection

Table I shows the result of the clickbait detection task. We executed our setup on our own generated dataset and received better performance as 65.09% of accuracy, 65.07% of precision and 73.01% recall.

[3]https://huggingface.co/google-bert/bert-large-uncased-whole-word-masking-finetuned-squad

[4]https://www.anthropic.com/api

[5]https://huggingface.co/docs/transformers/en/model_doc/t5

TABLE I
THE RESULT OF CLICKBAIT DETECTION TASK

Model	Accuracy	Precision	Recall
Multiple-Models	65.09%	65.07%	73.0%

B. Result of spoiler generation

As we discussed in section 2, we tried with the finally selected prompt with the Flan-t5-xl model, we received a 17.9% BLUE score. The table II shows the comparison of the SOTA model's result with our result on multi-part spoiler generation.

TABLE II
ACCURACY COMPARISON WITH SOTA RESULTS

Methods in mult-part spoiler	Accuracy
Mateusz Wozny, et al -2023	31.71%
TohokuNLP at SemEval-2023 Task 5	43.96%
Our approach	17.9%

IV. CONCLUSION

Our approach is right now not giving very good results. For improving our performance we have to fine tune the T5 models hyperparameter's. In our approach we are using a threshold value for determining whether a post is clickbait or not a clickbait, we can try to set this threshold value more precisely so that we can detect clickbait with more precision and accuracy.

REFERENCES

[1] Mateusz Woźny and Mateusz Lango. 2023. Generating clickbait spoilers with an ensemble of large language models. In Proceedings of the 16th International Natural Language Generation Conference, pages 431–436, Prague, Czechia. Association for Computational Linguistics.

[2] Hiroto Kurita, Ikumi Ito, Hiroaki Funayama, Shota Sasaki, Shoji Moriya, Ye Mengyu, Kazuma Kokuta, Ryujin Hatakeyama, Shusaku Sone, and Kentaro Inui. 2023. TohokuNLP at SemEval-2023 Task 5: Clickbait Spoiling via Simple Seq2Seq Generation and Ensembling. In Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023), pages 1756–1762, Toronto, Canada. Association for Computational Linguistics.

[3] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., & Zhou, D. (2022). Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. ArXiv. /abs/2201.11903

[4] Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., & Iwasawa, Y. (2022). Large Language Models are Zero-Shot Reasoners. ArXiv. /abs/2205.11916

[5] Abhijnan Chakraborty, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly. "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), San Fransisco, US, August 2016.