Matt Bai at SemEval-2023 Task 5: Clickbait spoiler classification via BERT

Nukit Tailor
International Institute of
Information Technology,
Hyderabad
nukit.t@research.iiit.ac.in

Radhika Mamidi
International Institute of
Information Technology,
Hyderabad
radhika.mamidi@iiit.ac.in

Abstract

The Clickbait Spoiling shared task aims at tackling two aspects of spoiling: classifying the type of spoiler based on its length and generating the spoiler. This paper focuses on the task of classifying the spoiler type. A better classification of the spoiler type would eventually help to generate a better spoiler for the post. We use BERT-base (cased) to classify the clickbait posts. The model achieves a balanced accuracy of 0.63 as we give only the post content as the input to our model instead of the concatenation of the post title and post content to find out the differences that the post title might be bringing in.

1 Introduction

Clickbaits are becoming extremely popular nowadays, and people are falling for them way too easily. Since it plays on people's innate curiosity and need for rapid satisfaction, clickbait may be an effective tactic for driving traffic to a website or increasing interaction on social media, hence resulting in revenue. The general conception about clickbaits is that it is often seen as a bad strategy, since it frequently misleads users and gives little value after they click through to the content. This can lead to dissatisfaction, disillusionment, and a lack of faith in the clickbait's source. However, not all clickbait posts are bad and misleading, some people genuinely use clickbaits to gain more engagement, they also have good content.

The Clickbait spoiling shared task (Fröbe et al., 2023a) aims at generating short texts that satisfy the curiosity induced by a clickbait post. Spoilers are needed to briefly let the reader know what to expect when the article is clicked on or whether to click on it or not. The task has two subtasks, one in which we have to classify the clickbait spoiler in 3 classes namely "phrase", "passage", and "multi" which are short phrase spoilers, longer passage spoilers, and multiple non-consecutive pieces of

text, respectively. The second task is to generate the spoiler itself, which is not discussed here.

The main reference paper (Hagen et al., 2022) concluded that improving the spoiler type classification would also improve the generation of spoilers. The dataset contains 5000 English posts which have a manually extracted spoiler for each. In our approach, we have taken the BERT-base model and given it the content of the clickbait posts as the input as opposed to the the concatenation of the title and content of the post as done in the original approach. We find that removing the post title has a negative overall impact on the scores.

2 Background

Even though clickbaits have been around here for a long time now and topics like clickbait identification are being popular every passing day, the task of spoiling them is fairly new. Notable work in this field has been done by the organizers of shared task themselves in which they use multiple approaches to generate spoilers including Question Answering and Passage Retrieval. (Hagen et al., 2022) They also found that non-transformer-based methods like Naive Bayes, SVM, or logistic regression perform worse than transformer-based methods like DeBERTa and RoBERTa, which leads us to the conclusion that the task of classification requires much more knowledge and understanding of the language than what can be encoded in these systems.

The paper releases its own dataset, a total of 5000 clickbait posts with spoilers. These posts were first collected from social networks where accounts like r/savedyouaclick and @HuffPoSpoilers have already spoiled thousands of clickbait links and articles. Each of the 5,000 posts in this dataset consisted of a unique ID, the platform from which they were taken, the post's text, the manually extracted title and the main content divided by paragraphs of the linked document, the manually opti-

mized spoiler, the character position of the spoiler in the main content, and the type of spoiler.

3 System Overview

We approach the problem of classification of the clickbait posts by using the post's content as the only input. We use BERT-base model which is a pretrained model on English language using a masked language modeling (MLM) objective. (Devlin et al., 2018) We fine-tune this model for our task of classification of the spoiler type. We have used the case-sensitive version of this model, which differentiates between lowercase and uppercase characters.

4 Experimental Setup

The data of 5000 posts was split into 3200 for training, 800 for validation and the remaining 1000 for testing. The code to replicate the setup can be found here[1].The post text was then extracted and was given as input to the bert layer, with a dropout of 0.1 which was then passed through a linear layer and then a ReLU to get final classes prediction. The model is trained using Adam optimizer with a learning rate of 1e-6 for 10 epochs.

For evaluating the systems, the organizers use balanced accuracy and F1-scores to obtain overall quality for the classification.

5 Results

Our system managed to get a balanced accuracy of 0.63 with this approach and F1-score of 0.69, 0.65 and 0.61 for the classification into phrases, passages and multipart, respectively.

Although we couldn't manage to beat the original papers' scores which they achieved with RoBERTa, compared to their scores for the BERT in which they got a balanced accuracy of 0.67.

The main reason for this difference apart from the minor experimental setup, is the fact that the post title in the dataset was not used by us. The post content, although has more details than the title, is not sufficient in classification of the type of spoiler to be generated which means that from the title of the clickbait link or post the type of spoiler can be inferred. This can also be manually verified by taking some example annotated spoilers. For example, if the title of a clickbait post has lines like "5 ways to become healthy overnight" or "List of

Table 1: Overview of the effectiveness in spoiler-type prediction.

Metric	Phrase	Passage	Multi
Precision	0.63	0.67	0.85
Recall	0.77	0.64	0.47
F1	0.69	0.65	0.61
Balanced Accuracy	0.63		

things needed for office work to be done efficiently" it can be inferred that a multi part answer would be the perfect fit for it.

6 Conclusion

In this paper, we compare the performance of the BERT-base model on the task of clickbait spoiler classification with two different approaches that differ in the data presented to them. Our system uses only the content of the clickbait post and gave a balanced accuracy of 0.63 and other system which concatenated the post title and the post content achieved a balanced accuracy of 0.67. We conclude that the type of spoiler to be generated can be inferred more effectively with the help of the title and it cannot be ignored.

7 Acknowledgments

We would like to thank the organizers of the Clickbait Challenge for this opportunity.

References

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

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.

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.

---

[1]https://github.com/Nukitt/Clickbait-spoiler-classification