Spoiler detection and extraction
Project Proposal for NLP Course, Winter 2022

M. Kierznowski, Ł. Pancer, P. Wesołowski
Warsaw University of Technology

supervisor: Anna Wróblewska
Warsaw University of Technology
anna.wroblewska1@pw.edu.pl

Abstract

In this document, we present a literature review, describe project ideas and provide an initial outline of the solution for the first NLP course project. The project will address the spoiler detection task. Current research in this field mainly relates to classification without touching the model interpretability. The main goal is to create a novel benchmark for evaluating the performance of the spoiler classification. The benchmark would analyze the model's rationale and compare it with precise human knowledge. We are going to evaluate the performance of the state-of-the-art models with our tool. Besides, we are going to extend the previous spoiler detection works by utilizing a new dataset, hoping for an improvement in performance.

1 Introduction

In everyday life, information that reveals the important aspects, events, and twists of a plot of, e.g., a book, a movie, or a videogame is referred to as a spoiler. Spoilers are usually undesired, and getting to know one of them may contribute to a decrease in enjoyment (Abbott, 2020) or a lack of further interest in reading or seeing the particular position (Li et al., 2022). However, they can often become known by chance, for example, due to being a part of some review. Therefore, automatic spoiler detection has become one of the crucial tasks in the NLP field (Guo and Ramakrishnan, 2010).

Several methods on this issue have been proposed. While the authors often focus on their classification performance, we believe that the interpretability aspects are also important and may be an adequate measure of the actual performance. Therefore we are going to develop a tool for the assessment of the spoiler classification performance on the basis of the model rationale.

The use of transfer learning has already become pretty standard due to better performance, reduction of training time, electricity usage, etc. However, proper fine-tuning on the task-related dataset may be important for further improvement in the model's robustness and accuracy. Similarly to our predecessors, we are going to utilize powerful pre-trained models. However, we would like additionally to make use of the IMDB reviews dataset (Biswas, 2021), which does not appear to be frequently used. We believe that due to its size (and, therefore, probably its diversity), it may be a good source of additional knowledge for our models.

Our project is driven mainly by two questions:

1. How do the numeric measurements of the model performance (accuracy, recall, etc.) relate to its performance evaluated in a more strict but interpretability-based way?
2. Does providing a larger task-related dataset for fine-tuning result in the improvement of the spoiler detection model?

2 Literature Review

Spoiler detection tasks had been previously neglected, and only in recent years it saw more rapid development. Early studies related to spoiler detection treated it as a traditional classification task, most commonly utilizing methods such as Support Vector Machines. The progress revolved around the process of building a larger dataset with more linguistic features available. Publications explored basic Bag-of-Words approaches: creating a blacklist of words for a given topic (Golbeck, 2012), introducing temporal filtering system (Nakamura and Tanaka, 2007), using topic models based on Latent Dirichlet Allocation (Guo and Ramakrishnan, 2010). Significant contributions to publicly available datasets include TV Tropes Movies (Boyd-Graber et al., 2013) and GoodReads dataset (Wan et al., 2019). Chang

(2018) proposed an attention-based solution, and today similar approaches still remain state-of-the-art. More recently, a transfer learning approach was suggested, laying the stress on interpretability (Wróblewska et al., 2021). On the topic of interpretability, publications of LIME (Ribeiro et al., 2016b), and SHAP (Lundberg and Lee, 2017) are particularly interesting. LIME is an explanation technique designed to explain black-box models, which are common in the NLP field due to the high-dimensional nature of features. It uses a local linear approximation to explain the model's predictions. SHAP is a superset of LIME, basing the explanations on a game theory concept of Shapley values. LIME and SHAP in NLP models allow extract words that are responsible for making a certain prediction and provide metrics on the strength of their contribution. DeYoung (2019) introduced a benchmark based on datasets with rationales annotated by humans to assess the interpretability of NLP models by comparing their rationales with humans'.

3 State of the Art

The provided literature, as well as other state-of-the-art solutions, are based upon a Bidirectional Encoder Representations from Transformers (BERT) architecture (Devlin et al., 2018). BERT is an eminently versatile solution, finding use in many corners of the Natural Language Processing field. It was originally developed at Google AI Language and includes the encoder part of the transformer architecture to create word embeddings. Figure 1 shows the bidirectional nature of the encoder. A sequence of words is processed at once, making it possible to extract the context of a word from both the preceding as well as following words.

GloVe (Global Vectors) (Pennington et al., 2014) is another popular algorithm for creating word embeddings. It makes use of a word co-occurrence counts matrix to calculate a conditional probability of words being present in a given context, which is used as a measure of semantic similarity. The words are arranged in the vector space to connect the distance between words to similarity. The adoption of word co-occurrence allows extracting global as opposed to only local relationships between words. However, an inevitable disadvantage of GloVe is the inability to differentiate homographs, i.e., words that are spelled the same but have a different meaning.

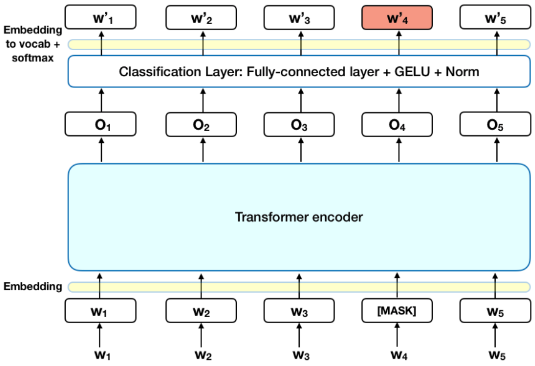
w'1
w'2
w'3
w'4
w'5
Embedding to vocab + softmax
Classification Layer: Fully-connected layer + GELU + Norm
O1
O2
O3
O4
O5
Transformer encoder
Embedding
w1
w2
w3
[MASK]
w5
w1
w2
w3
w4
w5

Figure 1: BERT Architecture Diagram `https://miro.medium.com/max/1400/0*ViwaI3Vvbnd-CJSQ.webp`

4 General outline

To our knowledge, the current research on spoiler detection task mainly concerns classification, and the performance is assessed using widely known quantitative metrics. Still, there exist datasets, mentioned in section 2, with annotated spoiler phrases (with a sentence- or word-level annotations). We would like to investigate whether the machine learning models developed for the classification of the whole documents, viz. containing a spoiler or not, actually correctly identify these phrases. More strictly, we would like to know if the annotated document phrases are effectively responsible for classifying a review as a spoiler.

We would like to focus on the assessment of the document classification in the following way. For every document classified as containing a spoiler, we want to use XAI tools to get the phrases that determined the model decision. Then, we can compare the crucial set of phrases with the ones selected by annotators (either word- or sentence-level). The comparison can be made using different metrics. As for now, we consider Jaccard's similarity.

Going into the details, in our work, we are going to use the following datasets:

- Goodreads - 1.3M documents, 17M sentences, 570k spoiler sentences,
- TV Tropes Books - 340k documents, 670k sentences, 110k spoiler sentences,
- IMDB reviews - 5.5M documents, 1.1M spoiler documents.

The first dataset consists of reviews with spoiler fragments annotated by the community. Goodreads members can mark text using introduced spoiler tags in order to hide specific phrases. We are going to create a balanced version of this dataset, as it was done by Wróblewska (2021). The TV Tropes Books dataset provides word-level annotations hence it's the most valuable for us. The IMDB reviews dataset offers only document-level annotations.

Regarding the models used, we would probably utilize the ones used in similar works. We are going to use deep learning models built on top of pre-trained state-of-the-art solutions. However, we would like to fine-tune the models first using IMDB dataset. We would like to know if such a procedure is helpful for the task considered.

5 Concept and Work Plan

Regarding topic our plan is to conduct a research on detecting spoilers using state-of-the-art architectures like BERT or RoBERTa. Subsequently we want to use various XAI techniques in order to understand with words model consist to be spoilers. Going into the details we want to test Local Interpretable Model-agnostic Explanations (LIME) which perturbs instance we want to explain, learns a sparse linear model around it, as an explanation (Ribeiro et al., 2016a). For the next XAI method Evaluating Rationales And Simple English Reasoning (ERASER) which is benchmark to advance research on interpretable models in NLP. Furthermore attention layers is another candidate for incorporating XAI method in work. However we need to remember that not all languages models has it's own attention layers to work with. The goal of final analysis is to develop end-to-end architecture that properly distinguish spoilers reviews and provide information why specific words triggers model (and with what strength). However introduced approaches may be connected with some risk factors. Firstly not all language models will equally response for XAI techniques e.g previously mentioned attention layer. Subsequent issue may concern data sets which are build differently concerning annotation level. TV Tropes Books has every word-level annotation whereas Goodreads only sentence-level and IMDB only document annotations. It may lead to abandonment of some source of data e.g ERASER needs to have word-level annotations.

6 Summary

In this document, we described our preliminary findings pertaining to the first NLP course project. We believe that any interesting conclusions and observations may lead to more interpretability-oriented research in the future. We expect that analyzing the quality of a model not only based on hard measures but also on consistency with human annotators can lead to better solutions. In particular, we hope that it has the chance to help in the further advances of automatic spoiler detection, which from our perspective, is an important subject.

References

Marshall Abbott. 2020. Can spoilers in online reviews impact viewer enjoyment?

Enam Biswas. 2021. Imdb review dataset - ebd. https://www.kaggle.com/dsv/1836923.

Jordan Boyd-Graber, Kimberly Glasgow, and Jackie Sauter Zajac. 2013. Spoiler alert: Machine learning approaches to detect social media posts with revelatory information. *Proceedings of the American Society for Information Science and Technology*, 50(1):1–9.

Buru Chang, Hyunjae Kim, Raehyun Kim, Deahan Kim, and Jaewoo Kang. 2018. A deep neural spoiler detection model using a genre-aware attention mechanism. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 183–195. Springer.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C Wallace. 2019. Eraser: A benchmark to evaluate rationalized nlp models. *arXiv preprint arXiv:1911.03429*.

Jennifer Golbeck. 2012. The twitter mute button: a web filtering challenge. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2755–2758.

Sheng Guo and Naren Ramakrishnan. 2010. Finding the storyteller: automatic spoiler tagging using linguistic cues. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 412–420.

Yang Li, Xin Robert Luo, Kai Li, and Xiaobo Xu. 2022. Exploring the spoiler effect in the digital age: Evidence from the movie industry. *Decision Support Systems*, 157:113755.

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc.

Satoshi Nakamura and Katsumi Tanaka. 2007. Temporal filtering system to reduce the risk of spoiling a user's enjoyment. In *Proceedings of the 12th international conference on Intelligent user interfaces*, pages 345–348.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016a. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016b. "why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 1135–1144.

Mengting Wan, Rishabh Misra, Ndapa Nakashole, and Julian McAuley. 2019. Fine-grained spoiler detection from large-scale review corpora. *arXiv preprint arXiv:1905.13416*.

Anna Wróblewska, Paweł Rzepiński, and Sylwia Sysko-Romańczuk. 2021. Spoiler in a textstack: How much can transformers help? *arXiv preprint arXiv:2112.12913*.