Clickbait Spoiler Classification and detection using Transformers

Naman Khurpia, Mohammed Junaid Shaik, Mamatha Yarramaneni
School Engineering and Applied Sciences
State University of New York at Buffalo

Abstract

The two objective of this paper is spoiler classification and spoiler generation of three different sorts of spoilers Phrase, Passage, and Multi. We have devided the two objectives as task 1 and task 2 of the experiment. As you can see here, our specialized train function, which is based on a number of models, provides us with final spoiler type predictions. For spoiler generation we have different model that generated the correct type of spoiler using unique Huggingface transformer. For task 1, we tried seven models, including bert-base-cased, bert-large, distilbert-base-cased, roberta-base, roberta-large, deberta-base, and deberta-large, with roberta-large coming out on top. For Task 2, we trained models like distilbert-base-cased-distilled-squad, all-MiniLM-L6-v2, obrizum/all-MiniLM-L6-v2, roberta-base, sentence-transformers/quora-distilbert-base, Bert-large-uncased-whole-word-masking-finetuned-squad, deberta-v3-base-squad, roberta-base-squad2 and msmarco-bert-base-dot-v5 in which roberta-base-squad2, all-MiniLM-L6-v2 and msmarco-bert-base-dot-v5 were the best models for various types of spoilers. Later, the remaining findings are contrasted with the baseline outcomes of simple models and the ACL paper, which we utilized as the primary project objective.

1 Introduction

The challenge "Clickbait Detection and generation of spoiler" aims to create and test automated tools for spotting clickbait headlines on news websites and social media platforms. The term "clickbait" describes headlines that rely on sensationalism, exaggeration, or false information to persuade readers to click on a link. This baits the user to make them click on it. The task gives us a dataset of headlines, text inside, human spoiler, and some keywords and text that has been classified as clickbait or not, and we must build a machine learning model that can correctly identify fresh headlines as clickbait or not. The aim is to enhance the capacity to differentiate between legitimate news and clickbait.

We have developed a pipeline to preprocess the data, train several models using huggingface transformers, and then utilize the preprocessed data to provide spoilers for the posts that were essentially clickbait. In milestone 2, we produced findings using only vanilla models. For milestone 3, we added our own pipelines to further improve these results. We made an effort to provide results that were as near as feasible to the state-of-the-art results shown in the ACL article that we were given.

2 Related Work: A literature survery

We consulted many publications and papers for the understanding of the task, breaking the task into smaller sub problems and coming up with the creation of process and pipe-lining the entire experiment, managing multiple models and generating results in compliance with each of them.

- Clickbait detection and the personalized blocking by Chakraborty et al. (2016)
- Clickbait Challenge 2017, Regression Model for Clickbait detection and strength by Potthast et al. (2018)
- Headline Generation for Clickbait Detection by Shu et al. (2018)
- Headline Generation for Clickbait Detection using auto-tuned reinforcement learning by Xu et al. (2019)
- Clickbait Spoiler generation introduced by Hagen et al.(2022)
- Thai Clickbait Detection Algorithms Using Natural Language Processing with Machine Learning Techniques

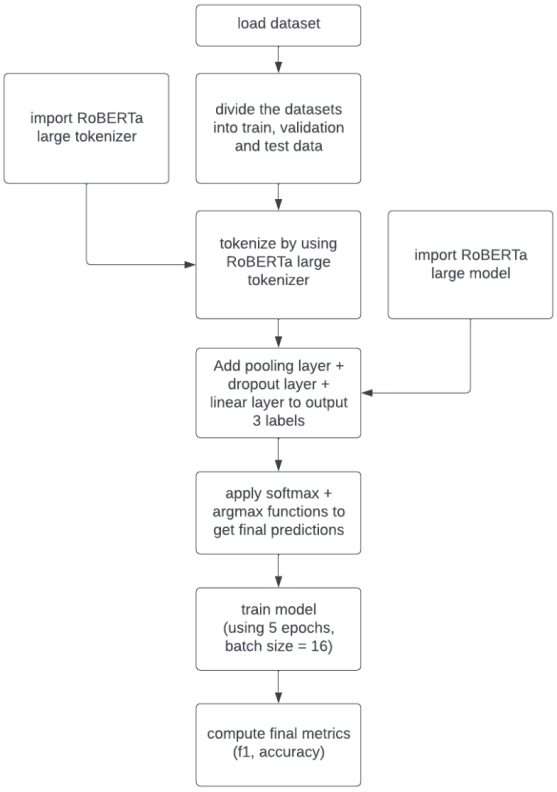

Figure 1: Task one Architecture - Classification of type of clickbait

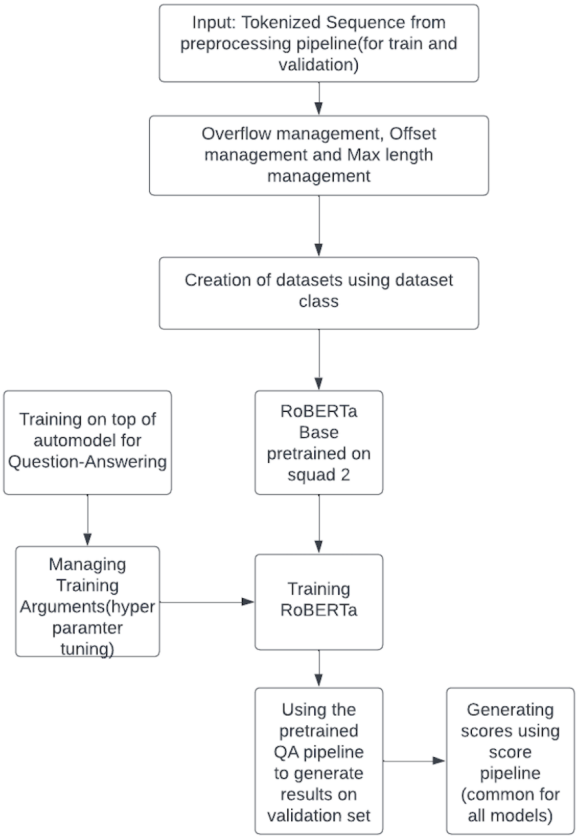

Figure 2: Task two Architecture - Generation of tokens for phrase type spoiler

- Question-Answering Models
- Clickbait spoiling as a question-answering problem
- SQuAD v1.1 (107,785 questions and answers) and TriviaQA (95000 questions and answers)
- BERT, Big Bird, DeBERTa, ELECTRA, MPNet, RoBERTa
- Passage Retrieval
- Benchmark - MS MARCO
- MonoBERT, MonoT5, BM25, Query Likelihood

3 Methods/Model Architecture

3.1 Task one (Spoiler type classification)

Refer Figure 1 for this process. The dataset is imported and divided into a train, test, and validation set, we used test train split library for this, the question provided us with validation set to calculate the accuracies with. Each dataset has 800 records for testing, 640 records for validation, and 2560 records for training. The Roberta big is the final clickbait classification model. The model is imported, and its ultimate dimension is decreased to three to match the output by adding the pooling layer, dropout layer, and linear layers. The pre-trained model is then adjusted for our situation using the newly learned models. A batch size of 16 is used for both the training and validation phases of the procedure, the learning rate is set at 2e-5. The model's weight decay is set at 0.01. Every checkpoint is saved during the five epochs of training for this model. The very last and finest checkpoint is loaded at the conclusion. The outcomes of the test dataset are predicted using this final model. The completed model's predictions are then processed using the softmax and argmax routines. Finally, parameters like accuracy and f1 are calculated.

3.2 Task two (Generation of tokens for phrase type spoiler)

The dataset is initially imported and divided into a train, test, and validation set using the library

test train split from sklearn. Each dataset has 800 records for testing, 640 records for validation, and 2560 records for training. The Roberta base, which was pretrained using the SQUAD 2.0 dataset, is the final model for the production of clickbait phrase spoilers. Tokenizing input text into tokens is done using the appropriate model's tokenizer. The pre-trained model is then adjusted for our situation using the newly learned models. A batch size of 2 is used for both the training and validation phases of the procedure, and the learning rate is set at 2e-5. The model's weight decay is set at 0.01. Every checkpoint is saved during the two epochs of training for this model. Refer Figure 2 for understanding the entire architecture step by step.

3.3 Task two (Passage retrival using cosine similarities)

We use sentence transformer's use cosine similarity method to calculate the cosine similarity between input question which is the posttext and the sentences in the context which are targeted paragraphs and then we rank these sentences in descending order, finally return maximum similarity sentence. Finally, metrics like BERT, BLEU and METEOR scores are computed. Refer Figure 3 for clear understanding of architecture.

3.4 Task two (Multi type spoiler token generation)

We used a sentence transformer model which is pre-trained on MSMARCO dataset, which translates the input to a 768-dimensional dense vector space. The process for generation of multi type spoiler is no different than for phrases which we discussed in 3.2 subsection. First we train our model using the same method, but during prediction we pass the context as sentences being splitted from targeted paragraphs. Answers generated have a threshold value which are added to the final result if the probability score is greater than 5 percent. Refer Figure 4 for clear understanding.

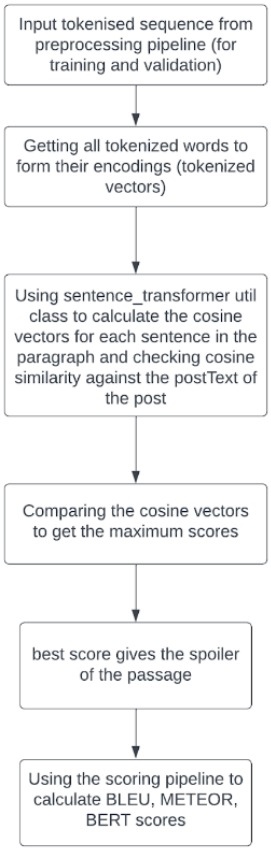

Figure 3: Task two Architecture - Paragraph similarity calculation and retrieval

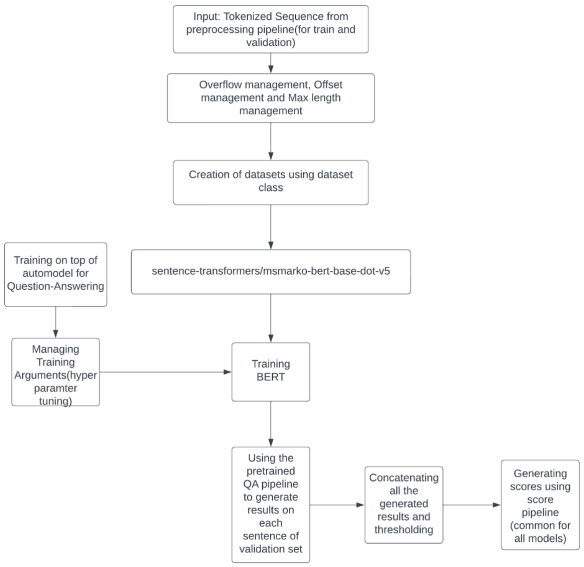

Figure 4: Task two Architecture - Multi type spoiler, generation of multiple tokens

4 Results and comparison

4.1 Results of our experiment

4.1.1 Spoiler classification

Here we can see that Roberta large outperformed any other model. We achieved an accuracy of 73 percent and F1 score of 73 percent. Roberta large is a large size variant of birth, which is pre-trained model that has a chief state of the art performance

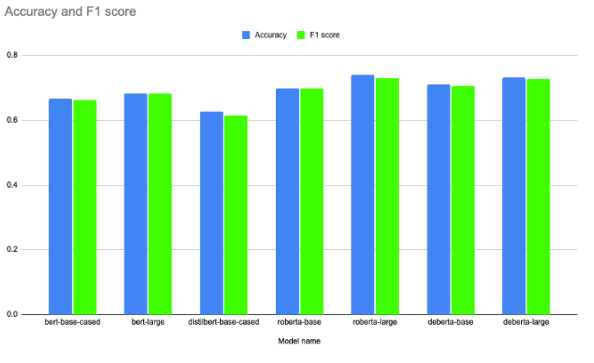

Figure 5: Task one results - classification of spoiler type

Model name	Accuracy	F1 score
bert-base-cased	0.66625	0.6618399351725451
bert-large	0.68125	0.6821537241704321
distilbert-base-cased	0.62625	0.6136566019433247
roberta-base	0.6975	0.696888314371547
roberta-large	0.73875	0.7302641134426479
deberta-base	0.70875	0.7056227340408437
deberta-large	0.7325	0.7276887505552079

Figure 6: Task one results - classification of spoiler type

on wide range of NLP task. Overall, the benefits of using Roberta large has improved performance, transfer learning, large capacity and multilingual support. These make it an excellent choice for any NLP task where we need to classify on the basis of multiple tokens. Refer Figure 5 and 6.

4.1.2 Phrase spoiler generation

Here we can see that Roberta-base-squad-d2 outperformed any other model we got BERT precision score of 90, BERT recall score of 89, BERT F1 scores of 90, BLEU score of 49 and METEOR your score of 38. For phrases, Deberta also performed slightly similar to Roberta. This Roberta-base-squad-d2 was pretrained on SQUAD dataset. Refer Figure 7 and 8.

4.1.3 Passage spoiler generation

all-MiniLM-L6-v2 outperformed any other model we got BERT precision score of 85 percent, BERT recall score of 85 percent, BERT F1 scores of 85 percent, and METEOR your score of 25 percent. For passages, even calculating dot scores was a good approach and we got similar results but cosine vectors were more meaningful as per conclusions. Refer image 9 and 10.

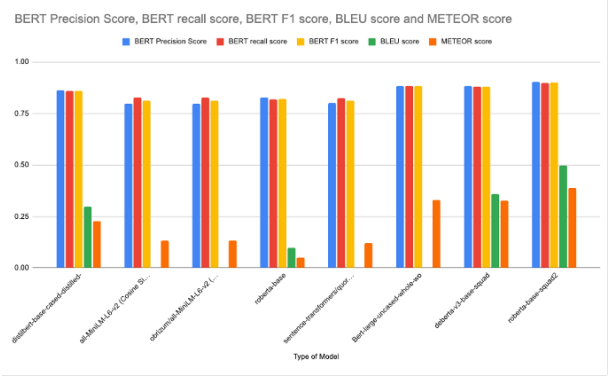

Figure 7: Task two results - generation of phrase type spoiler

Type of Model	BERT Precision Score	BERT recall score	BERT F1 score	BLEU score	METEOR score
distilbert-base-cased-distilled-squad (with finetuning)	0.8611	0.861	0.8601	0.2963	0.2274
all-MiniLM-L6-v2 (Cosine Similarities)	0.7992	0.8281	0.8127	5.52E-233	0.1324
obrizum/all-MiniLM-L6-v2 (Dot scores)	0.7992	0.8281	0.8127	4.54E-232	0.1324
roberta-base	0.827	0.8184	0.8216	0.09807566	0.0491804
sentence-transformers/quora-distilbert-base	0.8019	0.8244	0.8123	4.18E-232	0.1195
Bert-large-uncased-whole-word-masking-finetuned-squad (cosine similarity)	0.883	0.8823	0.8821	1.42E-233	0.32956
deberta-v3-base-squad	0.8824	0.8811	0.881	0.35779428	0.3261704
roberta-base-squad2	0.9046	0.8972	0.9002	0.49842812	0.3897169

Figure 8: Task two results - generation of phrase type spoiler

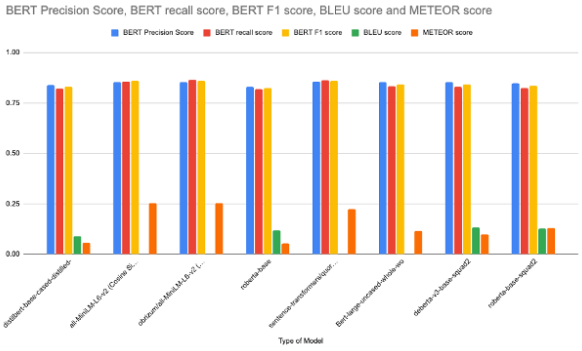

Figure 9: Task two results - passage retrieval

Type of Model	BERT Precision Score	BERT recall score	BERT F1 score	BLEU score	METEOR score
distilbert-base-cased-distilled-squad (with finetuning)	0.8403	0.8213	0.8304	0.0868	0.0569
all-MiniLM-L6-v2 (Cosine Similarities)	0.854	0.8565	0.8594	5.52E-233	0.2522
obrizum/all-MiniLM-L6-v2 (Dot scores)	0.854	0.8656	0.8594	5.52E-233	0.2522
roberta-base	0.8314	0.8177	0.8243	0.11739245	0.0516242
sentence-transformers/quora-distilbert-base	0.8558	0.8631	0.8592	4.41E-233	0.2229
Bert-large-uncased-whole-word-masking-finetuned-squad (cosine similarity)	0.8528	0.8339	0.8429	4.96E-236	0.11453
deberta-v3-base-squad2	0.8546	0.8314	0.8423	0.13267946	0.0956198
roberta-base-squad2	0.8492	0.8235	0.8358	0.12564868	0.1282661

Figure 10: Task two results - passage retrieval

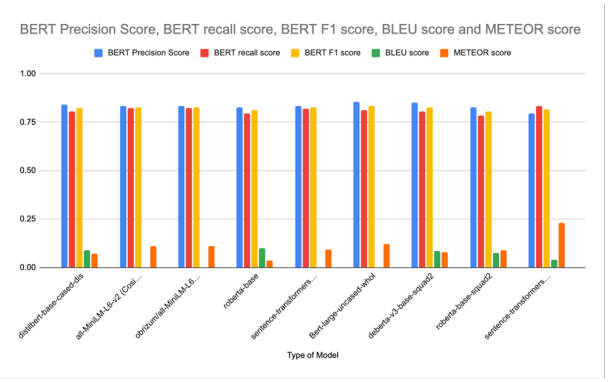

Figure 11: Task two results - generation of multi type spoiler

Type of Model	BERT Precision Score	BERT recall score	BERT F1 score	BLEU score	METEOR score
distilbert-base-cased-distilled-squad (with finetuning)	0.8402	0.8047	0.8217	0.0884	0.0728
all-MiniLM-L6-v2 (Cosine Similarities)	0.8319	0.8223	0.8268	6.62E-233	0.1116
obrizum/all-MiniLM-L6-v2 (Dot scores)	0.8319	0.8223	0.8268	6.62E-233	0.1116
roberta-base	0.8278	0.7962	0.8113	0.0979553497	0.03553560362
sentence-transformers/quora-distilbert-base	0.8321	0.8182	0.8249	4.39E-233	0.0936
Bert-large-uncased-whole-word-masking-finetuned-squad (cosine similarity)	0.8547	0.8132	0.833	2.18E-234	0.1213
deberta-v3-base-squad2	0.8519	0.8054	0.8275	0.0854094376	0.07868105274
roberta-base-squad2	0.8268	0.7834	0.8039	0.0740118560	0.08734685195
sentence-transformers	0.7946	0.8352	0.8141	0.0404891681	0.2292876338

Figure 12: Task two results - generation of multi type spoiler

4.1.4 Multi spoiler generation

Sentence transformer -msmarco model outperformed any other model we got BERT precision score of 79 percent, BERT recall score of 83 percent, BERT F1 scores of 81 percent, and METEOR your score of 22 percent. For multi, none of the other models were even close to this approach. Refer image 11 and 12.

4.2 Comparison of Baseline results (milestone 2) and milestone 3 results

4.2.1 Comparison between baseline results and current results - for spoiler type classification

we can see that for task one, baseline results were significantly improved in milestone 3. Using Roberta Larch had a lot of benefits, we gained an accuracy of 73 percent and F1 score of 73 percent. Roberta large was pre-trained on a large corpus of text data than BERT base, specifically Roberta large was trained on 160GB of text data hence performs better classification. Refer Figure 13 and 14.

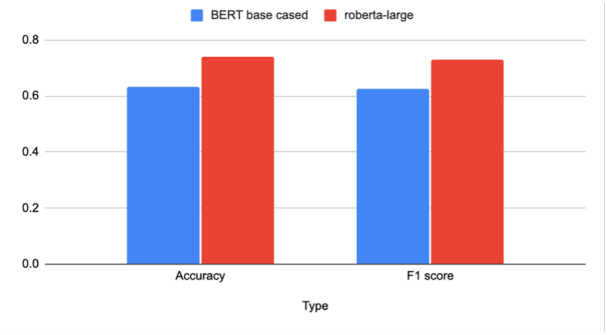

Figure 13: Task one results - Comparison between baseline results and current results - for spoiler type classification

Model name	Accuracy	F1 score
BERT base cased	0.63125	0.6242281795
roberta-large	0.73875	0.7302641134

Figure 14: Task one results - Comparison between baseline results and current results - for spoiler type classification

4.2.2 Comparison between baseline results and current results - phrase spoiler generation

Roberta and distill bert both have same architecture, but Roberta base squad 2 was trained on SQUAD2 dataset which gives it an upperhand. We improved out accuracies by a lot, we have almost 16 percent improved performance as compared to baseline results, we improved the meteor score by 16 percent. Refer Figure 15 and 16.

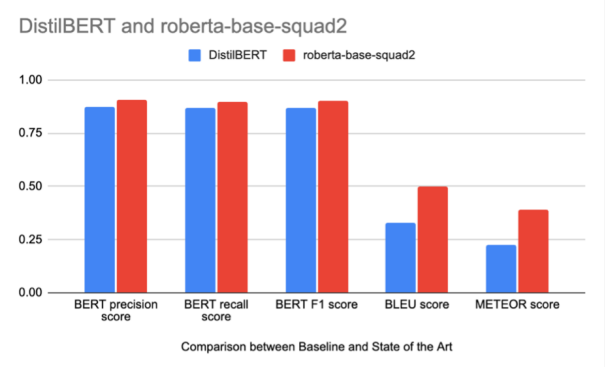

Figure 15: Task two results - Comparison between baseline results and current results - phrase spoiler generation

Type	BERT precision sc	BERT recall score	BERT F1 score	BLEU score	METEOR score
DistilBERT	0.8717	0.8708	0.8704	0.3277	0.2225
roberta-base-squad2	0.9046	0.8972	0.9002	4.98E-01	0.3897169587

Figure 16: Task two results - Comparison between baseline results and current results - phrase spoiler generation

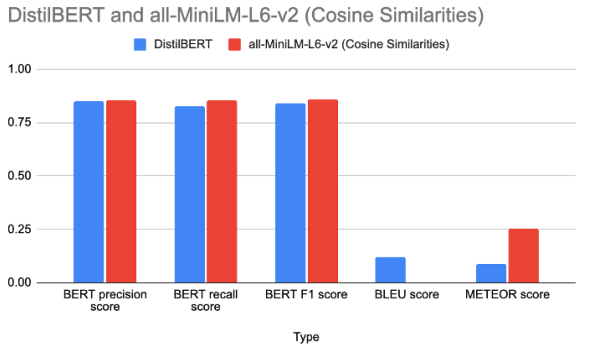

Figure 17: Task two results - Comparison between baseline results and current results - passage spoiler generation

4.2.3 Comparison between baseline results and current results - passage spoiler generation

For passage we changed our model and approach for finding the best paragraph, we used distillbert in milestone 2 but now we are comparing cosine similarity vectors which gives us better scores, calculating cosine similarities makes more sense as well, since we can compare synonyms and similar words. Refer Figure 17 and 18.

4.2.4 Comparison between baseline results and current results - multi spoiler generation

DistillBERT and sentence transformer both are based on transformer based architecture but, sentence transformer is pretrained on MSMARCO dataset which is specifically designed for information retrieval and question answering task. DistillBERT is significantly smaller than sentence transformer. Refer Figure 19 and 20.

Type	BERT precision sc	BERT recall score	BERT F1 score	BLEU score	METEOR score
DistilBERT	0.8518	0.8273	0.839	0.1209	0.0868
all-MiniLM-L6-v2 (Cosine Similarities)	0.854	0.8565	0.8594	5.52E-233	0.2522

Figure 18: Task two results - Comparison between baseline results and current results - passage spoiler generation

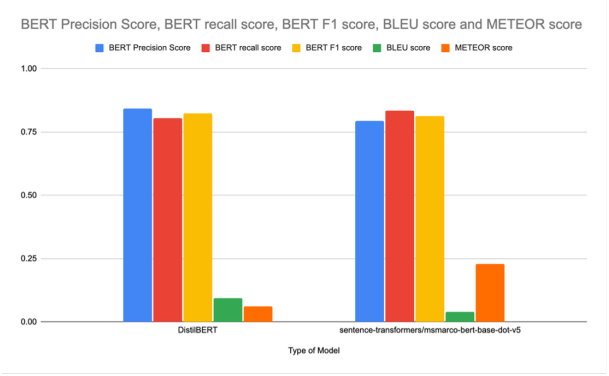

Figure 19: Task two results - Comparison between baseline results and current results - multi spoiler generation

Type of Model	BERT Precision Score	BERT recall score	BERT F1 score	BLEU score	METEOR score
DistilBERT	0.8415	0.806	0.823	0.0929	0.0607
sentence-transformers	0.7946	0.8352	0.8141	0.0404891681	0.2292876338

Figure 20: Task two results - Comparison between baseline results and current results - multi spoiler generation

4.3 Comparison of State of the Art results and our results

4.3.1 Comparison between State of the art results and our results - for spoiler type classification

We have chosen the results from the ACL paper that has the highest METEOR score to compare our model with. We see that our model has a Accuracy of 0.738 which is nearly 6.6 percent lower than the ACL paper result. Refer Figure 21 and 22.

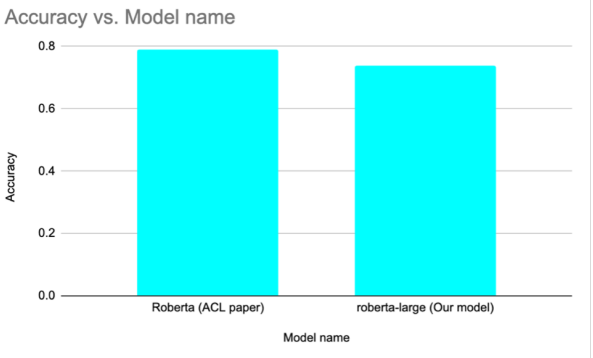

Figure 21: Task one results - Comparison between State of the art results and our results - for spoiler type classification

Model name	Accuracy
Roberta (ACL paper)	*0.7912*
roberta-large (Our model)	0.73875

Figure 22: Task one results - Comparison between State of the art results and our results - for spoiler type classification

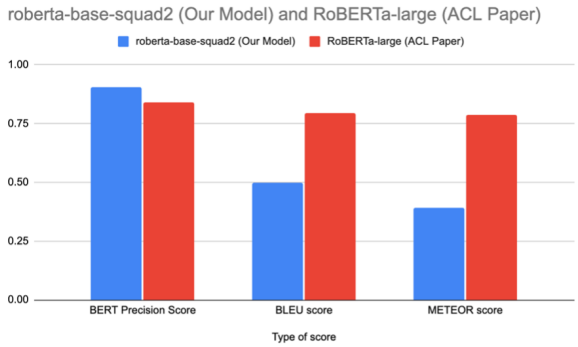

Figure 23: Task two results - Comparison between State of the art results and our results - phrase spoiler generation

4.3.2 Comparison between State of the art results and our results - phrase spoiler generation

In this task, first the phrase type spoilers have been compared and in this we compare the BERT precision score, the BLEU score and the METEOR score and we see that our model under performs by one-thirds in the BLEU score and nearly half in the METEOR score, but over performs in the BERT score by nearly 7 percent. Refer Figure 23 and 24.

4.3.3 Comparison between State of the art results and our results - passage spoiler generation

In passage type spoilers we see that there is a huge difference between the model we chose and our model in all the three scores.Refer Figure 25 and 26.

Type of Model	BERT Precision Score	BLEU score	METEOR score
roberta-base-squad2 (Our Model)	0.9046	0.4984281217	0.3897169587
RoBERTa-large (ACL Pa	*0.8404*	*0.7947*	*0.7861*

Figure 24: Task two results - Comparison between State of the art results and our results - phrase spoiler generation

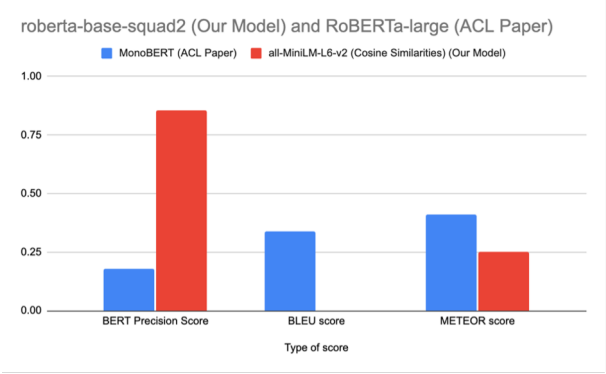

Figure 25: Task two results - Comparison between State of the art results and our results - passage spoiler generation

Type of Model	BERT Precision Score	BLEU score	METEOR score
MonoBERT (ACL Paper)	0.18	0.34	0.41
all-MiniLM-L6-v2 (Cosine Similarities) (Our Model)	0.854	5.52E-233	0.2522

Figure 26: Task two results -Comparison between State of the art results and our results- passage spoiler generation

4.3.4 Comparison between State of the art results and our results - multi spoiler generation

For multi the ACL paper did not had any scores, so we are claiming to be the new state of the art and since we have a meteor score of 22 percent which is very impressive given the ambiguities that exist in the dataset. Refer Figure 27.

5 Discussion and Error Analysis

Overall, for task one we achieved similar accuracy as compared to State of the Art Results, probably one of the best scope of improvement would be tokenizing the series of sequences, and after analyzing the part of speech, we can use a better, three pre-trained model, and then perform

Type of Model	BERT Precision Score	BLEU score	METEOR score
No results from Paper	NA	NA	NA
sentence-transformer MSMARCO	0.79	0.04	0.22

Figure 27: Task two results - Comparison between State of the art results and our results - multi spoiler generation

fine-tuning to achieve better scores. After testing almost 7 models, we realized that these errors are erroneous, and in some passages spoilers can be represented in form of phrases and some passages can be represented as multi type spoiler. Hence, there was ambiguity in the task 1 classification. This ambiguity can be resolved if we had an improved dataset where the quantity of each type of spoiler with similar count.

Currently we have a lot of phrases spoiler-posttext example so training is easier and therefore the results are better as well. The count of multi example are very small and the accuracy is probably low because we tuning the hyper-parameter - threshold value was very tricky, having the same threshold for all type of data is not correct.

Apart from these inconsistencies, we can use a better pre-processing pipeline, which accurately labels, and performs stemming, lemmatization, and tokenization for the correct tokens. Incorporating more context is another approach. We can take where we can replay similar words with synonyms and short forms to full forms and replace any slangs to the original word. Using a more advanced model like GPT can be used for question answering system. Developing a feedback loop can help us improve the accuracy since the model will be feeding it to its answer and asking for feedback which can be used to improve the model performance.

6 Conclusions

In total we can conclude that, spoiler classification depends on the type of spoiler being expected, since the type is very ambiguous achieving an accuracy of 90 percent would be very challenging task. With respect to task 2 achieving a high accuracy for multi, phrases, passage is possible, but as mentioned in the results and conclusion that we need a higher count of data set, the current data set with 4000 rows of data was very limited, both in terms of count, and in terms of tokens of words, we need a data set with us which has a wide variety of tokens and ambiguous statements. For phrases, we have a competitive result, and for passage, we have a very unique approach, which is very fast in execution, for multi type spoiler we are giving the first of its solution with a good accuracy.

7 Contributions

Name	Contributions
Naman Khurpia	Preprocessing the input, Implementation of Training the dataset for DistillBERT, Cosine similarities and dot scores for the best passage retrieval methods. Creating a scoring pipeline for BLEU, METEOR, BERT scores for task 2.
Mohammad Junaid Shaik	Implementing the input pipeline for preprocessing,Implementation of Training the dataset for roberta-base, deberta-base and Bert-large-uncased for the best phrase retrival. Validating the scoring pipeline using vanilla models.
Mamatha Yarramaneni	Preprocessed the dataset for task1 and task2. Implemented task1 using classical models and neural models. Implemented clickbait spoiler generation using IR methods. Built the best neural model for task 1 and for multi spoiler generation

Table 1: Contribution Table

References

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I. (2017). Attention is all you need. In Advances in Neural Information Processing Systems (pp. 6000-6010).

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

He, P., Liu, X., Gao, J. and Chen, W. (2020). Deberta: Decoding-enhanced bert with disentangled attention. arXiv preprint arXiv:2006.03654.

Hagen, M., Fröbe, M., Jurk, A. and Potthast, M. (2022). Clickbait spoiling via question answering and passage retrieval. arXiv preprint arXiv:2203.10282.

Reimers, N. and Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084.

Hugging Face. (n.d.). Transformer

Models: Sentence-Transformers/multi-qa-MiniLM-L6-cos-v1. https://huggingface.co/sentence-transformers/multi-qa-MiniLM-L6-cos-v1.

scikit-learn. (n.d.). TfidfVectorizer. https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html.

Hugging Face Forum. (n.d.). Trainer only doing 3 epochs no matter the TrainingArguments. https://discuss.huggingface.co/t/trainer-only-doing-3-epochs-no-matter-the-trainingarguments/19347/5.

Hugging Face. (n.d.). Preprocessing Input Data: Question Answering. https://huggingface.co/docs/transformers/tasks/question_answering#preprocess.

Towards Data Science. (n.d.). Fine-tune Transformer Models for Question-Answering on Custom Data. https://towardsdatascience.com/fine-tune-transformer-models-for-question-answering-on-custom-data-513eaac37a80.

Hugging Face. (n.d.). Custom Datasets. https://huggingface.co/transformers/v3.3.1/custom_datasets.html#qa-squad.

Towards Data Science. Question Answering with Pretrained Transformers using PyTorch. https://towardsdatascience.com/question-answering-with-pretrained-transformers-using-pytorch-c3e7a44b4012.

Towards Data Science. https://github.com/skandavivek/transformerQA-finetuning

Zhang, Tianyi, Kishore, Varsha, Wu, Felix, Weinberger, Kilian Q., and Artzi, Yoav. "BERTScore: Evaluating Text Generation with BERT." In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1499–1510, Online, 2020. Association for Computational Linguistics. arXiv preprint arXiv:1904.09675.

Banerjee, Satanjeev and Lavie, Alon. "METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments." In Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, pages 65-72, Ann Arbor, Michigan, USA, 2005. Association for Computational Linguistics.

Papineni, Kishore, Roukos, Salim, Ward, Todd, and Zhu, Wei-Jing. "BLEU: A Method for Automatic Evaluation of Machine Translation." In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), pages 311-318, Philadelphia, Pennsylvania, USA, 2002. Association for Computational Linguistics.

Wolf, Thomas, Debut, Lysandre, Sanh, Victor, Chaumond, Julien, Delangue, Clement, Moi, Anthony, Cistac, Pierric, Rault, Tim, Louf, Rémi, Funtowicz, Morgan, Davison, Joe, Shleifer, Sergey, von Platen, Clara, Ma, Caglar, Jernite, Yacine, Plu, Jean-Baptiste, Xu, Canwen, Le Scao, Yoann, Gugger, Sylvain, Drame, Quentin, Lhoest, Quentin, and Rush, Alexander M. "HuggingFace's Transformers: State-of-the-art Natural Language Processing." In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 38-45, Online, 2020. Association for Computational Linguistics. arXiv preprint arXiv:1910.03771.