PrefScore: Pairwise Preference Learning for Reference-free Single-document Summarization Quality Assessment

Anonymous ACL submission

Abstract

Evaluating machine-generated summaries without a human-written reference summary has been a need for a long time. Inspired by preference labeling in existing works of summarization evaluation, we propose to judge summary quality by learning the preference rank of summaries using the Bradley-Terry power ranking model from generated inferior summaries of a base summary. Despite the simplicity of our method, extensive experiments on several datasets show that our weakly supervised scheme can produce scores highly correlate with human ratings.

1 Introduction

Summarization is an active field in natural language processing where researchers develop systems to automatically generate summaries for articles. The best way to evaluate the quality of system-generated summaries is to let human assessors score them. However, human evaluation is non-trivial and laborious, and thus leads to the birth of many automatic evaluation metrics.

Existing summarization quality metrics can be categorized as reference-based ones and reference-free ones, depending on whether reference summaries are needed in the evaluation stage. Reference-based metrics include ROUGE (Lin, 2004), BLEU (Papineni et al., 2002), CIDEr (Vedantam et al., 2015), METEOR (Banerjee and Lavie, 2005), S^3 (Peyrard et al., 2017), MoverScore (Zhao et al., 2019), BertScore (Zhang* et al., 2020), etc. This kind of metrics calculate the lexical overlap or the embedding similarity between a system-generated summary and its corresponding human-written reference summary. Reference-based metrics are reported having high correlation with human assessed scores but the process for human creating reference summaries is laborious and expensive.

Thus, recent works are shifting to reference-free metrics. SummaQA (Scialom et al., 2019) and BLANC (Vasilyev et al., 2020) leverage pretrained language models to carry out text understanding tasks to evaluate the helpfulness of a summary for understand the source article. While SUPERT (Gao et al., 2020b) measures the semantic similarity against a pseudo reference summary extracted from source articles. However, reference-free metrics may show a lower correlation (Fabbri et al., 2020) with human evaluation scores than some of the reference-based metrics. In addition, these unsupervised or self-supervised schemes may introduce extra noise to the evaluation. For example, SummaQA relies on a QA system, but a well trained QA can still make mistakes.

To trade off between the human effort needed and the quality of evaluation, some works pursue a pairwise preference approach which collects preference labels over sentences or summaries from a human assessor as it places a lower cognitive burden than writing a reference summary or manually scoring a machine-generated summary. Zopf (2018) proposes a reference-free evaluation approach by estimating sentence-level preferences on source documents rather than directly on the generated summaries. Gao et al. (2020a) train a linear model to estimate a summary preference utility function via active preference learning to guide a reinforcement learning based summarization system. But they do not examine the learned preference model as a metric for summarizaiton evaluation.

Inspired by human-involved pairwise preference in summarization evaluation (Zopf, 2018; Gao et al., 2020b) and simple NLP data augmentation methods like EDA (Wei and Zou, 2019), in this work, we explore reference-free summary quality assessment via pairwise preference learning using negative sampling. A pre-trained text embedding model is used in a siamese network to learn the preference utility in an end-to-end, weakly supervised

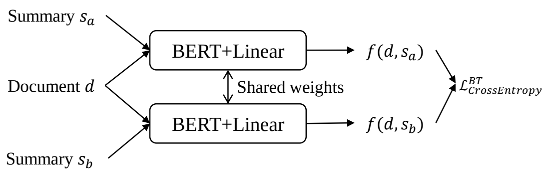

Figure 1: Model architecture

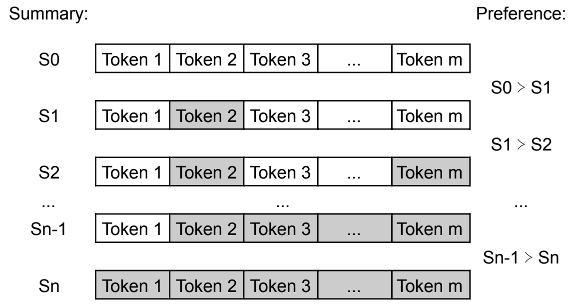

Figure 2: An example of negative sampling process. The original part is in white while the modified part is indicated as grey block.

fashion. The closest work to ours is LS_Score (Wu et al., 2020), however, our method is different from LS_Score in:

1. We use a simple network architecture targeting overall score instead of separately design different modules for different aspects of score.
2. Using the Bradley-Terry (Bradley and Terry, 1952) power ranking model, cross entropy loss is applied for estimating overall rank utilities rather than the contrastive loss for discriminating good summaries and bad summaries.
3. Our mixed negative sampling method allows rank learning over reference summary and generated negative samples while LS_Score does not differentiate within negative samples.

We show that the learned models are competitive compared to the state-of-the-art reference-free metrics. Our code and pretrained models are at `https://anonymous.4open.science/r/PrefScore-7C63/`.

2 Method

2.1 Model Architecture

The goal of a reference-free evaluation system is to learn a regressor f which takes a document d and its summary s as input and produce a score $f(d, s)$ which represents the quality of the summary s. Learning such a regressor via supervised learning is not applicable here. Because the supervised model is prone to overfitting if directly trained on the limited size of human rated summarization evaluation datasets.

Instead, our method uses pairwise preference learning as a workaround. An inferior summary can be obtained by perturbing a summary. This enables existing summarization datasets (no human ratings as training labels, but only gold, reference summaries) to be transformed into massive training data for preference learning.

The training label is designed based on the Bradly-Terry (BT) model (Bradley and Terry, 1952). Specially, given two summaries s_a and s_b of the document d, the BT model estimates $f(d, s_a)$ and $f(d, s_b)$ such that the probability of s_a being superior than s_b is:

$$p(s_a \succ s_b) = \frac{\exp(f(d, s_a))}{\exp(f(d, s_a)) + \exp(f(d, s_b))}. \quad (1)$$

This leads to our model design (Figure 1) using a siamese network. Leveraging the recent work of BERT-like (Devlin et al., 2019) contextualized embedding, a document d and a summary s are viewed as two sequence of tokens T_d and T_s. The input sequence are constructed as ([CLS], T_d, [SEP], T_s, [SEP]), then the output of the [CLS] token containing both information from document and summary will be sent to a linear layer to produce the final score $f(d, s)$. During the training, a pair of summaries will be send to the siamese network, it can be seen as training a classifier to determine which summary is better. A cross-entropy loss is applied therefore:

$$\begin{aligned}\mathcal{L}^{BT} = -\sum_{d}\sum_{s_a, s_b} &[y_{s_a, s_b} \log(p(s_a \succ s_b)) \\ &+ (1 - y_{s_a, s_b}) \log(p(s_b \succ s_a))] \quad (2)\end{aligned}$$

where y_{s_a, s_b} is the preference label for the summary pair s_a and s_b. The learned ranking utility f is used as our summary evaluator and it does not require a reference summary in the test/evaluation stage.

2.2 Negative Sampling

We generate perturbed summaries for learning the preference ranking by modifying a base summary s_0 to deviate it from its original semantics. Denote the deviated summary as s_1. By iteratively applying the purtubation modificaiton to s_i to generate a more deviated summary s_{i+1}, we obtain a sequence of preferred summaries $s_0 \succ s_1 \succ \cdots \succ s_n$. The process is illustrated in Figure 2. In each iteration, one or more unmodified tokens in s_i is randomly selected and mutated to generate summary s_{i+1}. The process continues until all tokens have been modified.

Specifically, we have implemented three mutation methods: 1) **deleting a sentence** from the summary, resulting in information loss in the summary. 2) **replacing a sentence** in the summary with a sentence from other summaries, introducing extra information and redundancy in the summary. 3) **deleting a word** from the summary, influencing the sentence structure and readability. By using a mixture of these methods, i.e., randomly selecting a mutation method in each iteration, the model should learn an overall score for different aspects in summarization task.

3 Experiments

3.1 Test sets

There are not many datasets with human evaluations to machine-generated summaries. Unfortunately, they are almost all in the news article domains. We use three established ones:

TAC2010 (NIST, 2010) is a multi-document summarization dataset which reports three scores: content, fluency and overall. For a summary, we calculate the mean score for all documents paired with the summary as an extend for our metric in the multi-document scenario. Only Set A for regular summarization task is used here.

Newsroom (Grusky et al., 2018) is a single-document summarization dataset reporting four scores: INFormativeness, RELevance, COHerence and FLUence. Each document-summary pair is rated by three human annotators. We use their mean score as the groundtruth score.

RealSumm (Bhandari et al., 2020), a recent single-document dataset reporting the LitePyramid (Shapira et al., 2019) score which is also content focused.

3.2 Training sets (documents and reference summaries only, no human evaluations)

Because the test sets are in the news article domain, we deliberately select training sets from different domains except the news article domain, to test the robustness and transferability of our methods. For every original, reference summary in the training sets, 5 negative samples (inferior summaries) are generated.

The train split of three datasets are used separately to train our model: **Billsum** (Kornilova and Eidelman, 2019) collects the summraization of legislative bills. **Scientific papers-ArXiv** (Cohan et al., 2018) dataset contains abstracts and articles from arXiv. **Big-Patent** (Sharma et al., 2019) consists of patent documents along with human written summaries.

3.3 Baselines and upperbounds

We compare our work with both reference-free and reference-based metrics. The recently developed SummaQA (Scialom et al., 2019), BLANC (Vasilyev et al., 2020) and SUPERT (Gao et al., 2020b) are our baselines because they are reference-free[1]. Reference-based metrics serve as soft upper bounds because they are provided with extra human guides which are reference summaries. ROUGE (Lin, 2004), BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), S^3 (Peyrard et al., 2017), MoverScore (Zhao et al., 2019), BertScore (recall) (Zhang* et al., 2020) are included in this study.

Results for LS_Score (Wu et al., 2020) is only reported for Newsroom, which is copied from their paper, as we have not succeed in reproducing their model using their code to test on other datasets [2].

3.4 Settings

For a fair comparison, we use the same pretrained language model BERT used by baselines. Specifically, we use bert-base-uncased variant of the BERT model in HuggingFace Transformer's Pytorch implementation. An input sequence is rounded to 512 tokens using round robin trimmer. We fine tune the model on NVIDIA RTX 3090 with 1 epoch using the Adam optimizer with a learning rate of 1e-5 and a batch size of 12.

3.5 Results

We use the summary-level (Peyrard et al., 2017) meta evaluation strategy to report an approach's average correlation with human ratings over summaries. Considering the page limit and that our

[1]By "reference-free", we mean that a reference summary is not needed to judge a machine-generated summary.

[2]Several other researchers reported the same issue `https://github.com/whl97/LS-Score/issues`

Table 1: Spearman's Correlation on TAC2010.

	Content	Fluency	Overall
Our approach			
Trained w/ Billsum	**0.5048**	**0.4158**	**0.4871**
Trained w/ ArXiv	0.4735	0.3334	0.4391
Trained w/ BigPatent	0.4504	0.2632	0.4132
Reference-free Baselines			
BLANC-tune	0.4272	0.2943	0.3966
SummaQA-F1	0.3007	0.2431	0.2864
SummaQA-CFD	0.2905	0.1516	0.2620
SUPERT	0.4794	0.3241	0.4266
Reference-based upper bounds			
R-1	0.5597	0.2570	0.5025
R-2	0.6448	0.3490	0.5894
R-L	0.5032	0.1772	0.4463
MoverScore	0.7213	0.3522	0.6453
BertScore	0.6769	0.3634	0.6162
BLEU	0.6018	0.3462	0.5636
METEOR	0.6682	0.3371	0.6184
S3_pyr	0.7257	0.3628	0.6562
S3_resp	0.7258	0.3578	0.6520

Table 2: Spearman's Correlation on Newsroom.

	COH	INF	FLU	REL
Our approach				
Trained w/ Billsum	**0.6564**	0.7129	0.6025	0.6405
Trained w/ ArXiv	0.6543	**0.7306**	0.5920	**0.6436**
Trained w/ BigPatent	0.6356	0.7205	**0.6075**	0.6408
Reference-free Baselines				
BLANC-tune	0.5862	0.6881	0.5310	0.6078
SummaQA-F1	0.4895	0.5690	0.4664	0.5163
SummaQA-CFD	0.4195	0.5449	0.3719	0.4405
SUPERT	0.6171	0.6929	0.5391	0.6046
LS_Score *	0.6390	0.7163	0.5933	0.6563
Reference-based Upper bounds				
R-1	0.2310	0.3231	0.2150	0.2775
R-2	0.0861	0.1534	0.1015	0.1336
R-L	0.2055	0.3005	0.2006	0.2629
MoverScore	0.1743	0.2186	0.1431	0.2163
BertScore	0.2705	0.3156	0.2390	0.2815
BLEU	-0.0556	-0.0782	-0.0422	-0.0071
METEOR	0.1740	0.2364	0.1690	0.2437
S3_pyr	0.1929	0.2680	0.1782	0.2450
S3_resp	0.1716	0.2519	0.1717	0.2226

* Excluded from comparison because it is trained on Newsroom. Others are not even trained on news domain, except BLANC-tune which is tuned on test data.

method is based on preference ranking, only the Spearman's correlation is reported (Tables 1, 2 and 3). The best scores in the reference-free class are **bold** while top 2 and 3 are underlined.

On TAC2010 (Table 1), our models trained with Billsum and ArXiv are among the top three models. Our model trained with Billsum beats all baselines on all aspects and all metrics on fluency. It further achieves the same level of performance with ROUGE-L on the content aspect.

On Newsroom (Table 2), our models beat all baselines on all aspects. Our models, and all reference-free baselines, outperform reference-based upper bounds. This is contradictory to common cases. It is probably due to that a reference

Table 3: Spearman's Correlation on RealSumm†.

	On abstractive systems	On extractive systems
Our approach		
Trained w/ Billsum	0.2831	0.1077
Trained w/ ArXiv	**0.3088**	**0.1211**
Trained w/ BigPatent	0.2796	0.1033
Reference-free Baselines		
BLANC-tune	0.3067	0.1139
SummaQA-F1	0.2173	0.0837
SummaQA-CFD	0.2433	0.0494
SUPERT	0.2532	0.0748
Reference-based Upper bounds		
R-1	0.6266	0.2182
R-2	0.5623	0.2206
R-L	0.6035	0.2140
MoverScore	0.4951	0.1899
BertScore	0.5682	0.1920
BLEU	0.3023	0.1639
METEOR	0.6270	0.2502
S3_pyr	0.6426	0.2369
S3_resp	0.6264	0.2369

† RealSumm has only one content-focused aspect, no linguistic aspects.

summary mostly has only one sentence in Newsroom.

On RealSumm (Table 3), results are reported separately for abstractive and extractive systems. Our models beat all baselines except BLANC-tune, which is outperformed by our model trained with ArXiv. All approaches perform better for abstractive summarizers than for extractive ones. Bhandari et al. (2020) ascribe this to the low inter agreement among human annotators for the extractive group.

3.6 Discussion: domain impact

Among our models trained with three domains, there is no gold one that is always the best on all test sets and on all aspects. However, on each test set, our worst model is only outperformed by up to one baseline (SUPERT in TAC2010, none in Newsroom, and BLANC-tune in RealSumm) on content/fact-focused aspects – the most important type of aspects in summary evaluation. Because the training domains differ from the test domain, such a performance of our approach suggests its domain robustness. In practice, one can train a model on the test domain or a domain close to the test domain for further performance boost.

4 Conclusion

In this paper, we propose to evaluate single-document summarization quality via preference learning and negative sampling. The experiments show the learned model is transferable across domains and its performance is on the par or better than existing reference-free based methods.

References

Satanjeev Banerjee and Alon Lavie. 2005. 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.

Manik Bhandari, Pranav Narayan Gour, Atabak Ashfaq, Pengfei Liu, and Graham Neubig. 2020. Re-evaluating evaluation in text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Ralph Allan Bradley and Milton E. Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Alexander R Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2020. Summeval: Re-evaluating summarization evaluation. *arXiv preprint arXiv:2007.12626*.

Yang Gao, Christian M Meyer, and Iryna Gurevych. 2020a. Preference-based interactive multi-document summarisation. *Information Retrieval Journal*, 23(6):555–585.

Yang Gao, Wei Zhao, and Steffen Eger. 2020b. SUPERT: Towards new frontiers in unsupervised evaluation metrics for multi-document summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1347–1354, Online. Association for Computational Linguistics.

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*.

Anastassia Kornilova and Vlad Eidelman. 2019. Billsum: A corpus for automatic summarization of us legislation.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*.

NIST. 2010. TAC2010 guided summarization competition. `https://tac.nist.gov/2010/Summarization/Guided-Summ.2010.guidelines.html`. Accessed: 2021-08-16.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Maxime Peyrard, Teresa Botschen, and Iryna Gurevych. 2017. Learning to score system summaries for better content selection evaluation. In *Proceedings of the Workshop on New Frontiers in Summarization*, pages 74–84, Copenhagen, Denmark. Association for Computational Linguistics.

Thomas Scialom, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. 2019. Answers unite! unsupervised metrics for reinforced summarization models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3246–3256, Hong Kong, China. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Ori Shapira, David Gabay, Yang Gao, Hadar Ronen, Ramakanth Pasunuru, Mohit Bansal, Yael Amsterdamer, and Ido Dagan. 2019. Crowdsourcing lightweight pyramids for manual summary evaluation. 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 682–687, Minneapolis, Minnesota. Association for Computational Linguistics.

Eva Sharma, Chen Li, and Lu Wang. 2019. Bigpatent: A large-scale dataset for abstractive and coherent summarization.

Oleg Vasilyev, Vedant Dharnidharka, and John Bohannon. 2020. Fill in the BLANC: Human-free quality estimation of document summaries. In *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems*, pages 11–20, Online. Association for Computational Linguistics.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE* *conference on computer vision and pattern recognition*, pages 4566–4575.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Hanlu Wu, Tengfei Ma, Lingfei Wu, Tariro Manyumwa, and Shouling Ji. 2020. Unsupervised reference-free summary quality evaluation via contrastive learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3612–3621, Online. Association for Computational Linguistics.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. 2019. MoverScore: Text generation evaluating with contextualized embeddings and earth mover distance. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 563–578, Hong Kong, China. Association for Computational Linguistics.

Markus Zopf. 2018. Estimating summary quality with pairwise preferences. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1687–1696, New Orleans, Louisiana. Association for Computational Linguistics.

A Appendix

A.1 Dataset statistics

For test set:

- **TAC2010 Guided Summarization Task Set A** consists of 46 topics, each of which is associated with a set of 10 documents. We evaluate the metrics over summaries generated by 43 systems.

- **Newsroom** contains human-rated summaries generated by 7 systems for 60 documents.

- **RealSumm** sampled 100 documents from the CNN/DailyMail (See et al., 2017) test set, and collected human ratings for summaries generated by 11 extrative systems and 14 abstractive systems.

For training set, the numbers of pairs of documents and reference summaries in the train split are:

- **Billsum**: 18949

- **Scientific papers-ArXiv**: 203037

- **Big-Patent**: 1207222

A.2 Evaluation Settings

We utilize the SummEval (Fabbri et al., 2020) evaluation toolkit to calculate scores for metrics whose scores are not reported by a test dataset. For all metrics, we use the batch evaluation API with default parameters provided by the package. The results of SummEval dataset is not included in this study as SummEval and RealSumm are similar datasets whose documents are both sampled from CNN/DailyMail (See et al., 2017).