BRIO: Bringing Order to Abstractive Summarization
- Yixin Liu1, Pengfei Liu, Dragomir Radev, Graham Neubig (Yale + CMU), ACL 2022

Presentation Author
Mathieu Ravaut

Nanyang Technological University

Table of content

Introduction

Current leading neural abstractive summarization approaches rely on the following :

- Sequence-to-sequence model (e.g, BART [2]).
- Pre-training then fine-tuning.
- Training is done with **MLE** and **teacher forcing**.
- Inference with auto-regressive decoding.

Introduction

There are 2 major issues :

- **MLE only favors the unique ground-truth**, while in practice, there are many suitable summary candidates.
- At inference time, we cannot use teacher forcing, so auto-regressive decoding is employed. This discrepancy between training and inference is known as the **exposure bias**.

Introduction

BRIO modifies the training objective to go from a **one-point** deterministic distribution assumed by MLE to a **non-deterministic distribution** in the which candidates are sorted by their quality.

The model now has two roles : a *generation* role to produce summary candidates, and an *evaluation* one to rank them.

Introduction

BRIO belongs to the category of second-stage summarization approaches. Second-stage summarization can be classified into :

- Training with a **new loss**, in an extra fine-tuning round. For example a **contrastive loss**, like ConSum [6] or SeqCo [8].
- Using **guidance**, including outputs from another summarization model. Example : GSum [1].
- Using a **meta-learning** approach like RefSum [3].
- **Re-rank** candidates, either with a binary classification like SummaReranker [5], or a contrastive loss like SimCLS [4].

BRIO is both in the 1st and 4th categories.

Model : contrastive loss

BRIO proposes to coordinate summaries such that their model log-probability matches their actual quality (as per comparison with the target). A contrastive loss is used :

$$\mathcal{L}_{ctr} = \sum_{i} \sum_{j>i} \max(0, f(S_j) - f(S_i) + \lambda_{ij})$$

where summaries $S_1 \dots S_m$ are sorted in **decreasing ROUGE order** and f is the log-probability assigned by the model :

$$f(S) = \frac{\sum_{t=1}^{l} \log p_{g_\theta}(s_t | D, S_{<t}; \theta)}{|S|^\alpha}$$

Model : overall loss

Multi-tasking is used to train jointly with the cross-entropy loss (generation) and the contrastive loss (evaluation) :

$$\mathcal{L}_{mul} = \mathcal{L}_{xent} + \gamma \mathcal{L}_{ctr}$$

Model : architecture

BRIO is initialized with the **fine-tuned BART or PEGASUS [9]** (depending on the dataset).

The contrastive loss coefficient γ is tuned on the validation set.

Summary candidates are obtained with **diverse beam search** [7].

Experiments : CNN/DM

Results on CNN/DM (BART-based) :

System	R-1	R-2	R-L
	CNNDM		
BART*	44.16	21.28	40.90
PEGASUS*	44.17	21.47	41.11
GSum*	45.94	22.32	42.48
ConSum*	44.53	21.54	41.57
SeqCo*	45.02	21.80	41.75
GOLD-*p**	45.40	22.01	42.25
GOLD-*s**	44.82	22.09	41.81
SimCLS*	46.67	22.15	43.54
BART‡	44.29	21.17	41.09
BRIO-Ctr	47.28†	22.93†	44.15†
BRIO-Mul	**47.78**†	**23.55**†	**44.57**†

New SOTA : +0.62 R-1 compared to SummaReranker.

Experiments : XSum

Results on XSum (PEGASUS-based) :

XSum			
BART*	45.14	22.27	37.25
PEGASUS*	47.21	24.56	39.25
GSum*	45.40	21.89	36.67
ConSum*	47.34	24.67	39.40
SeqCo*	45.65	22.41	37.04
GOLD-*p**	45.75	22.26	37.30
GOLD-*s**	45.85	22.58	37.65
SimCLS*	47.61	24.57	39.44
PEGASUS‡	47.46	24.69	39.53
BRIO-Ctr	48.13†	25.13†	39.84†
BRIO-Mul	**49.07**†	**25.59**†	**40.40**†

New SOTA : +0.95 R-1 compared to SummaReranker.

Experiments : NYT

Results on NYT (BART-based) :

	NYT		
BART‡	55.78	36.61	52.60
BRIO-Ctr	55.98	36.54	52.51
BRIO-Mul	**57.75**†	**38.64**†	**54.54**†

New SOTA : +1.46 R-1 compared to SimCLS.

Experiments : few-shot

Few-shot results : 100 samples on CNN/DM, 1k on XSum

Dataset	System	R-1	R-2	R-L
CNNDM	BART	44.29	21.17	41.09
	BRIO-Few	**45.81**	**21.91**	**42.61**
XSum	PEGASUS	47.46	24.69	39.53
	BRIO-Few	**47.95**	**24.89**	**39.71**

100 samples is enough to significantly improve on SOTA base models BART or PEGASUS.

Experiments : beam width

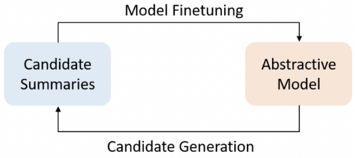
On CNN/DM :

Beams	BART		BRIO-Mul	
	R-1	R-2	R-1	R-2
4	**44.29**	**21.17**	47.78	23.55
10	43.83	20.76	47.98	23.81
20	43.53	20.49	48.07	23.92
50	43.06	20.05	48.18	24.01
100	42.79	19.76	**48.23**	**24.09**

Scaling the beam width leads to even greater gains : +0.45 R-1 when going from 4 to 100 (100 requires huge GPU RAM though).

Experiments : beam width

It's also possible to iterate rounds of fine-tuning with BRIO, then generating new candidates (assumed to be better than before), then fine-tuning with BRIO again, etc.



On CNN/DM, +0.23 R-1 with two rounds :

System	R-1	R-2	R-L
BART	44.29	21.17	41.09
BRIO-Mul	47.78	23.55	44.57
BRIO-Loop	**48.01**†	**23.80**†	**44.67**†

Experiments : other metrics

When ranking candidates with BERTScore [10] :

System	R-1	R-2	R-L	BS
BART	44.29	21.17	41.09	27.38
BRIO-Mul (R)	**47.78**	**23.55**	**44.57**	32.11
BRIO-Mul (B)	47.53	23.22	44.37	**32.59**

Leads to a greater BERTScore result (as expected).

Experiments : abstractiveness

Novel n-grams (on CNN/DM) :

System	Unigram	Bigram
Reference	.1110	.4865
BART	.0101	.0924
BRIO-Mul	.0262	.2381

Much more abstractive than the base BART.

Experiments : abstractiveness

Is the model able to rank candidates in the correct order ?
Correlations between probabilities assigned to candidates and candidate ROUGE :

	Own	**PEGASUS**
BART	.0470	.1205
BRIO-Mul	**.1839**†	**.2768**†

Own means candidates generated by the model (BART), **PEGASUS** are candidates generated by another PEGASUS model.

Conclusion

- New second-stage summarization fine-tuning objective.
- Multi-tasking : traditional cross-entropy + contrastive loss to learn the summary candidate order (coordination).
  - Contrastive loss the same as in SimCLS (shared co-authors).
  - BRIO re-uses the base summarization model (BART or PEGASUS) it's a unified second-stage fine-tuning.
- SOTA ROUGE on 3 datasets.
  - Only news datasets though.
- Several other desirable properties :
  - Good few-shot behavior.
  - High abstractiveness.
  - Optimizing other metrics for re-ranking works too.
  - Scalable with multiple rounds of fine-tuning, greater beam width.

[1] Zi-Yi Dou, Pengfei Liu, Hiroaki Hayashi, Zhengbao Jiang, and Graham Neubig. Gsum : A general framework for guided neural abstractive summarization. *arXiv preprint arXiv :2010.08014*, 2020.

[2] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart : Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv :1910.13461*, 2019.

[3] Yixin Liu, Zi-Yi Dou, and Pengfei Liu. Refsum : Refactoring neural summarization. *arXiv preprint arXiv :2104.07210*, 2021.

[4] Yixin Liu and Pengfei Liu. Simcls : A simple framework for contrastive learning of abstractive summarization. *arXiv preprint arXiv :2106.01890*, 2021.

[5] Mathieu Ravaut, Shafiq Joty, and Nancy F Chen. Summareranker : A multi-task mixture-of-experts re-ranking framework for abstractive summarization. *arXiv preprint arXiv :2203.06569*, 2022.

[6] Shichao Sun and Wenjie Li. Alleviating exposure bias via contrastive learning for abstractive text summarization. *arXiv preprint arXiv :2108.11846*, 2021.

[7] Ashwin K Vijayakumar, Michael Cogswell, Ramprasath R Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. Diverse beam search : Decoding diverse solutions from neural sequence models. *arXiv preprint arXiv :1610.02424*, 2016.

[8] Shusheng Xu, Xingxing Zhang, Yi Wu, and Furu Wei. Sequence level contrastive learning for text summarization. *arXiv preprint arXiv :2109.03481*, 2021.

[9] Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. Pegasus : Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR, 2020.

[10] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore : Evaluating text generation with bert. *arXiv preprint arXiv :1904.09675*, 2019.