FrugalScore: Learning Cheaper, Lighter and Faster Evaluation Metrics for Automatic Text Generation

Moussa Kamal Eddine[1], Guokan Shang[2], Antoine Tixier[1], Michalis Vazirgiannis[1,3]
[1]**DaSciM team**, École Polytechnique; [2]Linagora; [3]AUEB

ACL 2022, Dublin, Ireland, 22th – 27th May 2022
Contact: moussa.kamal-eddine@polytechnique.edu
Paper: https://openreview.net/forum?id=JT-Xedd1zJQ

LaTeXof the slides: https://www.overleaf.com/read/nbfsnrkxmhvc

Agenda

Introduction

Motivation

- Traditional natural language generation metrics (BLEU, ROUGE..) are fast but not reliable.
- New metrics based on large pretrained language models (BERTScore, MoverScore..) are more reliable, but require significant computational resources.

Contributions

- A data-free **Knowledge Distillation** approach for **NLG evaluation metrics**.
- Several orders of magnitude less parameters and run several times faster, while retaining most of the original performance.
- Doesn't rely on any similarity function.
- Regardless of how expensive the original metric is, querying FrugalScore always has the same low, fixed cost.

Our Approach

Three main phases:

- **Phase 1**. Create a synthetic dataset by sampling pairs of more or less related sequences and annotating them with the expensive metrics to be learned.
- **Phase 2**. We continue the pretraining of a miniature pretrained language model on the synthetic dataset built by Phase 1.
- **Phase 3.** *(Optional)*. We fine-tune the miniature on human annotations, which can boost performance.

Once the model is pretrained and optionally finetuned it can be used in inference mode to generate score to an unseen pair of sentences.

Synthetic Datasets

Pretraining Datasets

- **Summarization:** We use 4 different summarization models to generate summaries to all examples in the CNN/DM dataset. The different generated summaries can be associated together to create a pretraining example.
- **Backtranslation:** We translate 1M sentences to French, Arabic and German before translating them back to English.
- **Denoising:** We applied a noise function to a sampled wikipedia segments and we used BART to denoise them.

The sequence pairs are then annotated with the metrics to be learned. **Note that this is a one-time operation that does not need to be repeated regardless of which models are trained downstream.**

Metric Learning

We continue the pretraining of three BERT miniatures on our synthetic dataset: BERT-Tiny ($L = 2$, $H = 128$), BERT-Small ($L = 4$, $H = 512$) and BERT-Medium ($L = 8$, $H = 512$)

Objective

Given two sequences $x = \langle x_1, ..., x_k \rangle$ and $y = \langle y_1, ..., y_l \rangle$, the sequence of contextualised embeddings $\langle \mathbf{z_{[CLS]}}, \mathbf{x_1}, ...\mathbf{x_k}, \mathbf{z_{[SEP]}}, \mathbf{y_1}, ..., \mathbf{y_l} \rangle$ is then obtained.
Add a fully connected layer on top, that linearly projects the $\mathbf{z_{[CLS]}}$ vector to a scalar s.
Minimize the mean square error (MSE) loss between the learned metric s_i and the metric to be learned $\hat{s}_i$:

$$l = \frac{1}{N} \sum_{n=1}^{N} ||s_i - \hat{s}_i||^2 \quad (1)$$

BERTScore

Given two sequences of vector representations produced by a pretrained language model: $\mathbf{x} = \langle \mathbf{x_1}, ..., \mathbf{x_k} \rangle$ and $\mathbf{y} = \langle \mathbf{y_1}, ..., \mathbf{y_l} \rangle$, BERTScore computes:

$$R_{BERT} = \frac{1}{|\mathbf{x}|} \sum_{\mathbf{x_i} \in x} \max_{\mathbf{y}_j \in \mathbf{y}} \mathbf{x_i}^\mathsf{T} \mathbf{y}_j \tag{2}$$

$$P_{BERT} = \frac{1}{|\mathbf{y}|} \sum_{\mathbf{y_i} \in y} \max_{\mathbf{x}_j \in \mathbf{x}} \mathbf{y_i}^\mathsf{T} \mathbf{x}_j \tag{3}$$

$$F_{BERT} = 2 \frac{P_{BERT} R_{BERT}}{P_{BERT} + R_{BERT}} \tag{4}$$

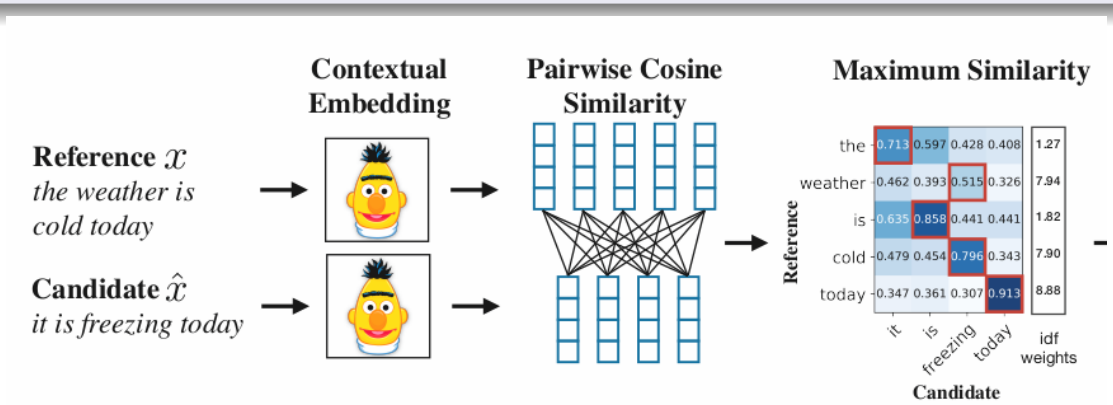

Figure: Taken from (Zhang et al., 2019)

MoverScore

Given two sequences of pre-normalized vector representations produced by a pretrained language model: $\mathbf{x} = \langle \mathbf{x_1}, ..., \mathbf{x_k} \rangle$ and $\mathbf{y} = \langle \mathbf{y_1}, ..., \mathbf{y_l} \rangle$ MoverScore solves for the optimal transportation flow matrix $F \in \mathbb{R}^{|\mathbf{x}| \times |\mathbf{y}|}$ between the two weighted sequences of *n*-grams:

$$WMD(\mathbf{x}, \mathbf{y}) = min_F \langle C, F \rangle \tag{5}$$
$$s.t. \quad F\mathbf{1} = f_{\mathbf{x}}, \quad F^T\mathbf{1} = f_{\mathbf{y}}$$

Where C is the transportation cost matrix (C_{ij} is the Euclidean distance between x_i and y_j) and $f_x \in \mathbb{R}_+^{|\mathbf{x}|}$ and $f_y \in \mathbb{R}_+^{|\mathbf{y}|}$ are the *n*-gram weight vectors.

- We use BERTScore and MoverScore in our experiments as expensive metrics.
- FrugalScore is used in inference mode to generate scores directly after pretraining.
- We evaluate on two text generation tasks: summarization (TAC) and translation (WMT 2019).
- We use evaluation datasets containing (reference, candidate) sequence pairs annotated with human scores.
- We measure the effectiveness of FrugalScore by measuring the Pearson correlation of its scores with the human judgments and comparing it to that of the original metrics.

Results

	Metric	Model	Scores (TAC)	Runtime (TAC)	Scores (WMT)	Runtime (WMT)	Params
a	BERTScore	BERT-Tiny	55.4/47.5	1m 27s	37.6	1m 22s	4.4M
b	BERTScore	BERT-Small	61.6/51.5	2m 20s	39.1	1m 42s	29.1M
c	BERTScore	BERT-Medium	62.7/52.4	2m 28s	39.8	2m 04s	41.7M
d	BERTScore	BERT-Base	64.7/54.7	3m 28s	41.9	2m 09s	110M
e	BERTScore	RoBERTa-Large	64.2/55.4	5m 17s	43.2	3m 03s	355M
f	BERTScore	DeBERTa-XLarge	64.5/**56.0**	6m 20s	**44.5**	3m 49s	900M
g	MoverScore	BERT-Base	**66.5**/55.4	301m 29s	44.0	64m 32s	110M
i	FrugalScore$_d$	BERT-Tiny	64.9/53.5	1m 28s	38.4	1m 18s	4.4M
ii	FrugalScore$_d$	BERT-Small	64.7/53.7	2m 29s	41.3	1m 35s	29.1M
iii	FrugalScore$_d$	BERT-Medium	64.8/54.2	3m 41s	41.9	1m 55s	41.7M
iv	FrugalScore$_e$	BERT-Tiny	60.0/50.1	1m 28s	37.5	1m 18s	4.4M
v	FrugalScore$_e$	BERT-Small	64.1/53.8	2m 29s	40.5	1m 35s	29.1M
vi	FrugalScore$_e$	BERT-Medium	63.9/52.1	3m 41s	41.7	1m 55s	41.7M
vii	FrugalScore$_f$	BERT-Tiny	61.7/51.0	1m 28s	38.0	1m 18s	4.4M
viii	FrugalScore$_f$	BERT-Small	66.0/54.9	2m 29s	41.5	1m 35s	29.1M
ix	FrugalScore$_f$	BERT-Medium	65.5/54.9	3m 41s	43.0	1m 55s	41.7M
x	FrugalScore$_g$	BERT-Tiny	**67.3**/**55.1**	1m 28s	39.8	1m 18s	4.4M
xi	FrugalScore$_g$	BERT-Small	65.9/54.7	2m 29s	42.8	1m 35s	29.1M
xii	FrugalScore$_g$	BERT-Medium	66.2/**55.1**	3m 41s	**43.6**	1m 55s	41.7M

Table: Scores are summary-level (TAC) and segment-level (WMT) Pearson correlations averaged over 2008 to 2011 for TAC (pyramid score/responsiveness) and over all source languages for WMT-2019. Runtimes include preprocessing. Subscripts refer to row labels and indicate which metric-model combination was used to annotate pairs (e.g., for FrugalScore$_d$, it is row d, i.e., BERTScore-BERT-Base).

Finetuning (1/2)

- We apply the third optional phase: fine-tuning FrugalScore on a dataset annotated with human scores.
- Because we cannot use the same human-annotated dataset for finetuning and evaluation, we finetune a BERT-Small on each year of TAC for four epochs, and we use another year as the test set. We use the remaining two years as the validation set.
- We finetune the model in two settings: (1) directly on the training set (2) after the continuation of its pretraining on our synthetic dataset.

Finetuning (2/2)

	Further Pretraining	TAC-2008	TAC-2009	TAC-2010	TAC-2011
TAC-2008	no	-	$67.7_{0.57}$	$66.1_{0.18}$	$63.6_{0.36}$
	yes		$74.4_{0.13}$	$71.3_{0.04}$	$67.3_{0.13}$
TAC-2009	no	$61.4_{0.41}$	-	$66.9_{0.24}$	$62.7_{0.55}$
	yes	$65.8_{0.25}$		$70.7_{0.32}$	$66.0_{0.18}$
TAC-2010	no	$59.7_{0.47}$	$67.3_{0.7}$	-	$62.4_{0.47}$
	yes	$64.7_{0.19}$	$74.3_{0.24}$		$67.2_{0.11}$
TAC-2011	no	$57.6_{1.39}$	$64.7_{1.03}$	$66.5_{0.66}$	-
	yes	$63.9_{0.31}$	$72.0_{0.44}$	$71.6_{0.44}$	

Table: Rows correspond to the training set and columns to the test set.

Everywhere, the pretraining step leads to a significant boost in the performance of the model in the finetuning setting.

Conclusion

- An approach to learn a fixed, low-cost version of any expensive NLG evaluation metric.
- Experiments on summarization and translation tasks show that our FrugalScore versions of BERTScore and MoverScore retain most of the original performance while running several times faster and having several orders of magnitude less parameters.
- On average FrugalScore retains 96.8% of the performance, runs 24 times faster, and has 35 times less parameters.
- Our approach leads to a significant improvement in the supervised setting.