Prophecy Distillation for Boosting Abstractive Summarization

Jiaxin Duan[1†], **Fengyu Lu**[1†] , **Junfei Liu**[2*]
School of Software and Microelectronics, Peking University
Beijing, China
[1]{duanjx, fengyul}@stu.pku.edu.cn, [2]liujunfei@pku.edu.cn

Abstract

Abstractive summarization models learned with maximum likelihood estimation (MLE) have long been guilty of generating unfaithful facts alongside ambiguous focus. Improved paradigm under the guidance of reference-identified words, i.e., guided summarization, has exhibited remarkable advantages in overcoming this problem. However, it suffers limited real applications since the prophetic guidance is practically agnostic at inference. In this paper, we introduce a novel teacher-student framework, which learns a regular summarization model to mimic the behavior of being guided by prophecy for boosting abstractive summaries. Specifically, by training in probability spaces to follow and distinguish a guided teacher model, a student model learns the key to generating teacher-like quality summaries without any guidance. We refer to this process as prophecy distillation, and it breaks the limitations of both standard and guided summarization. Through extensive experiments, we show that our method achieves new or matched state-of-the-art on four well-known datasets, including ROUGE scores, faithfulness, and saliency awareness. Human evaluations are also carried out to evidence these merits. Furthermore, we conduct empirical studies to analyze how the hyperparameters setting and the guidance choice affect TPG performance.

Keywords: Abstractive summarization, Guided summarization, Knowledge distillation

1. Introduction

Abstractive summarization is a fundamental task of natural language processing (NLP) that aims to rewrite a given document into a short summary, only preserving its essential information and meaning (Peyrard, 2019; Liu and Lapata, 2019; Li et al., 2018b). Following universal text-to-text generation, advanced summarization systems typically train an encoder-decoder model with maximum likelihood estimation (MLE) to predict gold reference. Although easy to follow, this generic technology gives up any task-specific bias during learning, causing the model-generated summaries to deviate from human focus, and on the other hand, tend to contain notorious hallucinations (Tang et al., 2022; Wang et al., 2022).

To address these problems, many studies argued that summaries should be logically entailed in the source document to reach faithful summarization. They either introduce auxiliary tasks defined on external datasets, such as entailment recognition (Li et al., 2018a) and entailment generation (Pasunuru et al., 2017), or directly learn a model to maximize logical entailment rewards (Pasunuru and Bansal, 2018; Roit et al., 2023). Yet, it hurts model performance when measured by the commonly used metrics (Dreyer et al., 2023), i.e., ROUGE (Lin, 2004). Guided summarization (Dou et al., 2021) (a variant paradigm of controllable summarization (He et al., 2022; Fan et al., 2018) see Section 2) instead constrains the summary to stick with the human interest by feeding a model reference-identified guidance words (i.e., prophecy). Compared with the entailment-aware approaches, guided summarization relies on no extra data or annotations while showing empirical effectiveness in generating faithful and human-like summaries. However, its practical advantages are heavily limited due to discrepancies between training and inference.

On the one hand, existing guided summarization methods learn with the standard MLE, which requires a teacher-forcing algorithm (Goyal et al., 2016) to ensure stability. Consequently, the model generates subsequent texts based on the accurate pre-texts during training while based on their preceding outputs at inference, resulting in *exposure bias* (Bengio et al., 2015; Goodman et al., 2020). Furthermore, the prophetic guidance (such as keywords or highlighted sentences) is identified based on the gold reference during training and is therefore agnostic in the inference stage. State-of-the-art (SOTA) methods additionally train a BERT-based model (Liu et al., 2019) to predict this absence, yet leading to significant performance deterioration (Dou et al., 2021; He et al., 2022).

The above limitations motivate us to develop a model that mimics the behavior of being guided by prophecy (actually not) to boost abstractive summaries, where holistic learning technologies (Liu et al., 2022; Xie et al., 2023) are also used to avoid exposure bias. In this paper, we embed the idea of prophecy-guiding into a novel teacher-student learning framework - TPG (**t**ransferring **p**rophetic **g**uidance to abstractive summarization). Taking salient textual spans as guidance, TPG first learns

† Equal contributions in this work.
* Corresponding author.

LREC-COLING 2024, pages 13257–13268

Document: Billy **Jones** has joined Sunderland on a free transfer after rejecting a new contract from West Bromwich Albion. The right-back, 27, was offered a **three-year deal at the Hawthorns** but has opted for the extra year tabled by the north-east club. Jones has been one of West Brom's most consistent performers since **signing** in 2011 even though he only played 22 games last season due to injury. Deal: Billy Jones has joined Sunderland on a free transfer after rejecting a new contract from West BromLooking out for his future: The **defender was offered** a four-year deal at the Stadium of Light. He said: 'I'm really happy to be here and I'm looking forward to getting back for pre-season and kicking on.' West Brom had been in negotiations with Jones since October and offered 'vastly-improved terms' but will now turn their attentions to other right-backs. Director of Football Administration, Richard Garlick, said: 'We're obviously disappointed by Billy's decision but wish him well in his future career. 'With there being no guarantees that Billy would re-sign, we have been preparing for this scenario and will pursue the options we have been exploring in the right-back position.
Guidance: **Jones** [SEP] **three-year deal at the Hawthorns** [SEP] **signing** [SEP] **defender was offered**
Input: **Jones** [SEP] **three-year deal at the Hawthorns** [SEP] **signing** [SEP] **defender was offered** [SEP] (Document)
Summary: **Defender was offered three-year deal at the Hawthorns**. **Jones** becomes Gus Poyet's first **signing** of the summer.

Table 1: A case of guided summarization. (Document) refers to the content of the document text. The document-summary pair is sampled from the CNN/DM training set. Salient spans are highlighted with yellow or **bold**.

a teacher model following SOTA controllable summarization (He et al., 2022). A student model in the regular setting is then learned to align with the teacher regarding token distributions and summary-level probability masses. To this end, we introduce contrastive summary-level knowledge distillation (KD) learning apart from the traditional token-level one (Hinton et al., 2015), where the teacher's output and gold reference individually serve as the soft and hard criteria to build contrastive objectives. In this way, the student learns from dual levels to distinguish and trace the effects of prophecy, thereby adaptively perceiving the keys for generating teacher-like summaries without requiring guidance. Simultaneously, the defective MLE with teacher-forcing is bypassed.

It is worth noting that our method is very distinct from guided summarization. Consider the latter refers to a specific summarization paradigm that features the additional input of prophetic clues. By contrast, TPG is a learning framework that enforces a regular summarization model without guidance to simulate the guided behavior. Furthermore, ideal guided summarization models are unreachable in practice, but the model learned with TPG is the opposite. We summarize the main contributions of this work as follows:

- As far as we know, we are the first to learn a model to capture the knowledge in prophetic guidance, which we call *prophecy distillation*.
- We perform prophecy distillation with a novel teacher-student learning framework to boost abstractive summarization.
- Extensive experiments are conducted on four well-known benchmarks to test our learned models, including TPG teacher and student. Results show that the TPG teacher outperforms previous guided models, and it contributes to the student's superiority in generating high-quality summaries, which is reflected as the new or matched SOTA on multiple representative metrics among lexical overlap, semantic similarity, faithfulness, etc. We further carry out human evaluations and get evidence supporting these observations[1].

2. Preliminary

Controllable Summarization. Abstractive summarization commonly treats it as a sequence-to-sequence task to model the probability of generating the reference summary $\mathbf{Y} = (y_1, y_2, \cdots, y_{|\mathbf{Y}|})$ given a source document $\mathbf{X} = (x_1, x_2, \cdots, x_{|\mathbf{X}|})$, i.e., $P(\mathbf{Y}|\mathbf{X})$, where x_t (y_t) is a token. Different from that, controllable summarization models the conditional probability $P(\mathbf{Y}|\mathbf{X}, \mathbf{G})$, where $\mathbf{G}$ is guidance typically instantiated as artificial codes used to control the summary attributes, including length (Liu et al., 2018; Kikuchi et al., 2016), entities (Zhang et al., 2022a), and styles (Fan et al., 2018).

Guided Summarization. The latest studies (Dou et al., 2021; He et al., 2022) are interested in controlling a model with reference-related keywords or highlighted sentences to close the gap between the generated summaries and human-written references. Since the content of signal $\mathbf{G}$ converts from control codes to prophetic phrases, Dou et al. (2021) named this line of methods guided summarization to distinguish with the original. However, although it shows unprecedented advantages in artificial experiments, the guidance must be identified through an oracle, which is agnostic during inference, causing guided summarization sound in theory while weak in practice.

In this paper, we make a compromise between the semantic-less keywords and tedious highlighted

[1]Our codes will be soon available at https://github.com/jaxdan23/TPG

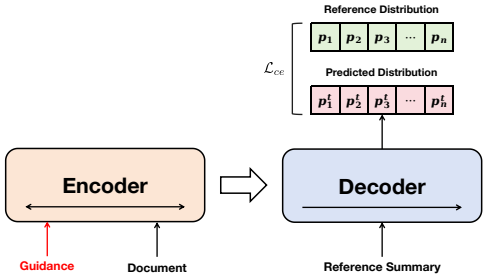


Figure 1: The illustration of teacher learning.

sentences and set the guidance as salient spans of a source document that can be retrieved in the reference summary. To pick out such spans, we split the source document into short sentences according to punctuation, and then identify the *common continuous sub-sequences (CCS)* between each sentence and the reference summary using a dynamic programming algorithm (DP). After that, we remove the stop words and greedily select a set of informative sub-sequences that achieves the maximum ROUGE score with the reference. We refer to each retained sub-sequence as a textual span, and this overall procedure is a so-called **oracle**, detailed in Algorithm 1.

To incorporate the guidance into a system without introducing additional modules, we connect the extracted spans with a [SEP] token and follow (He et al., 2022), prepend it to the source document. A case of our implementation is demonstrated in Table 1 for clear understanding.

Algorithm 1 DP Algorithm for Getting CCS

Input: Strings $\mathbf{x} = \{x_1, x_2, \cdots, x_m\}$ and $\mathbf{y} = \{y_1, y_2, \cdots, y_n\}$
Output: Common continuous sub-sequences set **s** of **x** and **y**

$\mathbf{s} = \{\}$, $\mathrm{M} = \mathbf{O}_{(m+1)\times(n+1)}$
for $i \leftarrow 1, m+1$ **do**
  for $j \leftarrow 1, n+1$ **do**
    if $x_{i-1} == y_{j-1}$ **then**
      $\mathrm{M}_{ij} = \mathrm{M}_{i-1,j-1} + 1$
    else
      $\mathrm{M}_{ij} = 0$
    end if
  end for
end for
$i = m$
while $i > 0$ **do**
  $k = max(\mathrm{M}_{i1}, \mathrm{M}_{i2}, \cdots, \mathrm{M}_{in})$
  if $x_{i-k:i} \notin \mathbf{s}$ **then**
    $\mathbf{s} = \mathbf{s} \cup \{x_{i-k:i}\}$
  end if
  $i = i - 1$
end while
return s

3. Method

Our TPG is a teacher-student learning framework, where the teacher model acts as an intermedia to learn a robust student. We start the teacher and student with an identical pre-trained language model, namely *baseline* so that no architectural differences lie between them. However, the two models have distinct input streams and learning objectives, which we will describe in this section.

3.1. Teacher Model

The teacher in TPG is a guided summarization model parameterized by θ. As Figure 1 shows, we train it with maximum likelihood estimation (MLE)-based teacher forcing, which aims to minimize the following negative log-likelihood (NLL) loss:

$$\mathcal{L}_{nll}(\theta) = -\sum_{t}^{|\mathbf{Y}|} \log \mathrm{P}_\theta \left(y_t \mid y_{<t}, \mathbf{X}, \mathbf{G}\right). \quad (1)$$

Many studies (Liu et al., 2022; Zhao et al., 2022) pointed out that such a trained model is biased because it ideally assigns the probability 1 to the gold reference while the probability 0 to any other possible candidate, causing a lack of relative information among candidates. When it comes to inference, the performance degrades a lot. To address this problem, we follow previous works and adopt label smoothing (Müller et al., 2019), which loosens the NLL loss function to the following cross-entropy (CE) loss:

$$\mathcal{L}_{ce}(\theta) = -\sum_{t=1}^{|\mathbf{Y}|}\sum_{y'} \tilde{\mathrm{P}}(y') \log \mathrm{P}_\theta \left(y' \mid y_{<t}, \mathbf{X}, \mathbf{G}\right) \quad (2)$$
$$\tilde{\mathrm{P}}\left(y'\right) = \begin{cases} 1-\gamma, y' = y_t \\ (1-\gamma)/N, y' \neq y_t \end{cases}$$

where $\tilde{\mathrm{P}}(y')$ is a soft label distribution, $\gamma \in (0, 1)$ is a coefficient to control the gap between the gold token and other tokens, and N is the size of the vocabulary.

Artificial experiments (see Section 4) show that the learned TPG teacher has absolute superiority in improving summarization performance. However, the ideal guidance introduced in Section 2 is practically unreachable at inference. We thus learn further a student model to break this limitation.

3.2. Student Model

The TPG student, with parameters ϕ, is a standard abstractive summarization model typically learned towards the MLE objective mentioned in Section 3.1. Let y'_t be the token generated at step t, it predicts the token distribution $\mathrm{P}_\phi\left(\cdot \mid y_{<t}, \mathbf{X}\right)$ during training while $\mathrm{P}_\phi\left(\cdot \mid y'_{<t}, \mathbf{X}\right)$ during inference, thus causing the train-inference discrepancy - exposure bias. Recent methods (Liu et al., 2022; Zhao et al., 2022; Zhang et al., 2022b; Xie et al., 2023) add holistic objectives in addition to the token-level MLE, which effectively addresses this problem. It inspires us to involve a similar technology in learning the student model. Concretely, we introduce

dual-level knowledge distillation (KD) based training as described below. We also call this method "prophecy distillation" in the following text, which means distilling prophetic guidance from a guided model to distinguish it from the traditional KD approaches that aim to distill knowledge from a large neural model.

3.2.1. Token-level Distillation

The only purpose of token-level prophecy distillation is to align the student and teacher behaviors in predicting a token. For this reason, this process follows the traditional knowledge distillation (Hinton et al., 2015; Kim and Rush, 2016) setting. As described in Figure 2, the smoothed teacher-output and gold reference are individual as the soft and hard label distributions in supervised learning, and a balance factor $\xi \in (0, 1)$ is introduced to mix the two parts of corresponding losses:

$$\mathcal{L}_t(\phi) = \xi \mathcal{L}^{t(\phi)}_{hard} + (1 - \xi)\mathcal{L}^{t(\phi)}_{soft}, \tag{3}$$

where the loss over the hard label follows NLL:

$$\mathcal{L}^{t(\phi)}_{hard} = -\sum_{t=1}^{|\mathbf{y}|} \log \mathrm{P}_\phi \left(y_t \mid y_{<t}, \mathbf{X}\right), \tag{4}$$

and over the soft label follows CE:

$$\begin{aligned} \mathcal{L}^{t(\phi)}_{soft} &= -\sum_{t=1}^{|\mathbf{y}|} \sum_{y'} \mathrm{p}^t_\theta \left(y'\right) \log \mathrm{p}^t_\phi \left(y'\right) \\ \mathrm{p}^t_\theta \left(y'\right) &= \frac{e^{g_\theta(y'|y_{<t}, \mathbf{X}, \mathbf{G})/T}}{\sum_{y_k} e^{g_\theta(y_k|y_{<t}, \mathbf{X}, \mathbf{G})/T}} \\ \mathrm{p}^t_\phi \left(y'\right) &= \frac{e^{g_\phi(y'|y_{<t}, \mathbf{X})/T}}{\sum_{y_k} e^{g_\phi(y_k|y_{<t}, \mathbf{X})/T}} \end{aligned}, \tag{5}$$

where y_k represents any token in the vocabulary, $g(\cdot)$ denotes a non-normalized likelihood distribution, i.e., logit, and $T > 1$ is the so-called temperature coefficient. By training with a relatively small ξ, the token distribution produced by the student will be close to that of the teacher, which contributes to generating quality summaries while bypassing the prophetic guidance.

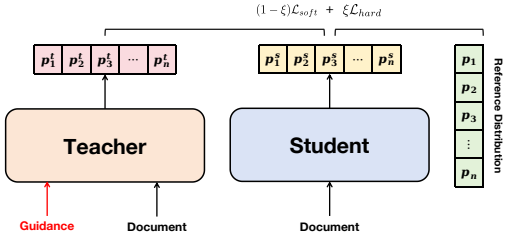
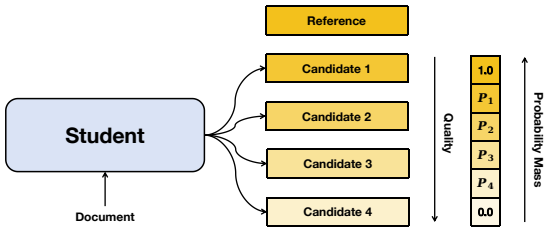
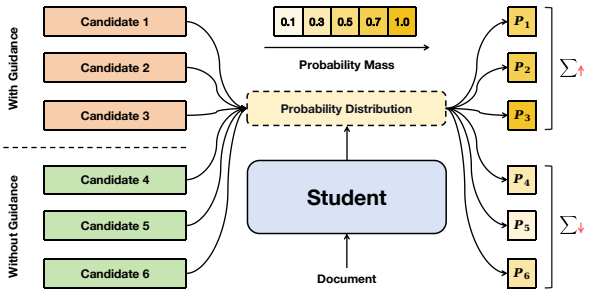


Figure 2: Token-level knowledge distillation.



(a) The hard objective of summary-level PD.



(b) The soft criterion of Summary-level PD.

Figure 3: Summary-level prophecy distillation.

3.2.2. Summary-level Distillation

We conduct summary-level distillation learning for dual purposes, as the solely token-level one is not guaranteed for the student to generate summaries holistically resemble the teacher, and it also contributes little to overcoming exposure bias.

As demonstrated in Figure 3a, with the reference as a hard criterion, we follow (Liu et al., 2022) and request the student to assign probability mass to candidate summaries according to their quality. Given a document $\mathbf{X}$ in training set, we first sample n candidate summaries $\mathbf{Y}_1, \mathbf{Y}_2, \cdots, \mathbf{Y}_n$ from the student outputs using diverse beam search (Vijayakumar et al., 2016). Then, we arrange these candidates in descending order based on their ROUGE score w.r.t the reference $\mathbf{Y}$ and define the following list-wise ranking loss:

$$\mathcal{L}^{h(\phi)}_{hard} = \sum_i \sum_{j>i} \max\left(0, f\left(\mathbf{Y}_j\right) - f\left(\mathbf{Y}_i\right) + \lambda_{ij}\right) \tag{6}$$

where $\lambda_{ij} = (j - i) \cdot \lambda$, λ is a hyperparameter, and $f(\cdot)$ refers to the length-regularized log-probability:

$$f(\mathbf{Y}) = \frac{\sum_{t=1}^{|\mathbf{Y}|} \log \mathrm{P}_\phi \left(y_t \mid y_{<t}, \mathbf{X}\right)}{|\mathbf{Y}|^\alpha}, \tag{7}$$

where α is a hyperparameter for length penalty. Now that $\mathcal{L}^h_{hard}$ has been proven effective in alleviating exposure bias, we additionally expect the student model to assign larger probability masses to the teacher-generated candidates (viewed as a soft criterion) than to other ones being generated without the guidance of prophecy. Inspired by (Xie et al., 2023), we sample n positive candidates $\mathbf{Y}^+_1, \mathbf{Y}^+_2, \cdots, \mathbf{Y}^+_n$ from the teacher outputs and

an equal number of negative ones $\mathbf{Y}_1^-, \mathbf{Y}_2^-, \cdots, \mathbf{Y}_n^-$ from a trained regular summarizer (i.e., the baseline), and we introduce the following contrastive loss function:

$$\mathsf{P}_{pos} = \frac{\sum_{i=1}^{n} w_i f(\mathbf{Y}_i^+)}{\sum_{i=1}^{n} w_i}, \mathsf{P}_{neg} = \sum_{i=1}^{n} \frac{f(\mathbf{Y}_i^-)}{n}, \quad (8)$$
$$\mathcal{L}_{soft}^{h(\phi)} = \log\left(1 + e^{\mathsf{P}_{neg} - \mu \mathsf{P}_{pos}}\right)$$

where w_i is a weight whose value is positively related to the quality of $\mathbf{Y}_i^+$. Figure 3b provides a graphical illustration of this objective. Since the baseline only differs with the teacher for lack of guidance, we train the student with $\mathcal{L}_{soft}^{h}$ to distinguish the guidance effects. Further, once the student assigns relatively high probability masses to the teacher-generated candidates, it tends to generate teacher-like summaries, which voids the need for additional guidance.

Finally, we follow the format of conventional knowledge distillation, combining the two types of contrastive losses with another balance factor ζ, i.e., $\mathcal{L}_h(\phi) = \zeta \mathcal{L}_{hard}^{h(\phi)} + (1-\zeta)\mathcal{L}_{soft}^{h(\phi)}$. The overall training loss for the student is then:

$$\mathcal{L}(\phi) = \mathcal{L}_t(\phi) + \beta \mathcal{L}_h(\phi). \quad (9)$$

Dataset	# Samples			# Avg. Words	
	Train	Valid	Test	Doc.	Sum.
CNN/DM	287,227	13,368	11,490	791.6	55.6
XSum	204,045	11,332	11,334	429.2	23.3
NYT	589,284	32,736	32,739	800.0	35.6
SAMSum	14,732	818	819	97.2	21.0

Table 2: Datasets Statistics. # counts the number of samples or words in a dataset. Avg.: average. Doc.: document. Sum.: summary.

Dataset	γ	β	ξ	T	λ	α	ζ	μ
Others†	0.2	10	0.5	2.0	0.001	2.0	0.3	2.0
XSum	0.2	10	0.5	2.0	0.1	0.6	0.9	2.0

Table 3: Hyperparameter settings. Others† mean the other three datasets except XSum.

4. Experiments

4.1. Datasets

CNN/DM (Hermann et al., 2015; Nallapati et al., 2016) is the most widely used dataset, which takes the summarization as news articles while summaries as associated highlights. The dataset includes anonymized and non-anonymized versions. We leverage the latter to be consistent with previous works.

XSum (Narayan et al., 2018) is an extremely abstractive summarization dataset with a one-sentence summary written by human experts for each news article collected from BBC online.

NYT (Mozzherina, 2013) consists of articles from the New York Times and the associated summaries. We use the splits and pre-processing steps of Paulus et al. (2018) in experiments.

SAMSum (Gliwa et al., 2019) is an abstractive dialogue summarization dataset annotated by human linguists.

Refer to Table 2 for detailed statistics of the above datasets.

4.2. Baselines

We compare our results with six existing SOTA models that fall into three categories: (1) Baseline models: **BART** (Lewis et al., 2020), a classical base model in summarization, which pre-trained to recover the corrupted text and achieved considerable performance on CNN/DM. **PEGASUS** (Zhang et al., 2020a), another typically used Seq2Seq model trained with Gap Sentences Generation (GSG) and Masked Language Modeling (MLM), and achieved early SOTA on XSum; (2) Controllable summarization models: **GSum** (Dou et al., 2021), a Seq2Seq model with two decoders for document and guidance respectively. **CTRLsum** (He et al., 2022) proposed prepending the oracle extracted keywords to document and achieved the comparable results with GSum; (3) Contrastive models: **SimCLS** (Liu and Liu, 2021), a two-stage model which trained an evaluator using contrastive ranking loss to select the best result from the generator's outputs. **BRIO** (Liu et al., 2022) was first proposed to assign probability mass to candidate summaries according to their quality, achieving the current SOTA on both CNN/DM and XSum. We also consider two recent variants of BRIO: **SLiC** (Zhao et al., 2022) adapts different types of contrastive losses, and **MoCa** (Zhang et al., 2022b) introduces online sampling.

4.3. Implementation Details

To facilitate comparison, we implement both TPG teacher and student models with an identical backbone, i.e., PEGASUS[2] on XSum and BART-large[3] on other datasets. We train our models using the Adam optimizer (Kingma and Ba, 2015) with an initial learning rate (lr) of 2e-3 and batch size of 16 for at least 100K steps until the performance on develop sets no longer be improved. We also schedule the learning rate according to $lr^* = lr \cdot \min(\text{step}^{-0.5}, \text{step} \times \mathcal{W}^{-1.5})$, where $\mathcal{W}$ indicates

[2]https://huggingface.co/google/pegasus-xsum
[3]https://huggingface.co/facebook/bart-large

Model	CNN/DM R-1	CNN/DM R-2	CNN/DM R-L	XSum R-1	XSum R-2	XSum R-L	SAMSum R-1	SAMSum R-2	SAMSum R-L	NYT R-1	NYT R-2	NYT R-L
BART†	44.16	21.28	40.90	45.14	22.27	37.25	53.42‡	28.14‡	49.03‡	55.78‡	36.61‡	52.60‡
PEGASUS†	44.17	21.47	41.11	47.21	24.56	39.25	-	-	-	-	-	-
GSum*†	45.94	22.32	42.48	45.40	21.89	36.67	-	-	-	-	-	-
CTRLsum*†	45.65	22.35	42.50	-	-	-	-	-	-	-	-	-
SimCLS†	46.67	22.15	43.54	47.61	24.57	39.44	-	-	-	-	-	-
BRIO†	48.01	23.80	44.67	49.07	25.59	40.40	53.74‡	29.06‡	49.37‡	57.75	38.64	54.54
SLiC†	47.97	24.18	44.88	49.77	**27.09**	**42.08**	-	-	-	-	-	-
MoCa†	48.88	24.94	45.76	49.32	25.91	41.47	**55.13**	30.57	50.88	-	-	-
TPG-S	**49.38**	**25.17**	**45.79**	**49.98**	25.77	41.52	54.89	**30.75**	**51.90**	**58.20**	**39.49**	**55.36**
- *w/o $\mathcal{T}$-KD*	48.16	24.08	44.65	48.81	25.04	40.85	54.59	29.67	50.87	58.13	39.43	54.24
- *w/o $\mathcal{S}$-KD*	45.77	22.30	42.38	47.52	24.71	39.69	54.09	28.40	50.45	57.65	38.30	53.67

Table 4: Evaluation results. †: results reported in the original papers. ‡: results from our own implementation. $*$: results from the guided model with the BERT-predicted guidance. The best results are in **bold**, and the baseline on each dataset is marked with underline. *w/o* means without. *$\mathcal{T}$-PD*: token-level prophecy distillation. *$\mathcal{S}$-PD*: summary-level prophecy distillation. R-1/2/L are the ROUGE-1/2/L F1 scores.

Model	CNN/DM BS	CNN/DM BaS	CNN/DM SwR	XSum BS	XSum BaS	XSum SwR
BART	85.95	-3.80	68.75	89.63	-3.77	15.23
PEGASUS	85.07	-3.91	65.00	89.68	-3.89	19.37
GSum	86.10	-3.87	68.09	88.61	-3.72	15.75
BRIO	89.14	-3.62	62.28	90.23	-3.64	19.97
TPG-S	**89.32**	**-3.25**	70.51	**92.13**	**-3.61**	20.30
- *w/o $\mathcal{T}$-PD*	89.14	-3.81	67.02	91.58	-3.88	**20.85**
- *w/o $\mathcal{S}$-PD*	89.11	-3.74	69.74	91.98	-3.69	19.06

Table 5: More evaluation results. All results are from our own implementation. BS: BERTScore. BaS: BARTScore-$\mathcal{F}$. SWR: Salient words recall.

the warmup steps and is set to 10K. To prevent premature convergence, we warm the student model during training by setting the weight of $\mathcal{L}_h(\phi)$, i.e., β to 0 in the first 20K steps. When decoding, beam search (Vijayakumar et al., 2016) is used, and the beam width is set to 20. All our experiments are implemented based on 8 NVIDIA RTX 3090 GPUs and the Pytorch[4] library. Table 3 lists detailed hyperparameter settings.

4.4. Main Results

By convention, we measure the model-generated summary and the reference regarding 1) lexical overlap based on ROUGE (Lin, 2004), 2) semantic similarity based on BERTScore (Zhang et al., 2020b) and BARTScore-$\mathcal{F}$ (Lewis et al., 2020), and 3) saliency based on salient words recall:

$$\text{SwR}(\mathbf{Y}', \mathbf{Y}) = \frac{|LCS(\mathbf{Y}', LCS(\mathbf{X}, \mathbf{Y}))|}{|\mathbf{Y}'|}\%, \quad (10)$$

where $LCS(\cdot,\cdot)$ denotes the longest common subsequence (LCS). We also refer to the TPG student model as TPG-S and the teacher model as TPG-T for convenience. The comparison results are reported in Table 4 and Table 5, from which we draw the following observations.

[4]https://pytorch.org

First, *TPG-S is more advanced than traditional guided summarization methods GSum and CTRLsum in real-life scenarios*. Without oracle-extracted guidance, the guided models surpass the baselines only by a limited margin, and GSum even fails against the baseline PEGASUS on XSum. By contrast, our TPG-S outperforms the baseline by more than 5 points ROUGE-1 on CNN/DM and 2 points ROUGE-2 on other datasets. Besides, BRIO, SLiC, and MoCa share a similar learning scheme of token-level MLE plus summary-level contrasting, leading to their close performance. Compared with them, TPG substitutes MLE with token-level KD and further introduces a soft contrastive objective on the summary level. *TPG-S achieves SOTA ROUGE scores across all four datasets*, proving the effectiveness of our strategy. We further find from Table 5 that *TPG-S does the best in tracing salient words*, which is reflected as the absolute advanced SwR score on CNN/DM. Results in Table 8 (discussed in Section 4.7) indicate that this merit is mainly derived from the teacher model and leads to the compatibility of TPG-S for moderate abstractive summarization.

4.5. Faithfulness Evaluation

Beyond conventional lexical-level evaluations, we highlight to assess how the model-generated summaries are faithful to the source document. **FactCC** (Kryscinski et al., 2020), **MNLI** (Choubey et al., 2023), and **QAFactEval** (Fabbri et al., 2022) are used as the metrics. We compared our method with three types of counterparts: **FASum** (Zhu et al., 2021) encodes an additional knowledge graph to promote the factual generation; **RLEF** (Roit et al., 2023) uses the feedback of an entailment analyzer to reward the summaries that are logically entailed in the source document; Contrastive learning methods **CLIFF** (Cao and Wang, 2021) and **CaPE** (Choubey et al., 2023) construct negative samples

Model	Auxiliary Systems	CNN/DM R-1	CNN/DM FC	CNN/DM MNLI	CNN/DM QAEval	XSum R-1	XSum FC	XSum MNLI	XSum QAEval
*BASE**	-	44.34	49.07‡	84.20‡	4.55‡	47.21	23.47‡	22.70‡	2.10‡
FASum	Knowledge Graph♭	40.53	51.24‡	81.33‡	4.48‡	30.28	**26.10**‡	22.40‡	1.88‡
CLIFF	Hybrid	44.18	51.84	81.09‡	3.63‡	46.20	24.26	23.10‡	1.68‡
CaPE	Hybrid	45.14	-	86.80	4.60	43.71	-	23.10	2.21
RLEF	Entailment Model	31.28	58.73‡	79.67‡	4.73‡	38.13	22.21‡	21.18‡	2.38‡
TPG-T	Oracle♭	**58.61**	**61.67**	**87.36**	**4.81**	47.12	22.26	**24.40**	1.61
TPG-S	TPG-T	49.38	60.98	86.46	4.71	**49.98**	23.88	24.32	**2.50**

Table 6: More automatic evaluation results. *BASE**: BART on CNN/DM while PEGASUS on XSum. Hybrid: the combination of diverse systems. FC: FactCC. QAEval: QAFactEval. ♭: the auxiliary systems used in both training and inference; otherwise, only in training. ‡: results from our own implementation.

Model	Faithfulness Win↑	Faithfulness Tie	Faithfulness Lose↓	Informativeness Win↑	Informativeness Tie	Informativeness Lose↓
CNN/DM						
RLEF	11.10	85.84	3.06	**20.63**	72.50	6.87
CLIFF	15.10	81.77	3.12	20.38	75.00	4.61
TPG-S	**19.69**	77.52	**2.78**	17.44	78.81	**3.75**
XSum						
RLEF	**16.38**	77.92	**5.69**	8.37	87.67	**3.96**
CLIFF	16.35	73.45	9.20	**9.67**	82.28	8.05
TPG-S	14.65	78.43	6.92	5.47	87.54	6.99

Table 7: Human evaluation results.

Model	Guidance	R-1	R-2	R-L	SwR
Automatic					
GSum	Sentences	**45.94**	22.32	42.48	68.09‡
CTRLsum	Keywords	45.65	**22.35**	**42.50**	**68.99**‡
TPG-T	Spans	45.85	22.04	42.38	68.48
	- *w/o* [SEP]	45.69	22.04	42.23	67.50
Oracle					
GSum	Sentences	55.18	32.54	52.06	74.41‡
CTRLsum	Keywords	64.65	40.42	60.92	-
CTRLsum‡	Keywords	56.39	35.12	56.77	74.58
TPG-T	Spans	**58.61**	**36.45**	**58.55**	**75.09**
	- *w/o* [SEP]	58.20	36.05	58.37	74.39

Table 8: Performance of guided summarization models on CNN/DM. ‡: results from our own implementation.

using diverse systems, including entity recognizer, entailment analyzer, and entailment analyzer, and train a summarization model to distinguish faithful and unfaithful summaries. Table 6 shows the comparison results.

Surprisingly, not all improved methods can effectively promote baseline faithfulness scores. Only CaPE improves nearly all observed metrics on both datasets. On the other hand, traditional faithful-aware methods sacrifice ROUGE for trading between faithfulness and abstractiveness. Our TPG models instead show significant superiority on the ROUGE-1 score. More specifically, TPG-T exceeds the others by a large margin across all metrics on CNN/DM, showing the comprehensive advantages of guided summarization. However, it achieves relatively lower faithfulness scores on XSum. We present in Section 5 that the low informativeness of salient spans in the extreme abstractive setting assumes the main reason. TPG-S is weaker than TPG-T but traces well with CaPE on CNN/DM and shows better faithfulness than the baseline on XSum. In summary, we highlight the advance of prophetic guidance (and prophecy distillation) in promoting faithful abstractive summarization. Under our settings, the highest faithfulness is reached in the moderate abstractive scenario without requiring external datasets or systems.

4.6. Human Evaluation

For human evaluation, we randomly selected 100 samples from each of the CNN/DM and XSum test sets. Three Ph.D. volunteers majoring in English were invited to rate model-generated summaries against the baseline (BART on CNN/DM and PEGASUS on XSum, respectively) summaries regarding faithfulness and informativeness. Table 7 displays the results. On the whole, both the two faithful-oriented methods and ours enhance the baseline a lot. TPG-S achieves a higher win rate of faithfulness on CNN/DM, but its wins in informativeness are relatively less. This finding is basically consistent with (Dreyer et al., 2023), which suggests that faithfulness and abstractiveness are contradictory. Results on XSum further reveal that our approach is more advanced in a moderate-abstractive scenario like CNN/DM.

4.7. Teacher Performance

To test the rationality of our teacher model settings, we compare TPG-T with existing guided summarization models on CNN/DM. The results are reported in Table 8. Following previous works, an additional sequential tagger is trained to predict prophetic guidance in the automatic mode. For our

System	Summary
	Case 1
Reference	RBS has racked up £50 billion in losses since it was bailed out by taxpayers. State-back giant warned that it faces 'another tough year' to come. Lurched to a £446 million loss for the first three months of the year.
BART	RBS has racked up £50 billion in losses since it was bailed out by taxpayers. Coming after RBS was forced to put aside £856 million to cover lawsuits and fines still being decided. *It includes an extra £334 million for its part in rigging the foreign exchange market with other banks.*
TPG-T†	State-backed giant lurched to a £446 million loss for the first three months of the year. RBS has racked up £50 billion in losses since taxpayers bailed it out. State-backed warned that it faces another 'tough year' in a statement last night.
TPG-S	RBS, which was bailed out by taxpayers, has reported losses of £50 billion. The state-backed company has warned that it will face another challenging year ahead. In the first three months of this year, RBS suffered a loss of £446 million.
	Case 2
Reference	Hungarian national Evelin Mezei, 12, has been found safe and well. She had gone missing from the Stratford area in London last night. Evelin had been seen on CCTV footage with an unknown man.
BART	Evelin Mezei, a 12-year-old Hungarian national who went missing from the Stratford area in London last night, has been found safe and well. She was last seen on CCTV footage with an unknown man on a city street, *but thankfully has been located*.
TPG-T†	A 12-year-old Hungarian girl named Evelin Mezei, who went missing from the Stratford area in London last night, has been found safe and well. She was last seen on CCTV footage with an unidentified man.
TPG-S	The Hungarian national Evelin Mezei, 12, went missing in Stratford, London, last night. She was seen on CCTV footage with an unknown man. She was found safe and in good health this morning.

Table 9: Case Study on CNN/DM. † indicates the TPG-T model with oracle-extracted salient spans as guidance. Blue highlights the key faithful facts, while red marks the hallucinations. *Italics* present the irrelevant statements, which degrades the candidate-reference overlap.

TPG-T, we fine-tune a RoBERTa (Liu et al., 2019) to identify the oracle-extracted salient spans. In this setting, all models moderately beat the baseline, and TPG-T shows no advantages over the other two. Since such guided models are trained with oracle-extracted guidance, the results indicate that guided summarization has significantly limited advantages once the quality of guidance varies between training and inference. Also, we perform the comparison in an oracle mode, which enjoys oracle-extracted guidance at inference. TPG-T and CTRLsum outperform GSum, and TPG-T further surpasses CTRLsum on both types of metrics. Considering the main discrepancy among the three models lies in the content of the guidance signal, the salient span we used is more effective than the highlighted sentence and keywords (see Section 5). Finally, we remove [SEP] tokens used to connect salient spans in guidance text, and TPG-T's performance is slightly degraded.

4.8. Case Study

We present two cases sampled from CNN/DM in Table 9 to witness the real efficacy of TPG models. We observe that the baseline BART tends to produce tedious texts containing ambiguous facts unsupported in the gold reference. On the contrary, TPG-T makes fewer mistakes in capturing key facts. However, it yields more extractive-style statements, which are not in line with human habits. Of the three systems, TPG-S produces the summaries that most closely resemble human writings, accurately rephrasing the source document's content with a suitable number of new words.

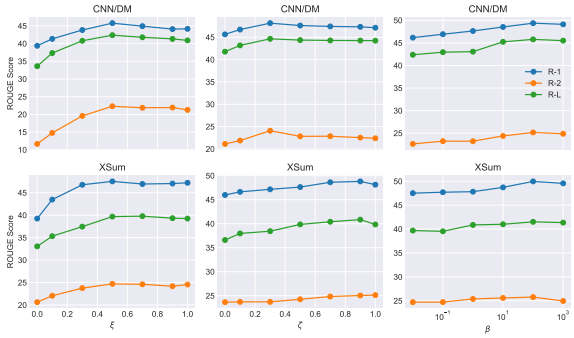


Figure 4: The performance of TPG-S with different hyperparameter settings.

5. More Analysis

5.1. Ablation Study & The Effects of Balance Factors

Three balance factors are introduced in Eq.9, which makes their ablation a bit troublesome. On the one hand, we list the evaluation results of TPG-S without token- and summary-level PD in Tables 4 and 5, confirming that both token- and summary-level distillations are contributed. To further detect the necessity of four types of losses used in our method, we first linearly increase the value of ξ from 0.0 to 1.0 with $\beta = 0$ and plot TPG-S performance on the left of Figure 4. Notably, TPG-S degrades to

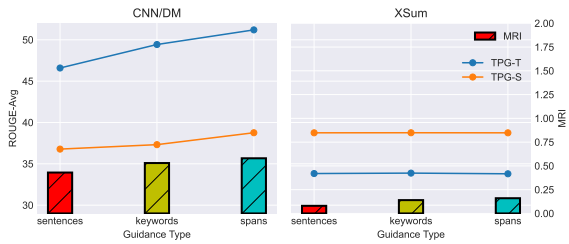


Figure 5: Teacher-student correlation visualizing. The left y-axis sticks ROUGE-Avg (the average of ROUGE-1/2/L F1 scores), and the right one sticks the MRI of each guidance.

the baseline at $\xi = 1$. The curve shows a similar tendency of attenuating after stable rising on both datasets. We see that the best value should be around 0.5. Next, we set ξ to 1 and β to 100 to offset the effect of toke-level PD and then find the optimal value of ζ. The process is depicted in the middle of Figure 4. We find that a larger ζ is preferred on XSum, while a small value is more compatible with CNN/DM. Recalling the previous analysis, the candidate produced by TPG-T differs significantly from that by the baseline on CNN/DM, but the situation is the opposite on XSum. It makes the soft contrastive learning more influential on CNN/DM. As a result, we set ζ to 0.3 on CNN/DM and 0.9 on XSum, respectively. Finally, we fix ξ and ζ to their optimal value and vary β from 0.01 to 100. According to the results in the right of Figure 4, $\beta = 100$ is the best for both datasets.

5.2. Guidance Choice & Teacher-Student Correlation

Although we have compared TPG-T with the peer models in Section 4.7, there are still unclearities: what impacts TPG-T performance and the correlation between TPG-T and TPG-S. In this paper, we focus on the amount of information that guidance leaks to the model, which can be quantified as the following mutual information recall (MIR):

$$MIR(\mathbf{G}, \mathbf{Y}) = \frac{|\{\mathbf{G}\} \cap \{\mathbf{Y}\}|}{|\{\mathbf{Y}\}|}, \qquad (11)$$

where $\{\cdot\}$ denotes token set. We use different kinds of guidance to learn TPG models and show testing results in Figure 5. *Firstly, the higher the MIR of guidance, the better the performance of the TPG models*. Our introduced salient spans achieve higher MIR than keywords and highlighted sentences, leading to the TPG-T best performance in Table 7. In contrast, salient spans share rare mutual information with the reference in the extreme abstractive setting, and TPG-T beats the baseline just a little on XSum. *Second, the performance of TPG-S echoes with that of TPG-T*. On CNN/DM, the ROUGE-Avg score of both TPG models improved with the MIR of guidance increases. As for XSum, we use a small ζ to limit the impact of TPG-T to TPG-S. In this way, the improved performance of TPG-S is mainly brought by token-level label smoothing and summary-level hard contrastive learning rather than exact prophecy distillation. This points out the direction for our future work, finding appropriate guidance for extreme summarization.

6. Conclusion

In this paper, we introduce a novel teacher-student framework - TPG, which learns a regular summarization model to mimic the behavior of a guided one via token- and summary-level knowledge distillations. TPG boosts abstractive summarization with the distilled prophecy and suffers few train-inference discrepancies, leading to compressively improved performance on four well-known benchmarks. In the future, we will dive into the diversity of prophecies to extend our method to more complex scenarios, such as multi-documents.

7. Acknowledgements

We sincerely appreciate the anonymous reviewers for their valuable suggestions and approval. This work is supported by the Changsha Science and Technology Major Special Project (No.kh2202006).

8. Bibliographical References

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 1171–1179.

Shuyang Cao and Lu Wang. 2021. CLIFF: contrastive learning for improving faithfulness and factuality in abstractive summarization. In *EMNLP 2021*, pages 6633–6649.

Prafulla Kumar Choubey, Alexander R. Fabbri, Jesse Vig, Chien-Sheng Wu, Wenhao Liu, and Nazneen Rajani. 2023. Cape: Contrastive parameter ensembling for reducing hallucination in abstractive summarization. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10755–10773.

Zi-Yi Dou, Pengfei Liu, Hiroaki Hayashi, Zhengbao Jiang, and Graham Neubig. 2021. Gsum: A

general framework for guided neural abstractive summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 4830–4842. Association for Computational Linguistics.

Markus Dreyer, Mengwen Liu, Feng Nan, Sandeep Atluri, and Sujith Ravi. 2023. Evaluating the tradeoff between abstractiveness and factuality in abstractive summarization. In *Findings of the Association for Computational Linguistics: EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 2044–2060. Association for Computational Linguistics.

Alexander R. Fabbri, Chien-Sheng Wu, Wenhao Liu, and Caiming Xiong. 2022. Qafacteval: Improved qa-based factual consistency evaluation for summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 2587–2601. Association for Computational Linguistics.

Angela Fan, David Grangier, and Michael Auli. 2018. Controllable abstractive summarization. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation, NMT@ACL 2018, Melbourne, Australia, July 20, 2018*, pages 45–54. Association for Computational Linguistics.

Sebastian Goodman, Nan Ding, and Radu Soricut. 2020. Teaforn: Teacher-forcing with n-grams. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 8704–8717. Association for Computational Linguistics.

Anirudh Goyal, Alex Lamb, Ying Zhang, Saizheng Zhang, Aaron C. Courville, and Yoshua Bengio. 2016. Professor forcing: A new algorithm for training recurrent networks. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 4601–4609.

Junxian He, Wojciech Kryscinski, Bryan McCann, Nazneen Rajani, and Caiming Xiong. 2022. Ctrlsum: Towards generic controllable text summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 5879–5915. Association for Computational Linguistics.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531.

Yuta Kikuchi, Graham Neubig, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. 2016. Controlling output length in neural encoder-decoders. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1328–1338. The Association for Computational Linguistics.

Yoon Kim and Alexander M. Rush. 2016. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1317–1327. The Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *EMNLP 2020*, pages 9332–9346.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.

Haoran Li, Junnan Zhu, Jiajun Zhang, and Chengqing Zong. 2018a. Ensure the correctness of the summary: Incorporate entailment knowledge into abstractive sentence summarization. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 1430–1441. Association for Computational Linguistics.

Wei Li, Xinyan Xiao, Yajuan Lyu, and Yuanzhuo Wang. 2018b. Improving neural abstractive document summarization with structural regularization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November*

4, 2018, pages 4078–4087. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 3728–3738. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Yixin Liu and Pengfei Liu. 2021. Simcls: A simple framework for contrastive learning of abstractive summarization. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 2: Short Papers), Virtual Event, August 1-6, 2021*, pages 1065–1072. Association for Computational Linguistics.

Yixin Liu, Pengfei Liu, Dragomir R. Radev, and Graham Neubig. 2022. BRIO: bringing order to abstractive summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 2890–2903. Association for Computational Linguistics.

Yizhu Liu, Zhiyi Luo, and Kenny Q. Zhu. 2018. Controlling length in abstractive summarization using a convolutional neural network. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 4110–4119. Association for Computational Linguistics.

Rafael Müller, Simon Kornblith, and Geoffrey E. Hinton. 2019. When does label smoothing help? In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 4696–4705.

Ramakanth Pasunuru and Mohit Bansal. 2018. Multi-reward reinforced summarization with saliency and entailment. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 646–653. Association for Computational Linguistics.

Ramakanth Pasunuru, Han Guo, and Mohit Bansal. 2017. Towards improving abstractive summarization via entailment generation. In *Proceedings of the Workshop on New Frontiers in Summarization, NFiS@EMNLP 2017, Copenhagen, Denmark, September 7, 2017*, pages 27–32. Association for Computational Linguistics.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Maxime Peyrard. 2019. A simple theoretical model of importance for summarization. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1059–1073. Association for Computational Linguistics.

Paul Roit, Johan Ferret, Lior Shani, Roee Aharoni, Geoffrey Cideron, Robert Dadashi, Matthieu Geist, Sertan Girgin, Léonard Hussenot, Orgad Keller, Nikola Momchev, Sabela Ramos Garea, Piotr Stanczyk, Nino Vieillard, Olivier Bachem, Gal Elidan, Avinatan Hassidim, Olivier Pietquin, and Idan Szpektor. 2023. Factually consistent summarization via reinforcement learning with textual entailment feedback. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 6252–6272. Association for Computational Linguistics.

Xiangru Tang, Arjun Nair, Borui Wang, Bingyao Wang, Jai Desai, Aaron Wade, Haoran Li, Asli Celikyilmaz, Yashar Mehdad, and Dragomir R. Radev. 2022. CONFIT: toward faithful dialogue summarization with linguistically-informed contrastive fine-tuning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 5657–5668. Association for Computational Linguistics.

Ashwin K. Vijayakumar, Michael Cogswell, Ramprasaath R. Selvaraju, Qing Sun, Stefan Lee, David J. Crandall, and Dhruv Batra. 2016. Diverse beam search: Decoding diverse solutions from neural sequence models. *CoRR*, abs/1610.02424.

Bin Wang, Chen Zhang, Yan Zhang, Yiming Chen, and Haizhou Li. 2022. Analyzing and evaluating faithfulness in dialogue summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 4897–4908. Association for Computational Linguistics.

Jiawen Xie, Qi Su, Shaoting Zhang, and Xiaofan Zhang. 2023. Alleviating exposure bias via multi-level contrastive learning and deviation simulation in abstractive summarization. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 9732–9747. Association for Computational Linguistics.

Haopeng Zhang, Semih Yavuz, Wojciech Kryscinski, Kazuma Hashimoto, and Yingbo Zhou. 2022a. Improving the faithfulness of abstractive summarization via entity coverage control. In *Findings of the Association for Computational Linguistics: NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 528–535. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020a. PEGASUS: pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 11328–11339. PMLR.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020b. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Xingxing Zhang, Yiran Liu, Xun Wang, Pengcheng He, Yang Yu, Si-Qing Chen, Wayne Xiong, and Furu Wei. 2022b. Momentum calibration for text generation. *CoRR*, abs/2212.04257.

Yao Zhao, Misha Khalman, Rishabh Joshi, Shashi Narayan, Mohammad Saleh, and Peter J. Liu. 2022. Calibrating sequence likelihood improves conditional language generation. *CoRR*, abs/2210.00045.

Chenguang Zhu, William Hinthorn, Ruochen Xu, Qingkai Zeng, Michael Zeng, Xuedong Huang, and Meng Jiang. 2021. Enhancing factual consistency of abstractive summarization. In *NAACL-HLT 2021*, pages 718–733.

9. Language Resource References

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. Samsum corpus: A human-annotated dialogue dataset for abstractive summarization. *CoRR*, abs/1911.12237.

Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 1693–1701.

Elena Mozzherina. 2013. An approach to improving the classification of the new york times annotated corpus. In *Knowledge Engineering and the Semantic Web - 4th International Conference, KESW 2013, St. Petersburg, Russia, October 7-9, 2013. Proceedings*, volume 394 of *Communications in Computer and Information Science*, pages 83–91. Springer.

Ramesh Nallapati, Bowen Zhou, Cícero Nogueira dos Santos, Çaglar Gülçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Berlin, Germany, August 11-12, 2016*, pages 280–290. ACL.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1797–1807. Association for Computational Linguistics.