Noisy Self-Knowledge Distillation for Text Summarization

Yang Liu[♡], Sheng Shen[♢], and Mirella Lapata[△]

[♡]Microsoft Cognitive Services Research
[♢]University of California, Berkeley
[△]School of Informatics, University of Edinburgh
yaliu10@microsoft.com   sheng.s@berkeley.edu   mlap@inf.ed.ac.uk

Abstract

In this paper we apply self-knowledge distillation to text summarization which we argue can alleviate problems with maximum-likelihood training on single reference and noisy datasets. Instead of relying on one-hot annotation labels, our student summarization model is trained with guidance from a teacher which generates smoothed labels to help regularize training. Furthermore, to better model uncertainty during training, we introduce multiple noise signals for both teacher and student models. We demonstrate experimentally on three benchmarks that our framework boosts the performance of both pretrained and non-pretrained summarizers achieving state-of-the-art results.[1]

1 Introduction

Automatic summarization has enjoyed renewed interest in recent years, thanks to the popularity of neural network models and their ability to learn continuous representations without recourse to pre-processing tools or linguistic annotations. The availability of large-scale datasets (Sandhaus, 2008; Hermann et al., 2015; Grusky et al., 2018; Narayan et al., 2018) containing hundreds of thousands of document-summary pairs has driven the development of neural architectures for summarization. Several approaches have been proposed, in the vast majority sequence-to-sequence models which are trained in an end-to-end fashion with a maximum likelihood estimation loss (See et al., 2017; Celikyilmaz et al., 2018; Paulus et al., 2018; Gehrmann et al., 2018).

Despite promising results, there are specific characteristics of the summarization task which render it ill-suited to standard sequence-to-sequence training. For instance, maximum-likelihood training on *single* reference datasets might not be optimal for summarization which is subject to a great deal of human variation (Harman and Over, 2004; Nenkova, 2006). In the context of extractive summarization, different people select different sentences to include in a summary (Rath et al., 1961), and when writing abstracts, disagreement exists both in terms of writing style and the specific content deemed important for the summary (Harman and Over, 2004). Although summarization models would naturally benefit from multiple target references, it is unrealistic to expect that multi-reference datasets can be created at scale for neural network training. In fact, most popular benchmarks are collated opportunistically, based on summaries which only loosely correspond to the source input.

For example, Narayan et al. (2018) create a dataset by pairing the first sentence of a news article with the rest of the document under the assumption that the introductory sentence expresses the gist of the article. Grusky et al. (2018) pair articles with metadata available in HTML pages under the assumption that HTML tags (e.g., *description*) denote summary-like content. In other work (Liu et al., 2018; Perez-Beltrachini et al., 2019), multi-document summarization datasets are created by viewing lead sections in Wikipedia articles as summaries of documents cited therein. The inherent *noise* in the data collection process further hampers training with models often being prone to hallucination (Song et al., 2018; Maynez et al., 2020), and struggling to identify which content units are salient (Tan et al., 2017).

In this paper, we propose to alleviate these problems by turning to *knowledge distillation* (Bucilu et al., 2006; Ba and Caruana, 2014; Hinton et al., 2015; Kim and Rush, 2016). Knowledge distillation transfers knowledge from a larger "teacher" network to a smaller "student" model by training the student to imitate the teacher's outputs (in addition to learning from the training data set). In "born-again networks", (Furlanello et al., 2018) the teacher and student have the *same* neural archi-

---

[1]Our code is available at https://github.com/nlpyang/NoisySumm.

tecture and model size, and yet surprisingly the student is able to surpass the teacher's accuracy. Intuitively, such *self-knowledge* distillation is effective because the teacher's output distribution provides a richer training signal capturing additional information about training examples. In the context of summarization, the teacher can benefit student training in two ways. It provides a softened distribution over reference summaries thereby enriching the single reference setting. Moreover, the teacher's distribution is (to a certain extent) denoised enabling the student to circumvent inaccuracies in the training data. We further capitalize on the idea that both the teacher and the student should be robust to noise and introduce several noise injection techniques which together with knowledge distillation improve model generalization and performance.

We present experiments on several summarization benchmarks (Narayan et al., 2018; Perez-Beltrachini et al., 2019; Hermann et al., 2015) covering single- and multi-document summarization settings as well as different types of summaries (e.g., verbose or more telegraphic). Across datasets, the proposed framework boosts the performance of pretrained and non-pretrained abstractive summarizers, achieving new state-of-the-art results.

2 Background

2.1 Neural Abstractive Summarization

Neural approaches to abstractive summarization conceptualize the task as a sequence-to-sequence problem, where the encoder maps the sequence of tokens in the source document $\mathbf{x} = [x_1, ..., x_n]$ to a sequence of continuous representations $\mathbf{z} = [z_1, ..., z_n]$, and the decoder autoregressively generates the target summary $\mathbf{y} = (y_1, ..., y_m)$ token-by-token, hence modeling the conditional probability $p(y_1, ..., y_m | x_1, ..., x_n)$.

Rush et al. (2015) and Nallapati et al. (2016) were among the first to apply the neural encoder-decoder architecture to text summarization. See et al. (2017) enhance this model with a pointer-generator network which allows to copy words from the source text, and a coverage mechanism which keeps track of words that have been summarized. Other work develops abstractive models trained end-to-end with reinforcement learning based on multiple encoders and hierarchical attention (Celikyilmaz et al., 2018) or a coverage mechanism where the decoder attends over previously

generated words (Paulus et al., 2018). Gehrmann et al. (2018) follow a bottom-up approach where a content selector first determines which phrases in a source document should be part of the summary, and a copy mechanism is applied only to preselected phrases during decoding. Although the majority of summarization systems are composed of LSTM units, Narayan et al. (2018) and (Perez-Beltrachini et al., 2019) propose abstractive models based on convolutional neural networks.

Pretrained language models have recently emerged as a key technology for achieving impressive gains in abstractive summarization (Liu and Lapata, 2019; Lewis et al., 2020; Song et al., 2019). These models first pretrain a language model with self-supervised objectives on large corpora and then fine-tune it on summarization datasets. Liu and Lapata (2019) combine a pretrained encoder based on BERT (Devlin et al., 2019) with a randomly initialized decoder, demonstrating substantial gains on summarization performance. Song et al. (2019) pretrain an encoder-decoder framework to reconstruct (masked) fragments within a sentence and then fine-tune it on summarization datasets. In the same vein, Lewis et al. (2020) present BART, an encoder-decoder Transformer (Vaswani et al., 2017), pretrained by reconstructing a text corrupted with several arbitrary noising functions. Bao et al. (2020) design UNILMv2, a Transformer-based neural network pretrained as a pseudo-masked language model. Qi et al. (2020) introduce their own novel self-supervised task based on future n-gram prediction.

2.2 Knowledge Distillation

Knowledge Distillation refers to a class of methods for training a new smaller *student* network by learning from a *teacher* network (in addition to learning from the training data). It is generally assumed that the teacher has been previously trained, and the parameters for the student are estimated by matching the student's predictions to the teacher.

Let T and S denote teacher and student models, respectively. Let f_T and f_S be functions of the teacher and student. The models are typically neural networks and function f can be in principle defined using the output of any network layer (e.g., a hidden or softmax layer). Knowledge distillation methods are commonly expressed as minimizing

an objective function over training set $\mathcal{X}$:

$$\mathcal{L}_{KD} = \sum_{x_i \in \mathcal{X}} l(f_T(x_i), f_S(x_i)) \qquad (1)$$

where $l()$ is a loss function that penalizes the difference between the teacher and the student.

Specific instantiations of this general framework include minimizing the teacher/student difference based on output logits, intermediate hidden representations, attention maps, and derivatives of the loss to the input (Ba and Caruana, 2014; Romero et al., 2014; Zagoruyko and Komodakis, 2017; Czarnecki et al., 2017). Other work integrates an ensemble of teachers in order to improve the student (Urban et al., 2016), trains a succession of students (Furlanello et al., 2018), introduces a "teacher assistant" for better knowledge transfer (Mirzadeh et al., 2019), and regularizes multi-task agents (Parisotto et al., 2015; Teh et al., 2017) in reinforcement learning. Compared to direct training, knowledge distillation provides a more stable training process which leads to better performing student models (Hinton et al., 2015; Phuong and Lampert, 2019). Recent work (Furlanello et al., 2018; Hahn and Choi, 2019) also sheds light on leveraging knowledge distillation for training a high-performing student model with the same size as the teacher (see the discussion in the next section).

Knowledge distillation has been also shown to improve results for various NLP tasks. Tan et al. (2019) use it to transfer knowledge from BERT to smaller models, helping them approach or exceed the quality of much larger pretrained neural networks. Aside from distilling large models into smaller ones (Kim and Rush, 2016; Mou et al., 2016) or ensembles of models into single models (Kuncoro et al., 2016; Liu et al., 2019), knowledge distillation has been further used in multi-task learning, e.g., to teach a multi-task student from single-task teachers (Clark et al., 2019).

3 Self-Knowledge Distillation for Text Summarization

Self-knowledge distillation refers to the special case where the teacher and student have *identical* neural network architectures. Surprisingly, perhaps, it has been consistently observed (Furlanello et al., 2018; Yang et al., 2019; Ahn et al., 2019; Liu et al., 2020) that students trained with self-knowledge distillation outperform their teachers by significant margins in several computer vision and language

modeling tasks. Recent efforts have also focused on understanding why this happens, e.g., by observing that knowledge transferred by the teacher is localized mainly in higher layers and does not affect early (feature extraction) layers much (Gotmare et al., 2019), by interpreting the teacher's knowledge as importance weighting (Furlanello et al., 2018), by showing that early-stopping is crucial (Dong et al., 2019), and by studying how self-distillation modifies regularization (Mobahi et al., 2020).

For text summarization, we argue that self-knowledge distillation can potentially alleviate problems in conventional maximum likelihood training. Summarization models are typically trained on single reference document-summary pairs, however considering a single summary as the only correct reference during maximum likelihood training can harm model generalization (Elbayad et al., 2018) and is counter-intuitive. There can be multiple valid summaries for a source input (Harman and Over, 2004; Nenkova, 2006) and even the single reference summaries available are not entirely goldstandard due to the inherent noise in the automatic construction of large-scale summarization datasets (Kryściński et al., 2019). With self-knowledge distillation, teacher outputs provide softened distributions of the reference summaries, which can be viewed as an enrichment of the single reference setting and a reweighting of gold summaries to prevent the student from becoming overconfident in its predictions.

The standard objective for an abstractive summarization model is negative log likelihood:

$$\mathcal{L}_{\text{NLL}} = -\sum_{t=1}^{T} log(p(y_t|y_1^{t-1}, x)) \qquad (2)$$

where x indicates the source document, y_1^t indicates the t-th token in the target summary and y_1^{t-1} are the first $t-1$ tokens in the target summary. We further assume that the teacher is a fully trained neural model, the student has the same architecture with the teacher, and access to the learned teacher's output distribution $p_T(y_t|y_{1:t-1}, x))$:

$$\mathcal{L}_{\text{KD}} = \sum_{t=1}^{T} \text{KL}(p_T(y_t|y_1^{t-1}, x), p_S(y_t|y_1^{t-1}, x)) \qquad (3)$$

where $p_T(y_t|y_1^{t-1}, x)$ and $p_S(y_t|y_1^{t-1}, x)$ are model outputs from the teacher and student, respectively.

It is common practice to compensate for no direct access to the training data (see Equation (3)) by interpolating between the two losses in Equations (3) and (2). So, the final objective for training the student becomes:

$$\mathcal{L}_{\text{FINAL}} = (1 - \lambda)\mathcal{L}_{\text{NLL}} + \lambda\mathcal{L}_{\text{KD}} \qquad (4)$$

where λ is a mixture parameter combining the one-hot distribution and the teacher distribution.

We further want our summarization systems to be robust to natural noise found in existing datasets. Injecting noise onto training samples has been proven useful for improving model generalization (Xie et al., 2019). We extend this idea for knowledge distillation, and propose a novel framework for introducing noise to both distillation signals and training data. We design different noise mechanisms for the teacher and student, and select the best noise configuration experimentally.

Noisy Teacher   To inject noise into the distillation signals, we incorporate a *teacher dropout* mechanism (Bulò et al., 2016), where dropout is kept active while generating teacher predictions for training the student. In this manner, the teacher generates variable supervision labels for the student with some degree of uncertainty, alleviating the problem of overfitting to the teacher predictions. Meanwhile, it can also be considered as approximating an average ensemble from many neural networks (Bulò et al., 2016).

The knowledge distillation loss now becomes:

$$\mathcal{L}_{\text{KD}} = \sum_{t=1}^{T} \text{KL}(\tilde{p}_T^\alpha(y_t|y_1^{t-1}, x), p_S(y_t|y_1^{t-1}, x)) \quad (5)$$

where $\tilde{p}_T^\alpha$ indicates the predictions from the teacher model with active dropout α.

Noisy Student   To inject noise into the training data, we propose various mechanisms to perturb the source input. Random perturbation is effective in enforcing local smoothness for training text generation models under the assumption that semantically similar inputs can be mapped to the same or similar targets. A related approach has been shown to improve the performance of machine translation models in self-training settings (He et al., 2019). For text summarization, where the input is usually a long document, we design the following perturbation policies:

1. *Word Drop*: a word in the source document is removed with probability p_d.

2. *Word Replacement*: for each word x_i in the source document, we calculate a candidate replacement list by selecting k words most similar to x_i from the vocabulary. The similarity is calculated as the cosine distance between the embedding of x_i and embeddings of all other words in the vocabulary. Then, a source word is replaced with a word randomly selected from its candidate replacement list with probability p_r.

3. *Sentence Drop*: a sentence in the source document is removed with probability p_s.

4. *Gaussian Noise*: a Gaussian noise vector $\mathbf{e}$ is multiplied with the embeddings $\mathbf{x}$ of input words: $\mathbf{x} \leftarrow \mathbf{x} \otimes \mathbf{e}, \mathbf{e} \sim N(I, \sigma^2 I)$.

These perturbation policies can be applied simultaneously or successively as a pipeline. We experimentally found the best combination for our task to be the sequential application of word drop, followed by word replacement, and sentence drop. Although Gaussian noise has been effective in natural language understanding tasks (Zhang and Yang, 2018), we found it not to be helpful in our summarization experiments. The knowledge distillation loss with a student trained on noisy data becomes:

$$\mathcal{L}_{\text{KD}} = \sum_{t=1}^{T} \text{KL}(\tilde{p}_T^\alpha(y_t|y_1^{t-1}, x), p_S(y_t|y_1^{t-1}, \tilde{x})) \quad (6)$$

where $\tilde{x}$ indicates perturbed source input.

4   Experimental Setup

In this section, we describe the summarization datasets used in our experiments and discuss various implementation details.

4.1   Summarization Datasets

We evaluated our model on two single-document summarization datasets, namely the CNN/DailyMail news highlights (Hermann et al., 2015) and XSum (Narayan et al., 2018), and one multi-document summarization dataset, i.e., WikiCatSum (Perez-Beltrachini et al., 2019). These datasets represent different summary styles ranging from highlights to very brief-one sentence summaries. The summaries also vary with respect to the type of rewriting operations they exemplify (e.g., CNN/DailyMail showcases more cut and paste operations while XSum is genuinely abstractive). Finally, two of these

		CNN/DailyMail			XSum		
Without Pretraining		R1	R2	RL	R1	R2	RL
LEAD		40.42	17.62	36.67	16.30	1.60	11.95
PTRNET		39.53	17.28	36.38	28.10	8.02	21.72
TransformerAbs		40.21	17.76	37.09	31.04	10.48	24.54
+SKD		40.64	18.10	37.43	32.22	11.45	25.56
+SKD +Noisy T		40.79	18.24	37.57	32.32	11.56	25.72
+SKD +Noisy T +Noisy S		40.86	18.27	37.66	32.76	11.88	26.07
BASE-size Pretrained Models		R1	R2	RL	R1	R2	RL
MASS$_{BASE}$	(123M)	42.12	19.50	39.01	39.75	17.24	31.95
BERTSUMABS	(156M)	41.72	19.39	38.76	38.76	16.33	31.15
UNILMv2$_{BASE}$	(110M)	43.45	20.71	40.49	43.69	20.71	35.73
+SKD	(110M)	43.44	20.68	40.51	43.76	21.04	36.04
+SKD +Noisy T	(110M)	43.59	21.01	40.66	44.11	21.30	36.32
+SKD +Noisy T +Noisy S	(110M)	43.77	20.98	40.82	44.14	21.34	36.35
LARGE-size Pretrained Models		R1	R2	RL	R1	R2	RL
UNILM$_{LARGE}$	(340M)	43.08	20.43	40.34	—	—	—
BART$_{LARGE}$	(400M)	44.16	21.28	40.90	45.14	22.27	37.25
T5$_{11B}$	(11B)	42.05	20.34	39.40	—	—	—

Table 1: ROUGE F1 results on **CNN/DailyMail** and **XSUM** test sets (R1 and R2 are shorthands for unigram and bigram overlap; RL is the longest common subsequence). *SKD* refers to a system trained with self-knowledge distillation, *Noisy T* are SKD models trained with noisy signals while *Noisy S* are student models trained on noisy data. Results for comparison systems are taken from the authors' respective papers or obtained on our data by running publicly released software.

datasets (XSum and WikiCatSum) were created automatically following various assumptions about the correspondence of purported summaries to the source input.

CNN/DailyMail contains news articles and associated highlights, i.e., a few bullet points written by journalists which give a brief overview of the article. We used the standard splits of Hermann et al. (2015) for training, validation, and testing (90,266/1,220/1,093 CNN documents and 196,961/12,148/10,397 DailyMail documents). We did not anonymize entities. Sentences were split with the Stanford CoreNLP toolkit (Manning et al., 2014) and the dataset was pre-processed following See et al. (2017). Input documents were truncated to 512 tokens.

XSum contains 226,711 news articles accompanied with a one-sentence summary, answering the question "What is this article about?". We used the splits of Narayan et al. (2018) for training, validation, and testing (204,045/11,332/11,334) and followed the pre-processing introduced in their work. Input documents were also truncated to 512 tokens.

WikiCatSum is a multi-document summarization dataset derived from WikiSum (Liu et al., 2018). The target summary is the lead section of a Wikipedia article, and the source input are webpages related to this article. WikiCatSum (Perez-Beltrachini et al., 2019) represents three domains from the original Wikisum dataset under the assumption that these vary in terms of the topics the summaries discuss and their linguistic characteristics. Aside from the summaries, the dataset contains the input webpages whose length is truncated to the first 800 tokens. WikiCatSum contains 62,545 samples for the Company domain, 59,973 samples for the Film domain, and 60,816 samples for the Animal domain.

4.2 Implementation Details

For all datasets, we evaluated our self-knowledge distillation framework in two settings. In the first setting, our models are *non-pretrained* while in the second setting we take advantage of *pretrained* language models which have demonstrated impressive improvements in summarization (Lewis et al., 2020; Liu and Lapata, 2019; Bao et al., 2020).

Specifically, we adopt UNILMv2 (Bao et al., 2020) as the pretrained model. UNILMv2 is a Transformer-based neural network (Vaswani et al., 2017) with 12 Transformer layers and 12 attention

	Company			Film			Animal			All		
Without Pretraining	R1	R2	RL	R1	R2	RL	R1	R2	RL	R1	R2	RL
CV-S2S	24.5	9.4	19.9	34.6	19.8	30.7	42.2	28.4	38.5	33.8	19.2	29.7
CV-S2D	27.6	10.5	21.3	37.7	20.8	32.0	42.3	27.3	37.1	35.9	19.5	30.1
TF-S2S	26.0	9.5	20.4	36.5	18.8	31.0	44.0	28.8	40.0	35.5	19.0	30.5
+SKD	26.8	9.9	20.9	37.2	19.3	31.8	44.3	29.0	40.3	36.1	19.4	31.0
+SKD +Noisy T	27.2	10.3	21.0	37.7	20.6	32.0	44.6	29.1	40.4	36.5	20.0	31.1
+SKD +Noisy T +Noisy S	27.4	10.4	21.3	37.9	21.0	32.2	44.6	29.0	40.4	36.6	20.1	31.3
With Pretraining	R1	R2	RL	R1	R2	RL	R1	R2	RL	R1	R2	RL
UNILMv2$_{BASE}$	33.32	14.36	25.39	42.51	25.92	36.54	45.45	31.69	40.91	40.4	24.0	34.3
+SKD	33.20	14.66	25.53	42.39	25.90	36.53	45.59	31.87	41.12	40.4	24.1	34.4
+SKD +Noisy T	33.42	14.87	25.80	42.60	26.02	36.65	45.75	32.19	41.30	40.6	24.4	34.6
+SKD +Noisy T +Noisy S	33.50	14.95	25.85	42.71	26.09	36.77	45.86	32.23	41.40	40.7	24.4	34.7

Table 2: ROUGE F1 results on **WikiCatSum** test sets (R1 and R2 are shorthands for unigram and bigram overlap; RL is the longest common subsequence). Results are reported separately on three domains and in combination (All). *SKD* refers to systems trained with self-knowledge distillation, *Noisy T* are SKD systems trained with noisy signals, and *Noisy S* are SKD students trained on noisy data. Results for comparison systems are taken from the authors' respective papers or obtained on our data by running publicly released software.

heads. It is pretrained as a pseudo-masked language model on a large corpus (label smoothing is applied with smoothing factor 0.1). We fine-tuned our teacher models following the procedure outlined in Bao et al. (2020). In the non-pretrained setting, we adopt a Transformer encoder-decoder model with 6 layers, 768 hidden size and 2,048 feed-forward filter size. Label smoothing was also used with smoothing factor 0.1. All teacher models in this setting were trained from randomly initialized parameters following Liu and Lapata (2019).

In all knowledge distillation experiments, student models have the same neural network architecture with their teachers and are trained with the same hyperparameters as the teacher models. The best teacher and student model are selected by evaluating perplexity on the development set. For *noisy* distillation models, word drop probability p_d was set to 0.1. The candidate length k for word replacement was 10 and word replacement probability p_r was 0.1. Sentence drop probability p_s was 0.05.

During decoding we used beam search (size 5), and tuned α for the length penalty (Wu et al., 2016) between 0.6 and 1 on the validation set; we decode until an end-of-sequence token is emitted. Repeated trigrams are blocked (Paulus et al., 2018).

5 Results

5.1 Automatic Evaluation

We evaluated summarization quality automatically using ROUGE (Lin, 2004). We report unigram and bigram overlap (ROUGE-1 and ROUGE-2) as a means of assessing informativeness and the longest common subsequence (ROUGE-L) as a means of assessing fluency. Examples of system output are shown in Table 5.

Table 1 summarizes our results on the CNN/DailyMail and XSum (single document) datasets. The first block includes the results of non-pretrained models. We present the LEAD baseline (which simply selects the first three sentences in a document for CNN/DailyMail and the first sentence for XSum). We also report the results of See et al.'s (2017) pointer generator network (PTRNET), and an abstractive system from Liu and Lapata (2019) based on Transformers (TransformerAbs; see Section 4.2 for details). The latter forms the backbone of our self-knowledge distillation models (SKD). We present a variant without noise (+SKD), a variant with noise in the teacher training signal (+Noisy T), and a third variant where the student is additionally trained on noisy data (+Noisy S).

The second and third blocks in Table 1 include the results of pretrained models. To make comparisons fairer, we separate LARGE- (second block) from BASE-size (third block) pretrained models based on parameter size (shown within parentheses). With regard to LARGE-size models, we report the results of three very strong summarization systems finetuned with UNILM$_{LARGE}$ (Bao et al., 2020), BART$_{LARGE}$ (Lewis et al., 2020), and T5$_{11B}$ (Raffel et al., 2019). Our BASE-size models include BERTSUM$_{BASE}$ (Liu and Lapata, 2019), a

Models	CNN/DailyMail	XSum
TRANSFORMERABS	20.8	32.7
+Noisy SKD	21.4	33.6
UNILMv2$_{\text{BASE}}$	23.7	38.7
+Noisy SKD	24.8	39.9

Table 3: Factual correctness on CNN/DailyMail and XSum test set. +Noisy SKD are students trained on noisy signals and noisy data.

CNN/DailyMail	Succinct	Inform	Fluent
UNILMv2$_{\text{BASE}}$	0.47	0.40	0.54
+Noisy SKD	0.53	0.60	0.46

XSum	Succinct	Inform	Fluent
UNILMv2$_{\text{BASE}}$	0.46	0.36	0.53
+Noisy SKD	0.54	0.64	0.47

WikiCatSum	Company	Film	Animal
UNILMv2$_{\text{BASE}}$	0.62	0.47	0.45
+Noisy SKD	0.38	0.53	0.55

Table 4: Human evaluation on CNN/DailyMail, XSum, and WikicatSum test sets. +Noisy SKD is UNILMv2$_{\text{BASE}}$ trained with self-knowledge distillation (on noisy signals and noisy data). All pairwise differences between systems are significant ($p < 0.05$) using a paired t-test.

summarizer based on a BASE-size BERT encoder and a randomly initialized decoder, MASS$_{\text{BASE}}$ (Song et al., 2019) and UNILM$_{\text{BASE}}$ which are both finetuned with BASE-size pretrained models.

As can be seen in Table 1, SKD improves over teacher models in both pretrained (BASE-size) and non-pretrained settings. We also observe that injection of noise brings further improvements with noise in the training signal (+Noisy T) seeming more effective compared to noisy data augmentation (+Noisy S). Overall, we obtain competitive results with SKD and BASE-size pretrained models and even manage to outperform UNILM$_{\text{LARGE}}$ and T5$_{\text{11B}}$ on the CNN/DailyMail dataset.

Table 2 presents experimental results on the WikiCatSum dataset. The first block in the table includes results for non-pretrained models. CV-S2S and CV-S2D (Perez-Beltrachini et al., 2019) are convolutional encoder-decoder models. The former is a standard convolutional decoder, while the latter adopts a hierarchical convolutional decoder which first generates target sentence vectors, and then generates target words based on sentence vectors. TF-S2S is a standard Transformer encoder-decoder model trained on WikiCatSum (Perez-Beltrachini et al., 2019). TF-S2S is the model used in our SKD system and its noisy version (+Noisy T, +Noisy S). The second block includes the results of a system using the BASE-size pretrained model UNILM$_{\text{BASE}}$ on its own and with SKD. Results are reported per domain (Company, Film, and Animal) and across domains (All).

Under pretrained and non-pretrained settings, we observe that SKD boosts the performance of the teacher model (UNILM$_{\text{BASE}}$ and TF-S2S, respectively) and that the injection of noise is beneficial. Improvements in performance vary across domains, with Film showing the least gains. Column All in Table 2 shows average ROUGE across domains. Although SKD and noise injection improve results, we observe that non-pretrained models benefit more.

5.2 Factual Consistency Evaluation

Besides ROUGE, we also use FactCC (Kryściński et al., 2019) to evaluate the factual correctness of the generated summaries. FactCC is a BERT-based classifier trained to identify conflicts between a source document and a generated summary. Given a document-sentence pair as input, it assigns a positive label if factual information mentioned in a summary sentence is consistent with the document, otherwise it assigns a negative label. We view the percentage of positive labels assigned by FactCC to all generated summaries as a factual correctness score for a summarization system.

We performed experiments with the publicly released version of FactCC.[2] Our results on the CNN/DailyMail and XSum datasets are presented in Table 3. Here, we only focus on single-document summarization, as there is no version of FactCC trained on multi-document datasets. As can be seen, the application of SKD (trained with noisy signals and on noisy data) improves factual consistency for non-pretrained and pretrained models on both datasets. All +Noisy SKD students are significantly ($p < 0.05$) more factually correct compared to their teachers (TransformerAbs and UNILMv2$_{\text{BASE}}$), using a paired student t-test.

5.3 Human Evaluation

In addition to automatic evaluation, we also assessed system output by eliciting human judgments. We compared the quality of the summaries produced by a teacher model (UNILMv2$_{\text{BASE}}$)

---

[2]https://github.com/salesforce/factCC

CNN/Daily Mail	
GOLD	LZ Granderson: millennials say they'll marry if and when they want. He says that's not the case; they're happily single and happy. Granderson says marriage is about family, not money.
UNILMV2	LZ Granderson: millennials say they don't care what their generation thinks about marriage. He says they'll get married if and when they want. LZ: marriage is linked to economic well-being, but it's not clear if that's true.
+Noisy SKD	Carol Costello: talk to any millennial and you can envision an America virtually marriage-free. In countries like Sweden or Denmark, people don't feel pressured to marry even if they have kids together.

XSum	
GOLD	More than half of pupils in Wales have passed their GCSE exam for the third year running.
UNILMV2	More than 66.6% of pupils in Wales have achieved the top grades in their GCSE exams.
+Noisy SKD	Two thirds of Welsh pupils who took GCSEs got A* to C grades, according to this year's results.

WikiCatSum (Animal)	
GOLD	The Conception Bank silver boa (Chilabothrus Argentum) is a species of boa described in May 2016. It is only known from the conception island bank in the Bahamas. It is the first known discovery of a West Indian boa species in 73 years. It is named for its unique silver color.
UNILMV2	The Conception Bank silver boa (Chilabothrus Argentum) is a species of snake in the family Boidae. It is endemic to the Bahamas. The species was discovered on Conception Island Bank, which comprises uninhabited islets.
+Noisy SKD	The Conception Bank silver boa (Chilabothrus Argentum) is a species of nonvenomous boa endemic to the Bahamas. It was discovered in 2016 on Conception Island Bank, an uninhabited islet in the Bahamas.

Table 5: GOLD reference summaries and automatic summaries produced by UNILMV2$_{BASE}$ and its distilled student on the CNN/DailyMail, XSum, and WikiCatSum datasets.

against its distilled student (+Noisy SKD). For CNN/DailyMail and XSum, human participants were presented with the output of two systems (and the original document) and asked to decide which one was better according to the following criteria: *Succinctness* (Does the summary avoid repetition?), *Informativeness* (Does the summary capture the document's most important information?), and *Fluency* (Is the summary fluent and grammatical?). Evaluation was conducted on the Amazon Mechanical Turk crowdsourcing platform. We used the same test documents (20 in total) from Liu and Lapata (2019) for both CNN/DailyMail and XSum. We elicited five responses per HIT. Systems were rated along each dimension, and assigned a score corresponding to the proportion of times a system was selected as better against another.

Human evaluation results are shown in Table 4 (upper part). On both CNN/DailyMail and XSum datasets participants perceive the student (+Noisy SKD) as significantly ($p < 0.05$) more succinct and informative compared to the teacher (UNILMV2$_{BASE}$). However, on Fluency, the student tends to be worse. Upon inspection we found student summaries to be rather telegraphic, and hypothesize that crowdworkers tend to penalize them in terms of fluency, even though they are grammatical.

Human evaluation was performed slightly different for WikiCatSum. Recall that this is a multi-document dataset, where input documents are discontinuous webpage fragments. To allow participants to perform the experiment in a timely fashion, we used the gold summary as a proxy for the content of the input. Crowdworkers were presented with the output of two systems (again UNILMV2$_{BASE}$ and +Noisy SKD) and asked to decide which one was better according to the in-

formation contained in the gold summary. Evaluation was conducted on AMT, we randomly selected 20 samples from the test set and elicited three responses per HIT. For each domain, we report the proportion of times a system was chosen as better.

Human evaluation results are shown in Table 4 (lower part). AMT Crowdworkers prefer the summaries produced by the student for the Animal and Film domains, but not for Company; we found that the distilled model tends to generate too many entities in one sentence which render the summaries too dense for this domain.

6 Conclusions

In this paper we advocated the use of self-knowledge distillation for abstractive summarization, as a means to alleviate problems associated with maximum-likelihood training for this task. We also introduced several noise functions (in the training signal and training data) which help regularize training and further boost performance. Experiments on three benchmark datasets demonstrate that our framework can improve both non-pretrained and pretrained summarizers. In the future we would like to investigate more thoroughly which aspects of pretrained models improve and how self-knowledge distillation can be enhanced with more sophisticated noise functions.

References

Sungsoo Ahn, Shell Xu Hu, Andreas C. Damianou, Neil D. Lawrence, and Zhenwen Dai. 2019. Variational information distillation for knowledge transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9155–9163, Long Beach, California.

Jimmy Ba and Rich Caruana. 2014. Do deep nets really need to be deep? In *Advances in Neural Information Processing Systems 27*, pages 2654–2662. Curran Associates, Inc.

Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Xiaodong Liu, Yu Wang, Songhao Piao, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2020. UniLMv2: Pseudo-masked language models for unified language model pre-training. *CoRR*, abs/2002.12804.

Cristian Bucilu, Rich Caruana, and Alexandru Niculescu-Mizil. 2006. Model compression. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, page 535–541, New York, NY, USA. Association for Computing Machinery.

Samuel Rota Bulò, Lorenzo Porzi, and Peter Kontschieder. 2016. Dropout distillation. In *Proceedings of the International Conference on Machine Learning*, pages 99–107, New York, New York.

Asli Celikyilmaz, Antoine Bosselut, Xiaodong He, and Yejin Choi. 2018. Deep communicating agents for abstractive summarization. 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 1662–1675, New Orleans, Louisiana.

Kevin Clark, Minh-Thang Luong, Urvashi Khandelwal, Christopher D. Manning, and Quoc V. Le. 2019. BAM! born-again multi-task networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5931–5937, Florence, Italy. Association for Computational Linguistics.

Wojciech M. Czarnecki, Simon Osindero, Max Jaderberg, Grzegorz Swirszcz, and Razvan Pascanu. 2017. Sobolev training for neural networks. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4278–4287. Curran Associates, Inc.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Bin Dong, Jikai Hou, Yiping Lu, and Zhihua Zhang. 2019. Distillation $\approx$ early stopping? harvesting dark knowledge utilizing anisotropic information retrieval. In *NeurIPS 2019 Workshop on Machine Learning with Guarantees*, Vancouver, Canada.

Maha Elbayad, Laurent Besacier, and Jakob Verbeek. 2018. Token-level and sequence-level loss smoothing for RNN language models. *CoRR*, abs/1805.05062.

Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. 2018. Born-again neural networks. In *Proceedings of the 35th International Conference on Machine Learning*, pages 1602–1611, Stockholm, Sweden.

Sebastian Gehrmann, Yuntian Deng, and Alexander Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4098–4109, Brussels, Belgium.

Akhilesh Gotmare, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2019. A closer look at deep learning heuristics: Learning rate restarts, warmup and distillation. In *Proceedings of the 7th International Conference on Learning Representations (ICLR), New Orleans, Louisiana*. OpenReview.net.

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. 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 708–719, New Orleans, Louisiana. Association for Computational Linguistics.

Sangchul Hahn and Heeyoul Choi. 2019. Self-knowledge distillation in natural language processing. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 423–430, Varna, Bulgaria. INCOMA Ltd.

Donna Harman and Paul Over. 2004. The effects of human variation in DUC summarization evaluation. In *Text Summarization Branches Out*, pages 10–17, Barcelona, Spain. Association for Computational Linguistics.

Junxian He, Jiatao Gu, Jiajun Shen, and Marc'Aurelio Ranzato. 2019. Revisiting self-training for neural sequence generation. *CoRR*, abs/1909.13788.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 1693–1701. Curran Associates, Inc.

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

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

Wojciech Kryściński, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Neural text summarization: A critical evaluation. *CoRR*, abs/1908.08960.

Adhiguna Kuncoro, Miguel Ballesteros, Lingpeng Kong, Chris Dyer, and Noah A. Smith. 2016. Distilling an ensemble of greedy dependency parsers into

one MST parser. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1744–1753, Austin, Texas. Association for Computational Linguistics.

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

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Peter J Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. 2018. Generating Wikipedia by summarizing long sequences. In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada.

Weijie Liu, Peng Zhou, Zhiruo Wang, Zhe Zhao, Haotang Deng, and Qi Ju. 2020. FastBERT: a self-distilling BERT with adaptive inference time. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6035–6044, Online. Association for Computational Linguistics.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496, Florence, Italy. Association for Computational Linguistics.

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)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland. Association for Computational Linguistics.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online. Association for Computational Linguistics.

Seyed-Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, and Hassan Ghasemzadeh. 2019. Improved knowledge distillation via teacher assistant: Bridging the gap between student and teacher. *CoRR*, abs/1902.03393.

Hossein Mobahi, Mehrdad Farajtabar, and Peter L. Bartlett Bartlett. 2020. Self-distillation amplifies regularization in Hilbert space. *CoRR*, abs/2002.05715.

Lili Mou, Ran Jia, Yan Xu, Ge Li, Lu Zhang, and Zhi Jin. 2016. Distilling word embeddings: An encoding approach. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 1977–1980, Indianapolis, Indiana.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Caglar Gulcehre, 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*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

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*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

Ani Nenkova. 2006. Summarization evaluation for text and speech: issues and approaches. In *Proceedings of the 9th International Conference on Spoken Language Processing*, Pittsburgh, Pennsylvania.

Emilio Parisotto, Jimmy Ba, and Ruslan Salakhutdinov. 2015. Actor-mimic: Deep multitask and transfer reinforcement learning. *CoRR*, abs/1511.06342.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada.

Laura Perez-Beltrachini, Yang Liu, and Mirella Lapata. 2019. Generating summaries with topic templates and structured convolutional decoders. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5107–5116, Florence, Italy. Association for Computational Linguistics.

Mary Phuong and Christoph Lampert. 2019. Towards understanding knowledge distillation. In *Proceedings of the 36th International Conference on Machine Learning*, pages 5142–5151, Long Beach, California.

Weizhen Qi, Yu Yan, Yeyun Gong, Dayiheng Liu, Nan Duan, Jiusheng Chen, Ruofei Zhang, and Ming Zhou. 2020. ProphetNet: Predicting future n-gram for sequence-to-SequencePre-training. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2401–2410, Online. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683.

GJ Rath, A Resnick, and TR Savage. 1961. The formation of abstracts by the selection of sentences. part i. sentence selection by men and machines. *American Documentation*, 12(2):139–141.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. 2014. FitNets: Hints for thin deep nets. *CoRR*, abs/1412.6550.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Evan Sandhaus. 2008. The New York Times Annotated Corpus. *Linguistic Data Consortium, Philadelphia*, 6(12).

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.

Kaiqiang Song, Lin Zhao, and Fei Liu. 2018. Structure-infused copy mechanisms for abstractive summarization. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1717–1729, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. MASS: Masked sequence to sequence pre-training for language generation. In *Proceedings of the 36th International Conference on Machine Learning*, pages 5926–5936, Long Beach, California.

Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. 2017. Abstractive document summarization with a graph-based attentional neural model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1171–1181, Vancouver, Canada. Association for Computational Linguistics.

Xu Tan, Yi Ren, Di He, Tao Qin, Zhou Zhao, and Tie-Yan Liu. 2019. Multilingual neural machine translation with knowledge distillation. *CoRR*, abs/1902.10461.

Yee Teh, Victor Bapst, Wojciech M. Czarnecki, John Quan, James Kirkpatrick, Raia Hadsell, Nicolas Heess, and Razvan Pascanu. 2017. Distral: Robust multitask reinforcement learning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4496–4506. Curran Associates, Inc.

Gregor Urban, Krzysztof J Geras, Samira Ebrahimi Kahou, Ozlem Aslan, Shengjie Wang, Rich Caruana, Abdelrahman Mohamed, Matthai Philipose, and Matt Richardson. 2016. Do deep convolutional nets really need to be deep and convolutional? *CoRR*, abs/1603.05691.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144.

Qizhe Xie, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. 2019. Self-training with noisy student improves imagenet classification. *CoRR*, abs/1911.04252.

Chenglin Yang, Lingxi Xie, Siyuan Qiao, and Alan Yuille. 2019. Training deep neural networks in generations: A more tolerant teacher educates better students. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, volume 33, pages 5628–5635, Honolulu, Hawaii.

Sergey Zagoruyko and Nikos Komodakis. 2017. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *Proceedings of the 5th International Conference on Learning Represenatations*, Toulon, France.

Dongxu Zhang and Zhichao Yang. 2018. Word embedding perturbation for sentence classification. *CoRR*, abs/1804.08166.