UNISUMM and SUMMZOO: Unified Model and Diverse Benchmark for Few-Shot Summarization

Yulong Chen[1,2] * **Yang Liu**[3] † **Ruochen Xu**[3] **Ziyi Yang**[3]
Chenguang Zhu[3] **Michael Zeng**[3] **Yue Zhang**[2,4]
[1] Zhejiang University [2] Westlake University [3] Microsoft Research
[4] Westlake Institute for Advanced Study
{chenyulong, zhangyue}@westlake.edu.cn yaliu10@microsoft.com

Abstract

The high annotation costs and diverse demands of various summarization tasks motivate the development of few-shot summarization. However, despite the emergence of many summarization tasks and datasets, the current training paradigm for few-shot summarization systems ignores potentially shareable knowledge in heterogeneous datasets. To this end, we propose UNISUMM, a unified few-shot summarization model pre-trained with multiple summarization tasks and can be prefix-tuned to excel at any few-shot summarization task. Meanwhile, to better evaluate few-shot summarizers, under the principles of diversity and robustness, we assemble and release a new benchmark SUMMZOO. It consists of 8 summarization tasks with multiple sets of few-shot samples for each task, covering diverse domains. Experimental results and analysis show that UNISUMM outperforms strong baselines by a large margin across all sub-tasks in SUMMZOO under both automatic and human evaluations and achieves comparable results in human evaluation compared with a GPT-3.5 model.

1 Introduction

There has been a recent surge of interest in summarizers based on large pre-trained language models (PLMs) (Liu and Lapata, 2019; Yang et al., 2020; Zhong et al., 2020; Yu et al., 2022; Xu et al., 2022; Wang et al., 2023), where various summarization tasks (the term *task* later in this paper refers to a specific summarization task, e.g., query-focused meeting summarization, which is usually associated with a corresponding dataset, e.g., QMSum, unless otherwise specified.) have been proposed to meet different practical demands, such as comprehending different inputs (e.g., news (Fabbri et al., 2019) and dialogue (Zhong et al., 2022a)) and generating different outputs (e.g., headlines (Zhang and Tetreault, 2019) and paragraphs (Perez-Beltrachini and Lapata, 2021)). Because annotating gold summaries for newly-proposed summarization tasks is costly (Sen et al., 2008; Zhang et al., 2022), few-shot summarization, the task of building a model for a specific summarization scenario using very limited ground-truth data (Chen and Shuai, 2021), has gained increasing attention from the research community (Fabbri et al., 2021; Logan IV et al., 2022; Liu et al., 2022b; He et al., 2022).

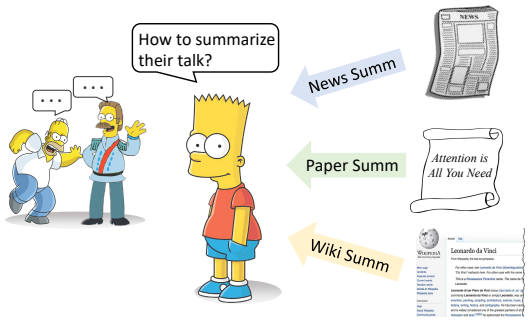


Figure 1: The few-shot summarization scenario in this paper. We are interested in how to re-use previous datasets (e.g., CNNDM) to improve the few-shot performance on unseen target tasks (e.g., DIALOGSUM).

Recently, prefix-tuning (Li and Liang, 2021) has established strong baselines on many few-shot natural language generation tasks, including summarization. The main idea is to extract knowledge from PLMs by prepending and tuning additional parameters (prefixes) before each layer of the PLM. Work has been done to improve the performance by designing more sophisticated prefixes (Ghazvininejad et al., 2022; Liu et al., 2022b). Despite being effective, PLMs can have limited summarization knowledge due to the salient gap between pre-training objectives (e.g., language modeling) and summarization objectives (Aribandi et al., 2022). In addition, existing summarization datasets can provide relevant knowledge to newly-proposed summarization tasks, and therefore benefit sum-

* Yulong Chen completed this work during his internship at Microsoft.

† Yang Liu is the corresponding author.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 12833–12855
July 9-14, 2023

marization tasks, especially under the few-shot scenario. However, existing work tends to tune PLMs directly on a new task, without exploiting cross-task knowledge from summarization datasets, which may limit the generalization and adaptation abilities of models (Zhong et al., 2019; Chen and Yang, 2021; Fang et al., 2022).

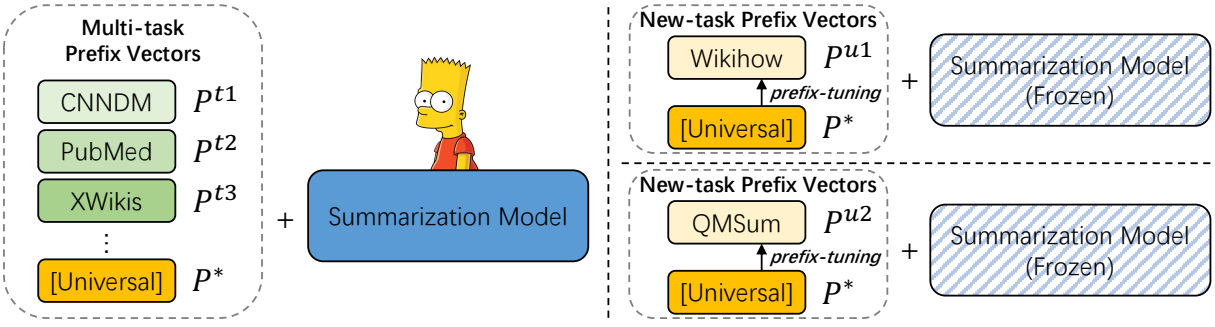
We address these issues by proposing a unified few-shot summarization framework, **UNISUMM**. The idea is to combine multi-task pre-training (Chen and Shuai, 2021) on existing summarization datasets with few-shot prefix-tuning (Li and Liang, 2021) on target tasks. To this end, we first build a multi-task model based on a Transformer-based language model as the backbone and equip it with task-specific prefix vectors, and then pre-train the multi-task model on diverse summarization datasets. In this stage, we optimize the summarization model together with task-specific prefixes and also a *universal prefix*, using an *asymmetrical weight decay* strategy. Using prefixes in the multi-task pre-training stage leads to two advantages: First, the mixture of shared summarization parameters and unique task-specific parameters helps to leverage natural benefits across datasets (Ruder, 2017). Second, the pre-trained prefixes can be tuned to serve as a knob for the second stage of prefix-tuning on unseen tasks. When facing an unseen few-shot summarization task, we freeze the multi-task learned backbone model and use the universal prefix as initialization for prefix-tuning.

A data obstacle for few-shot summarization research is the lack of a benchmark for fair comparison. Previous studies either focus on one type of data, e.g., news text (Liu et al., 2022b), or train their systems on non-public few-shot samples. However, because few-shot models can be highly sensitive to training data, the selection of different few-shot samples in different papers can lead to ambiguous comparisons (a.k.a. *Sample Selection Bias* (Cortes et al., 2008)). To address these issues, we assemble and release a new few-shot summarization benchmark, **SUMMZOO**, following two principles, namely *diversity of tasks* and *robustness of evaluation*. SUMMZOO collects summarization data from 8 existing datasets, which are diverse in terms of domain (news, academic papers, meetings, etc.), format (single-document and multi-document), and length on both source and target sides. For more robust evaluation, for each task, SUMMZOO provides 5 different (randomly sampled) few-shot training sets, and requires all systems to report their averaged results. Finally, SUMMZOO includes 10-shot and 100-shot settings.

We compare UNISUMM against several strong baselines, including a GPT-3.5 model (`text-davinci-002`) (Brown et al., 2020; Ouyang et al., 2022), on SUMMZOO and conduct thorough analysis. Experimental results of automatic evaluation metrics show that UNISUMM outperforms baselines across all sub-stasks and human evaluation shows that UNISUMM achieves better performance than baselines of similar sizes and comparable performance compared with `text-davinci-002`. Additionally, UNISUMM is empirically found to be more stable and robust when facing different few-shot samples. Analysis shows that combining multi-task pre-training and few-shot prefix-tuning is essential to the performance of UNISUMM and other techniques, such as universal prefix and asymmetrical weight decay strategy, can all improve its generalization ability. We release our code, model and benchmark at `https://github.com/microsoft/UniSumm`.

2 Related Work

Few-shot Summarization A critical challenge for neural summarizers is that they are data-hungry and require large-scale annotated data. To alleviate the data sparsity issue, Fabbri et al. (2021) extract characteristics of the target dataset and build pseudo summaries from the Wikipedia corpus. Small plug-in networks (Bražinskas et al., 2020) are injected into PLMs to predict the properties of the target dataset with only a small amount of labeled instances. To close the gap between pre-training and fine-tuning, Yu et al. (2021) propose a second stage of pre-training before fine-tuning with large-scale generative models. Such challenges of summarization have also been explored in the cross-lingual setting (Wang et al., 2022; Chen et al., 2022b). Although transfer learning methods make use of external data, one still needs to carefully select source domains and tasks to avoid negative transfer (Gururangan et al., 2020; Pilault et al., 2020). Compared with them, UNISUMM can be easily prefix-tuned to any target tasks without the effort of building large pseudo data or selecting relevant data. To our knowledge, we are the first to combine prefix-tuning and multi-task learning for few-shot summarization, showing very positive



(a) Multi-task Pre-training (b) Few-shot Prefix-tuning

Figure 2: The two-phase framework of UNISUMM. (a) The multi-task pre-training phase. The summarization model's parameters and prefixes are tuned together on multiple pre-training tasks, for example, *CNNDM*, *PubMed*, etc. (b) The few-shot tuning phase. For a new task, e.g., *WikiHow*, we only tune prefix parameters while keeping the summarization model's parameters frozen.

results.

Existing few-shot summarization evaluation suffers from two data-related problems. First, previous studies usually focus on only one type of summarization tasks in their experiments (Bražinskas et al., 2020; Liu et al., 2022b). Thus, it is difficult to evaluate their generalization ability. Second, the few-shot settings and selections of few-shot samples are miscellaneous, which makes evaluations from different research papers not comparable with each other (Cortes et al., 2008). Therefore, in this work, we propose SUMMZOO for better benchmarking future research on few-shot summarization. To our knowledge, SUMMZOO is the first public few-shot summarization benchmark that covers a set of diverse summarization tasks.

Prompt Learning for Text Generation The idea of prompt learning is first proposed in GPT-3 (Brown et al., 2020), where it aims to guide PLMs to do different tasks without further fine-tuning by prepending task-related examples to the input and has shown positive results on many text generation tasks, including summarization (Goyal et al., 2022). Prefix-tuning extends this idea from discrete tokens to continuous vectors (Li and Liang, 2021). It adds continuous embeddings (prefixes) to each Transformer layer as external value and key vectors. During training, only prefixes are updated while the other parameters are unchanged. Logan IV et al. (2022) and Gu et al. (2022) propose to use pre-training to boost the low performance for few-shot learning. Li et al. (2022) combines the transfer learning and prompt learning for text generation. Compared with them, we are interested in few-shot summarization and propose multi-task pre-training as an effective strategy to make use of data from related tasks to improve performance of diverse target tasks, which suits real-life scenarios.

3 Method

Following Chen and Shuai (2021), the task of *few-shot text summarization* is defined as follows. For an unseen target summarization task u, few-shot text summarization is to generate a summary Y, given an input text X, by learning from a limited number k ($k \leq 100$ typically) of labeled training instances of u, with the help of general knowledge K.

The overall framework of UNISUMM is shown in Figure 2. It consists of 2 phases: 1) Learning general knowledge by multi-task pre-training on existing summarization datasets (§ 3.1) and; 2) Learning target task knowledge by prefix-tuning on each target few-shot summarization dataset (§ 3.2).

3.1 Multi-Task Pre-Training with Prefix

As shown in Figure 2 (a), in the first stage, we take a Transformer-based pre-trained language encoder-decoder model (for example, BART (Lewis et al., 2020)) $M = [M_{en}; M_{de}]$ as the summarization model, parameterized by θ. We further pre-train this model on a set of popular summarization datasets (e.g., *CNNDM*, *PubMed* and *XWikis*) to learn general summarization knowledge. For each task t, we inject task-specific prefix vectors of encoder (P^t_{en}) and decoder (P^t_{de}), $P^t = [P^t_{en}; P^t_{de}]$, into the model, parameterized by θ_{p^t}. Following (Li and Liang, 2021), the prefix vectors are prepended to each Transformer layer of M as additional key and value vectors as: $[P^t_{en}; M_{en}; P^t_{de}; M_{de}]$.

For all pre-training tasks, given input text X, the multi-task optimization objective is to minimize the negative log-likelihood of generating the target summary $Y = \{y_1, y_2, ...y_{|Y|}\}$:

$$L(\theta, \theta_{p^t}) = \sum_{i}^{|Y|} \log \mathbb{P}(y_i|X, y_1, \cdots, y_{i-1}). \quad (1)$$

In the multi-task pre-training stage, we optimize θ and θ_{p^t} together.

3.2 Prefix-Tuning

Through multi-task pre-training, we obtain the UNISUMM model with diverse summarization knowledge. As shown in Figure 2 (b), for an unseen summarization task u (for example, *WikiHow* or *MultiNews*), given only k training samples, we conduct prefix-tuning (Li and Liang, 2021) on the UNISUMM model. A new-task prefix $P^u = [P^u_{en}; P^u_{de}]$ is created, parameterized by θ_{p^u}, which can be either initialized randomly or from a prefix of pre-training tasks. We then freeze the parameters θ of the shared summarization model and only tune θ_{p^u} using the objective defined in Equation 1. By doing this, we can maximize the learned summarization knowledge in UNISUMM and also avoid over-fitting the model to very few samples.

3.3 Universal Prefix

Empirically, given a target task, initializing new-task prefix from the most related pre-training tasks can be helpful. However, for a brand new task, selecting meta tasks can be a complicated process, which requires large efforts of feature engineering (Chen and Shuai, 2021). Therefore, during multi-task pre-training, we also pre-train a universal prefix, which can be used as a stable initialization for few-shot prefix-tuning.

In particular, during multi-task pre-training (§ 3.1), we initialize a universal encoder and decoder prefix vector $P^* = [P^*_{en}; P^*_{de}]$, parameterized by θ_{p^*}. For each training instance from task t, it has a 15% probability to be coupled with this universal prefix vector instead of its task-specific prefix P^t. The parameters θ_{p^*} are optimized together with θ. Then in prefix-tuning, we use this universal vector as initialization for the unseen task parameter θ_{p^u} (§ 3.2).

3.4 Asymmetrical Weight Decay

A potential problem in multi-task learning is the negative transfer among different pre-training tasks. To alleviate this, inspired by previous work (Evgeniou and Pontil, 2004; Bengio, 2012; Liu et al., 2019), we set different weight decay regularizations on different parameters of UNISUMM. Specifically, we separate optimizers of the prefixes and the summarization model in pre-training. We assign a lower weight decay value d_p=0.01 on the prefix optimizer, enabling prefixes to flexibly learn task-specific knowledge, and a higher weight decay value d_l=0.05 on the summarization model optimizer, enforcing it to learn a broader generalization across different tasks.

Formally, at training step i:

$$\begin{aligned} \theta^{i+1} &= (1 - d_l)\theta^i - \alpha^i \nabla f^i(\theta^i), \\ \theta^{i+1}_p &= (1 - d_p)\theta^i_p - \alpha^i_p \nabla f^i_p(\theta^i_p), \end{aligned} \quad (2)$$

where α^i and α^i_p are the learning rates for summarization model and prefix, and $\nabla f^i(\theta^i)$ and $\nabla f^i_p(\theta^i_p)$ are the batch gradient for summarization model and prefix.

4 The SUMMZOO Benchmark

SUMMZOO is sourced from existing summarization benchmark based on the principles of diversity and robustness, where we assemble each dataset into few-shot evaluation settings.

Diversity of Tasks As a major goal, we ensure that SUMMZOO can include a diversity of different summarization tasks, covering multiple domains, text styles and compression ratios. Thus, we carefully select 8 summarization tasks including monologue/dialogue texts and single/multi-document summarization tasks. Their domains also span an assorted set such as news, scientific papers, instructions, online forums and meetings.

Robustness of Evaluation Our second goal is to ensure that experiments on SUMMZOO can be compared with each other in a robust manner. Also, we want to reduce the randomness from different selections of few-shot samples. Therefore, for each task, we provide 5 sets of few-shot training samples, and we ask all models to train on these 5 sets respectively and report their averaged results and standard deviations. We also formulate two few-shot training settings with the number of shots k set to 10 or 100, where the first can be considered as a more extreme low-resource scenario while the second is a more commonly tested setting.

Type		Domain	Dataset	Testset Size	Avg. D/S Length
Monologue	Multi-doc	News	MultiNews (Fabbri et al., 2019)	5,622	2,103/264
	Extreme single-doc		XSum (Narayan et al., 2018)	11,334	431/20
	Single-doc	Scientific Paper	ArXiv (Cohan et al., 2018)	6,440	4,938/220
	Single-doc	Instructions	WikiHow (Koupaee and Wang, 2018)	6,000	580/62
	Single-doc	Online Forum	Reddit-TIFU (Kim et al., 2019)	4,208	433/23
Dialogue	Single-doc	Online Chit-chat	SAMSum (Gliwa et al., 2019)	819	94/28
	Single-doc	Real-life	DIALOGSUM (Chen et al., 2021)	500	131/24
	Query-based single-doc	Meeting	QMSum (Zhong et al., 2021)	279	1,310/65

Table 1: Summary of sub-tasks in SUMMZOO. We report the sizes of test sets here. "Avg. D/S length" stands for "averaged document/summary token length". For QMSum, we concatenate the query and gold span as input.

Table 1 summarizes the statistics of sub-datasets in SUMMZOO. The detailed descriptions of each dataset can be found in Appendix A.

5 Experimental Setup

5.1 Training Datasets

For multi-task pre-training (§ 3.1), we use a combination of seven summarization datasets: CNNDM (Nallapati et al., 2016), BillSum (Kornilova and Eidelman, 2019), PubMed (Cohan et al., 2018), GovReport (Huang et al., 2021), MediaSum (Zhu et al., 2021), SummScreen (Chen et al., 2022a) and XWikis (Perez-Beltrachini and Lapata, 2021).

To balance the training data size of different datasets, we perform down-sampling on over-sized datasets and up-sampling on low-resource datasets respectively. The detailed descriptions of each dataset and statistics of resulting data for pre-training are shown in Appendix B and Table 8.

5.2 Baseline Models

PEGASUS (Zhang et al., 2020) is a large pre-trained encoder-decoder model, which is particularly designed for text summarization. The model is trained using the gap sentence generation task. We use $\text{PEGASUS}_{\text{LARGE}}$ (C4+HugeNews)[1] for comparison, which improves upon the results reported in the original paper.

BART (Lewis et al., 2020) is a pre-trained encoder-decoder language model using self-denoising tasks. We compare with the BART-large model[2] with two tuning strategies on few-shot summarization tasks, namely standard fine-tuning (**BART-FT**) and prefix-tuning (**BART-PT**). In BART-PT, the prefix vector is added in the same way as in UNISUMM.

MultiBART is a variant of BART-large. Similar to UNISUMM, it is first multi-task pre-trained on the ***same data*** (§ 5.1) but *without* prefixes. And it can also be fine-tuned or prefix-tuned to fit few-shot summarization tasks. We only show the results of prefix-tuned MultiBART because we find fine-tuning the entire MultiBART model always leads to worse performance in the few-shot setting. This strong baseline can be considered as an indicator to verify the effectiveness of using prefixes in both multi-task pre-training and few-shot tuning.

`Text-davinci-002` (Brown et al., 2020; Ouyang et al., 2022) is a large language model (175B) from the GPT-3.5 family,[3] using instruction tuning, and has shown great zero-/few-shot performance on many NLP tasks, including summarization. Specifically, recent work finds that GPT-3.5 models can show much better performance with the technique of in-context learning (ICL) (Brown et al., 2020; Liu et al., 2022a). We use `text-davinci-002` with ICL for experiments, and only show the performance of 1-shot ICL because of its input length limitation.[4]

All baseline models and UNISUMM are evaluated on SUMMZOO (Appendix C shows the implementation details). We conduct both automatic and human evaluation. As described, SUMMZOO requires models to report averaged results and their standard deviations over 5 sets of different few-shot samples (except for `text-davinci-002`). We use ROUGE (Lin, 2004) for automatic evalua-

[1] https://huggingface.co/google/pegasus-large
[2] https://huggingface.co/facebook/bart-large
[3] https://openai.com/
[4] For MultiNews and ArXiv, due to the length limitation of GPT-3.5 API, we only include the summary part in their ICL examples.

Task		PEGASUS			BART-FT			BART-PT			MultiBART			UNISUMM		
		R1	R2	RL	R1	R2	RL	R1	R2	RL	R1	R2	RL	R1	R2	RL
MN	10	39.12	11.15	19.44	38.29	10.05	18.32	38.27	11.38	19.28	42.31	14.55	21.53	**45.13**	**15.19**	**21.63**
	100	42.36	12.78	20.56	42.65	13.27	20.69	43.86	13.97	20.79	45.71	15.78	22.21	**45.91**	**15.86**	**22.24**
XSum	10	20.55	3.98	14.80	24.89	6.42	19.18	14.29	2.77	11.52	20.76	5.76	17.01	**26.10**	**7.20**	**19.92**
	100	**37.30**	**13.69**	**29.08**	27.45	7.21	21.74	29.70	9.87	23.70	31.48	10.88	25.00	**33.33**	**11.36**	**25.85**
ArXiv	10	34.81	8.46	29.12	28.40	4.98	25.15	29.85	8.08	26.76	41.45	14.68	37.01	**43.33**	**15.38**	**38.69**
	100	38.08	10.14	31.06	36.69	10.07	32.67	38.03	11.46	34.20	43.56	15.97	39.01	**44.33**	**16.42**	**39.71**
WH	10	27.74	7.80	19.61	17.09	2.37	12.01	25.31	7.45	19.02	27.64	7.99	19.91	**30.87**	**9.35**	**21.72**
	100	33.21	10.86	24.41	26.46	6.91	18.83	32.35	10.42	23.23	34.10	11.31	25.03	**34.90**	**11.73**	**25.70**
Reddit	10	18.90	3.89	14.27	13.80	1.20	10.48	19.01	4.07	14.46	21.44	5.17	16.22	**22.88**	**5.60**	**17.02**
	100	23.40	5.71	17.99	17.91	2.58	13.33	23.10	5.41	17.42	24.06	5.89	17.97	**24.54**	**6.17**	**18.30**
DS	10	36.44	10.89	28.49	28.62	5.97	22.83	33.46	10.08	27.90	37.05	12.61	30.24	**38.76**	**13.38**	**31.07**
	100	41.02	14.53	32.29	38.77	12.91	31.40	41.20	13.97	32.76	42.16	**15.71**	**33.79**	**42.43**	15.64	33.74
SS	10	38.58	13.79	30.37	18.07	4.23	14.70	35.53	12.96	28.26	39.69	16.28	32.11	**43.89**	**18.53**	**34.76**
	100	44.60	18.40	35.16	37.36	14.14	30.02	43.39	17.82	34.42	45.47	19.68	36.60	**46.93**	**20.65**	**37.28**
QM	10	31.77	9.70	21.48	23.64	3.56	14.88	27.58	8.39	19.41	33.71	10.59	22.27	**36.00**	**12.12**	**23.56**
	100	35.54	11.68	23.74	33.96	10.30	22.10	35.07	11.66	23.10	37.67	13.38	24.68	**38.38**	**13.89**	**25.36**
Average	10	30.99	8.71	22.20	24.10	4.85	17.19	27.91	8.15	20.83	33.01	10.95	24.54	**35.87**	**12.09**	**26.05**
	100	36.94	12.22	26.79	32.66	9.67	23.85	35.84	11.82	26.20	38.03	13.58	28.04	**38.84**	**13.97**	**28.52**

Table 2: Main results of PEGASUS, BART-FT, BART-PT, MultiBART and UNISUMM on the SUMMZOO benchmark. MN, WH, DS, SS and QM are abbreviations of MultiNews, WikiHow, DIALOGSUM, SAMSum and QMSum. Best results on each sub-dataset are in bold. All models are trained on the same 5 sets of few-shot samples and we report their averaged ROUGE scores. The bottom block presents the averaged results of all 8 sub-tasks in SUMMZOO.

Task	GPT-3.5	10-UNI	100-UNI
MultiNews	11.01	15.19	15.86
Xsum	8.87	7.20	11.36
Arxiv	10.83	15.38	16.42
WikiHow	8.56	9.35	11.73
Reddit	6.03	5.60	6.17
DIALOGSUM	13.08	13.38	15.64
SAMSum	17.65	18.53	20.65
QMSum	11.62	12.12	13.89
Average	10.96	12.09	13.97

Table 3: R2 scores of 1-shot `text-davinci-002` (GPT-3.5) using ICL compared with 10-shot UNISUMM and 100-shot UNISUMM.

tion[5], which evaluates the n-gram overlap in the model-generated summary against the reference summary. We report the F-1 scores of ROUGE-1 (R1), ROUGE-2 (R2) and ROUGE-L (RL).

6 Automatic Evaluation

6.1 Main Results

The main results are shown in Table 2 and 3. First, compared with PEGASUS, UNISUMM outperforms it across all tasks except 100-shot XSum, and shows the best averaged scores in both 10-shot and 100-shot settings. We also find that 10-shot UNISUMM can outperform 100-shot PEGASUS on MultiNews, Arxiv and QMSum by a large margin, suggesting that UNISUMM can benefit from diverse training data and effectively adapt indirect knowledge to unseen tasks. It is notable that although the foundation BART model is inferior to PEGASUS, the BART-based UNISUMM can still outperform PEGASUS with the learned summarization knowledge. Overall, UNISUMM surpasses both BART-FT and BART-PT by a large margin on all tasks in all settings, which suggests the equipment of multi-task learning can substantially improve model performance on few-shot summarization tasks, in particular in the 10-shot setting.

UNISUMM also outperforms MultiBART by a large margin, especially in the 10-shot setting (Avg. 2.86 R1 improvements). Considering that MultiBART is multi-task pre-trained on the exact same data as UNISUMM does, the main difference from UNISUMM is whether to use prefixes in both multi-task pre-training and few-shot tuning. The result verifies the effectiveness of UNISUMM framework, in particular the prefix addition in the multi-task pre-training phrase (§ 3.1).

The comparison between `text-davinci-002` and UNISUMM is shown in Table 3. Generally, 100-shot UNISUMM achieves higher ROUGE scores than 1-shot `text-davinci-002` on all tasks and overall performance and 10-shot UNISUMM shows better performance compared with 1-shot `text-davinci-002` except for XSum and Reddit.

[5]We use the files2rouge for evaluation.

Task		PEG	B-PT	Mul	UNI
MultiNews	10	0.37	1.04	0.68	**0.33**
	100	0.20	**0.11**	0.26	0.19
XSum	10	1.45	1.60	1.65	**1.21**
	100	0.37	0.27	**0.11**	0.27
Arxiv	10	0.57	1.08	**0.32**	0.93
	100	0.55	0.83	0.64	**0.54**
WikiHow	10	0.79	0.66	0.66	**0.40**
	100	0.46	0.25	0.38	**0.21**
Reddit	10	**0.83**	1.61	1.20	1.16
	100	0.71	0.72	0.68	**0.52**
DIALOGSUM	10	1.18	**0.96**	1.46	0.99
	100	**0.83**	0.90	1.01	0.91
SAMSum	10	1.61	1.58	1.91	**1.07**
	100	0.47	**0.29**	0.40	0.47
QMSum	10	0.84	0.75	0.71	**0.45**
	100	0.72	0.55	0.34	**0.30**
Average	10	0.96	1.16	1.07	**0.82**
	100	0.54	0.49	0.48	**0.43**

Table 4: Comparison of model robustness towards different few-shot samples. We report the *standard deviations* of R1 scores on 5 sets of few-shot samples provided in SUMMZOO. Lower standard deviation indicates the model is more robust towards different few-shot samples. The bottom block shows the averaged results of all 8 sub-tasks. Models are PEGASUS (PEG), BART-PT (B-PT), MultiBART (Mul) and UNISUMM (UNI).

Task		Gold	GPT-3.5	PEG	B-PT	UNI
QM	*Flu.*	4.80	4.93	4.46	4.40	4.90
	Coh.	4.93	4.80	4.10	3.87	4.50
	Con.	5.00	4.03	3.33	3.13	3.80
	Rel.	4.90	4.17	3.27	2.80	3.97
WH	*Flu.*	4.72	4.90	4.43	4.30	4.68
	Coh.	4.57	4.83	4.17	4.00	4.43
	Con.	4.87	4.63	4.17	3.93	4.67
	Rel.	4.88	4.58	4.33	4.17	4.67
MN	*Flu.*	4.70	4.97	4.23	4.17	4.63
	Coh.	4.70	4.73	3.95	3.80	4.17
	Con.	4.93	3.07	3.53	3.27	4.07
	Rel.	4.77	2.73	3.72	3.63	4.30

Table 5: Human evaluation for gold summaries, 1-shot `text-davinci-002` (GPT-3.5), 100-shot PEGASUS (PEG), BART-PT (B-PT) and UNISUMM (UNI) on QMSum, WikiHow and MultiNews. *Flu.*, *Coh.*, *Con.* and *Rel.* stand for *Fluency*, *Coherence*, *Consistency* and *Relevance*, respectively.

Such improvements can be attributed to the fact that UNISUMM is few-shot trained on more samples. It is also worth noting that UNISUMM is based on BART-large (400M), while GPT-3.5 is orders of magnitude larger (175B). Also, we note that 10-shot UNISUMM can achieve higher ROUGE scores on some tasks such as MultiNews and Arxiv compared with `text-davinci-002`. Besides UNISUMM is multi-task trained on relevant data, one possible reason is that `text-davinci-002` is only presented with 1-shot summary as ICL context, due to the length limitation. However, given the previous finding (Goyal et al., 2022) that GPT-3.5 generated summaries can be favored by human evaluators with even lower ROUGE scores, we also conduct human evaluation in § 7.

6.2 Model Robustness

The sample selection bias (Cortes et al., 2008) has been a major problem for few-shot tasks, where model performance is strongly correlated with the selection of few-shot samples. And a sound system should be robust and stable when taking different few-shot samples. To demonstrate the robustness and stability of different few-shot summarization models, we report their standard deviations of ROUGE-1 scores on 5 different sets of few-shot samples provided in SUMMZOO in Table 4.

Overall, the standard deviations of UNISUMM are lower than all other baselines on most tasks in both settings, suggesting that UNISUMM is most stable and robust when facing different few-shot samples. Also, MultiBART outperforms BART-PT and shows better averaged results than PEGASUS in the 100-shot, showing that reusing related summarization datasets is valuable. However, it can still be unstable in the 10-shot setting. In contrast, UNISUMM shows the least averaged standard deviations across all tasks in both settings. This suggests that the two-phase training with prefixes in the UNISUMM framework is essential for enhancing the model robustness.

We present the full table, including standard deviations of R2 and RL scores, in Appendix D. Overall, we find that UNISUMM is most robust and stable towards different training samples.

7 Human Evaluation

To better understand the outputs of different few-shot summarization systems, following Kryscinski et al. (2019, 2020), we conduct a human evaluation from four dimensions: *Fluency*, *Consistency*, *Coherence* and *Relevance*. We select 30 samples from QMSum, WikiHow and MultiNews, respectively, covering both monologue and dialogue texts. Then, for each sample, we ask a judge with experience in human evaluation for summarization tasks, to give scores from 1 to 5 (higher score indicates better quality) along each evaluation dimen-

Model	MN		XSum		Arxiv		WH		Reddit		DS		SS		QM		Avg.	
	10	100	10	100	10	100	10	100	10	100	10	100	10	100	10	100	10	100
3-Task	**15.3**	15.8	4.8	10.9	15.0	15.7	9.2	**11.9**	**5.7**	6.1	12.6	15.6	17.1	19.8	11.5	13.5	11.4	13.6
7-Task	15.2	**15.9**	**7.2**	**11.4**	**15.4**	**16.4**	**9.4**	11.7	5.6	**6.2**	**13.4**	**15.7**	**18.5**	**20.7**	**12.1**	**13.9**	**12.1**	**14.0**

Table 6: ROUGE-2 results of UNISUMM models which are multi-task pre-trained on different scale of pre-training tasks. We show the best results in **bold**.

Prefix	MN		XSum		Arxiv		WH		Reddit		DS		SS		QM		Avg.	
	10	100	10	100	10	100	10	100	10	100	10	100	10	100	10	100	10	100
Random	**15.6**	**16.0**	4.4	11.1	**16.2**	16.3	**9.4**	11.6	**6.0**	6.1	13.3	**15.7**	18.1	**21.0**	11.9	13.7	11.9	13.9
CNNDM	15.1	15.8	6.3	11.1	14.8	15.8	9.4	11.7	5.6	6.1	13.1	15.5	**18.7**	20.7	11.9	13.7	11.9	13.8
Universal	15.2	15.9	**7.2**	**11.4**	15.4	**16.4**	9.4	**11.7**	5.6	**6.2**	**13.4**	15.6	18.5	20.7	**12.1**	**13.9**	**12.1**	**14.0**

Table 7: ROUGE-2 results of UNISUMM using different prefix initialization strategies. We show the best results in **bold**.

sion. Candidate outputs are from gold summaries, 1-shot `text-davinci-002`, 100-shot PEGASUS, BART-PT and UNISUMM respectively. In total, we have 450 summaries to evaluate and the results are reported in Table 5. Appendix E gives detailed description of evaluation dimensions.

In human evaluation, UNISUMM outperforms PEGASUS and BART-PT on all datasets regarding all dimensions, achieving a higher fluency score than gold summaries on QMSum and a comparable score on MultiNews and WikiHow, suggesting that UNISUMM can generate very fluent sentences which can be comparable with human annotated summaries. A challenge of QMSum is that models are asked to generate summaries focusing on the input queries. Thus, *Relevance* is a very important metric for this task. However, *Relevance* sees very low score for PEGASUS (3.27) and BART-PT (2.80), suggesting they are weak in extracting relevant information based on user queries. In contrast, UNISUMM achieves a higher score (3.97). `Text-davinci-002` also performs very well on this task, even outperforming the gold summaries on *Fluency*, but UNISUMM still achieves comparable results with limited training samples and much lower cost.

On MultiNews, since `text-davinci-002` is only input with 1-shot summary as ICL example due to length limitation, although it can generate very fluent (4.97) and coherent (4.73) summaries, it is less preferred by human annotators w.r.t. *Consistency* and *Relevance*. UNISUMM still outperforms other systems and only loses to gold summaries on this two metrics. Similar results are also observed on WikiHow, where `text-davinci-002` tends to generate very long summaries, which can contain some hallucination and less important content, and UNISUMM shows comparable performance on *Consistency* and *Relevance*.

We show case studies and their analysis, including an error case where UNISUMM fails, in Appendix F.

8 Analysis

8.1 Task Scale in Multi-task Training

One common concern about multi-task training is that: when multiple tasks are combined, will newly added tasks hurt or help the performance? To verify this, we add one variant of UNISUMM for comparison, whose phase-1 is multi-task pre-trained on 3 tasks instead of all 7 tasks in Table 8. For the 3 tasks, we use the combination of CNNDM, PubMed and MediaSum, which are typical datasets for news summarization (MultiNews and Xsum), academic paper summarization (ArXiv) and dialogue summarization (DIALOGSUM, SAMSum and QMSum).

Results in Table 6 show that when extending the multi-task pre-training datasets from 3 to 7, UNISUMM achieves better results on multiple datasets. For example, taking ArXiv as the target task, 7-Task UNISUMM outperforms 3-Task UNISUMM in both 10 and 100-shot settings. It suggests that 7-Task UNISUMM can benefit from GovReport, XWikis, SummScreen and BillSum for scientific text summarization. On average, the R2 score improves by 0.4 for the 10-shot setting and 0.7 for the 100-shot setting. This shows that negative transfer is minor in UNISUMM and suggests that by training UNISUMM on even more datasets, its generalization can potentially be improved by learning more indirect summarization knowledge.

8.2 Different Prefix Initializations

UNISUMM is equipped with a universal prefix that was randomly (15%) picked by all tasks during multi-task pre-training (§ 3.3). In Table 7, we show the ablation study of using different prefix initialization strategies in few-shot prefix-tuning. Due to space limitation, we show R-2 scores here. We compare three strategies: initialized the prefix randomly, using *CNNDM* prefix or using universal prefix. The *CNNDM* prefix is selected to be compared here because it is considered as a general summarization task and has been proved helpful to many tasks, e.g., SAMSum (Gliwa et al., 2019).

We see that using universal prefix yields the best results on most tasks. Also, the universal prefix is particularly useful for the 10-shot setting, bringing a 0.23 improvement for R2 score. In addition, we find that using task-specific prefix (*CNNDM*) shows the worst performance on some tasks, such as QMSum and ArXiv, and has the lowest average score. This can be explained by that the task-specific prefix (*CNNDM*) stores abundant task specific knowledge, which however can be harmful to unseen target tasks, especially when the target task is very different from the pre-training task.

We show more analysis in Appendix G.

9 Conclusion

We introduced UNISUMM, a novel few-shot summarization system that can be easily prefix-tuned to excel at and generalize on a diversity of summarization tasks. We propose to combine multi-task learning and prefix-tuning by jointly training the prefixes and the summarizer on multiple existing summarization datasets. By only tuning the prefix parameters, UNISUMM shows superior performance over strong baseline systems, yielding fluent and faithful summaries across tasks. In addition, we assembled and released a new benchmark, SUMMZOO, for fairly and effectively evaluating few-shot summarization models. It covers an assorted set of summarization tasks and provides multiple few-shot sets for a more robust and fairer comparison.

Limitations

The limitation of UNISUMM can be stated from three perspectives. First, the multi-task pre-training of UNISUMM can be time and cost consuming, which requires large GPU resources. Second, the current framework uses prefixes of a fixed length for both multi-task training and few-shot prefix-tuning. However, different summarization task may prefer different size of prefixes. Third, in this work, we focus on summarization tasks in English. The performance of UNISUMM for languages that have a different morphology or syntactic structures from English needs further exploration.

Ethics Statement

Copyright and Citation Issue The copyright of individual datasets in SUMMZOO belongs to the original authors. The usage license of each dataset also applies to SUMMZOO. To ensure fair credit, when using SUMMZOO for evaluation, please also cite original papers, where individual datasets are introduced.

Data Availability and Safety Pre-training and fine-tuning summarization data studied in this paper are mostly publicly available, otherwise we will provide links to the access application. Although filtering has been conducted in building the original datasets, some contents can contain uncomfortable descriptions, e.g., news coverage of violent crimes and events.

Usage of Large PLM The GPT-3.5 model is used to generate text (summaries) for input documents of summarization tasks. The generated text is only used for experiments and analysis, which are presented in corresponding sections. No further usage, e.g., generating content for manuscripts, of GPT-3.5 or its family, is included in this paper.

Human Evaluation We conduct human evaluation with the help of one judge, who obtained their postgraduate degree in the United Kingdom and has a solid experience in evaluating summarization tasks. They were compensated through a payment of around 400 USD for 450 instances (§ 7).

Acknowledgements

We appreciate all reviewers and chairs from ACL 2023 for their valuable suggestions. We thank Dan Iter, Hiteshi Sharma, Zicheng Liu, Sen Yang and Leyang Cui for their proofreading and inspiring discussion. This publication has emanated from research conducted with the financial support of the Pioneer and "Leading Goose" R&D Program of Zhejiang under Grant Number 2022SDXHDX0003.

References

Vamsi Aribandi, Yi Tay, Tal Schuster, Jinfeng Rao, Huaixiu Steven Zheng, Sanket Vaibhav Mehta, Honglei Zhuang, Vinh Q. Tran, Dara Bahri, Jianmo Ni, Jai Prakash Gupta, Kai Hui, Sebastian Ruder, and Donald Metzler. 2022. Ext5: Towards extreme multi-task scaling for transfer learning. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Yoshua Bengio. 2012. Practical recommendations for gradient-based training of deep architectures. In *Neural networks: Tricks of the trade*, pages 437–478. Springer.

Arthur Bražinskas, Mirella Lapata, and Ivan Titov. 2020. Few-shot learning for opinion summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4119–4135, Online. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Jiaao Chen and Diyi Yang. 2021. Structure-aware abstractive conversation summarization via discourse and action graphs. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1380–1391, Online. Association for Computational Linguistics.

Mingda Chen, Zewei Chu, Sam Wiseman, and Kevin Gimpel. 2022a. SummScreen: A dataset for abstractive screenplay summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8602–8615, Dublin, Ireland. Association for Computational Linguistics.

Yi-Syuan Chen and Hong-Han Shuai. 2021. Meta-transfer learning for low-resource abstractive summarization. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 12692–12700. AAAI Press.

Yulong Chen, Yang Liu, Liang Chen, and Yue Zhang. 2021. DialogSum: A real-life scenario dialogue summarization dataset. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 5062–5074, Online. Association for Computational Linguistics.

Yulong Chen, Ming Zhong, Xuefeng Bai, Naihao Deng, Jing Li, Xianchao Zhu, and Yue Zhang. 2022b. The cross-lingual conversation summarization challenge. In *Proceedings of the 15th International Conference on Natural Language Generation: Generation Challenges*, pages 12–18, Waterville, Maine, USA and virtual meeting. Association for Computational Linguistics.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621, New Orleans, Louisiana. Association for Computational Linguistics.

Corinna Cortes, Mehryar Mohri, Michael Riley, and Afshin Rostamizadeh. 2008. Sample selection bias correction theory. In *International conference on algorithmic learning theory*, pages 38–53. Springer.

Theodoros Evgeniou and Massimiliano Pontil. 2004. Regularized multi–task learning. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 109–117.

Alexander Fabbri, Simeng Han, Haoyuan Li, Haoran Li, Marjan Ghazvininejad, Shafiq Joty, Dragomir Radev, and Yashar Mehdad. 2021. Improving zero and few-shot abstractive summarization with intermediate fine-tuning and data augmentation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 704–717, Online. Association for Computational Linguistics.

Alexander Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir Radev. 2019. Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1074–1084, Florence, Italy. Association for Computational Linguistics.

Yue Fang, Hainan Zhang, Hongshen Chen, Zhuoye Ding, Bo Long, Yanyan Lan, and Yanquan Zhou. 2022. From spoken dialogue to formal summary: An utterance rewriting for dialogue summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3859–3869, Seattle, United States. Association for Computational Linguistics.

Marjan Ghazvininejad, Vladimir Karpukhin, Vera Gor, and Asli Celikyilmaz. 2022. Discourse-aware soft prompting for text generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4570–4589, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. SAMSum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 70–79, Hong Kong, China. Association for Computational Linguistics.

Tanya Goyal, Junyi Jessy Li, and Greg Durrett. 2022. News summarization and evaluation in the era of gpt-3. *arXiv preprint arXiv:2209.12356*.

Yuxian Gu, Xu Han, Zhiyuan Liu, and Minlie Huang. 2022. PPT: Pre-trained prompt tuning for few-shot learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8410–8423, Dublin, Ireland. Association for Computational Linguistics.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Pengcheng He, Baolin Peng, Liyang Lu, Song Wang, Jie Mei, Yang Liu, Ruochen Xu, Hany Hassan Awadalla, Yu Shi, Chenguang Zhu, et al. 2022. Z-code++: A pre-trained language model optimized for abstractive summarization. *arXiv preprint arXiv:2208.09770*.

Luyang Huang, Shuyang Cao, Nikolaus Parulian, Heng Ji, and Lu Wang. 2021. Efficient attentions for long document summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1419–1436, Online. Association for Computational Linguistics.

Byeongchang Kim, Hyunwoo Kim, and Gunhee Kim. 2019. Abstractive summarization of Reddit posts with multi-level memory networks. 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 2519–2531, Minneapolis, Minnesota. Association for Computational Linguistics.

Anastassia Kornilova and Vladimir Eidelman. 2019. BillSum: A corpus for automatic summarization of US legislation. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 48–56, Hong Kong, China. Association for Computational Linguistics.

Mahnaz Koupaee and William Yang Wang. 2018. Wikihow: A large scale text summarization dataset. *arXiv preprint arXiv:1810.09305*.

Wessel Kraaij, Thomas Hain, Mike Lincoln, and Wilfried Post. 2005. The ami meeting corpus.

Wojciech Kryscinski, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Neural text summarization: A critical evaluation. 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 540–551, Hong Kong, China. Association for Computational Linguistics.

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online. 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 pre-training 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.

Junyi Li, Tianyi Tang, Jian-Yun Nie, Ji-Rong Wen, and Xin Zhao. 2022. Learning to transfer prompts for text generation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3506–3518, Seattle, United States. Association for Computational Linguistics.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online. Association for Computational Linguistics.

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

Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. 2022a. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. *arXiv preprint arXiv:2205.05638*.

Xiaochen Liu, Yang Gao, Yu Bai, Jiawei Li, Yinan Hu, Heyan Huang, and Boxing Chen. 2022b. PSP:

Pre-trained soft prompts for few-shot abstractive summarization. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6355–6368, Gyeongju, Republic of Korea. International Committee on 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.

Yang Liu, Ivan Titov, and Mirella Lapata. 2019. Single document summarization as tree induction. 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 1745–1755, Minneapolis, Minnesota. Association for Computational Linguistics.

Robert Logan IV, Ivana Balazevic, Eric Wallace, Fabio Petroni, Sameer Singh, and Sebastian Riedel. 2022. Cutting down on prompts and parameters: Simple few-shot learning with language models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2824–2835, Dublin, Ireland. Association for Computational Linguistics.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gùlçehre, 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.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *CoRR*, abs/2203.02155.

Laura Perez-Beltrachini and Mirella Lapata. 2021. Models and datasets for cross-lingual summarisation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9408–9423, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jonathan Pilault, Amine Elhattami, and Christopher Pal. 2020. Conditionally adaptive multi-task learning: Improving transfer learning in nlp using fewer parameters & less data. *arXiv preprint arXiv:2009.09139*.

Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. 2008. Collective classification in network data. *AI magazine*, 29(3):93–93.

Elizabeth Shriberg, Raj Dhillon, Sonali Bhagat, Jeremy Ang, and Hannah Carvey. 2004. The icsi meeting recorder dialog act (mrda) corpus.

Jiaan Wang, Fandong Meng, Duo Zheng, Yunlong Liang, Zhixu Li, Jianfeng Qu, and Jie Zhou. 2022. A Survey on Cross-Lingual Summarization. *Transactions of the Association for Computational Linguistics*, 10:1304–1323.

Yiming Wang, Zhuosheng Zhang, and Rui Wang. 2023. Element-aware summarization with large language models: Expert-aligned evaluation and chain-of-thought method. *arXiv preprint arXiv:2305.13412*.

Ruochen Xu, Chenguang Zhu, and Michael Zeng. 2022. Narrate dialogues for better summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3565–3575, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Ziyi Yang, Chenguang Zhu, Robert Gmyr, Michael Zeng, Xuedong Huang, and Eric Darve. 2020. TED: A pretrained unsupervised summarization model with theme modeling and denoising. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1865–1874, Online. Association for Computational Linguistics.

Tiezheng Yu, Zihan Liu, and Pascale Fung. 2021. AdaptSum: Towards low-resource domain adaptation for abstractive summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5892–5904, Online. Association for Computational Linguistics.

Wenhao Yu, Chenguang Zhu, Zaitang Li, Zhiting Hu, Qingyun Wang, Heng Ji, and Meng Jiang. 2022. A survey of knowledge-enhanced text generation. *ACM Computing Surveys (CSUR)*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020. 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.

Rui Zhang and Joel Tetreault. 2019. This email could save your life: Introducing the task of email subject line generation. In *Proceedings of the 57th Annual*

Meeting of the Association for Computational Linguistics, pages 446–456, Florence, Italy. Association for Computational Linguistics.

Yusen Zhang, Yang Liu, Ziyi Yang, Yuwei Fang, Yulong Chen, Dragomir Radev, Chenguang Zhu, Michael Zeng, and Rui Zhang. 2022. Macsum: Controllable summarization with mixed attributes. *arXiv preprint arXiv:2211.05041*.

Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2020. Extractive summarization as text matching. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6197–6208, Online. Association for Computational Linguistics.

Ming Zhong, Yang Liu, Yichong Xu, Chenguang Zhu, and Michael Zeng. 2022a. Dialoglm: Pre-trained model for long dialogue understanding and summarization. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 11765–11773. AAAI Press.

Ming Zhong, Yang Liu, Da Yin, Yuning Mao, Yizhu Jiao, Pengfei Liu, Chenguang Zhu, Heng Ji, and Jiawei Han. 2022b. Towards a unified multi-dimensional evaluator for text generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2023–2038, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Ming Zhong, Danqing Wang, Pengfei Liu, Xipeng Qiu, and Xuan-Jing Huang. 2019. A closer look at data bias in neural extractive summarization models. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 80–89.

Ming Zhong, Da Yin, Tao Yu, Ahmad Zaidi, Mutethia Mutuma, Rahul Jha, Ahmed Hassan Awadallah, Asli Celikyilmaz, Yang Liu, Xipeng Qiu, and Dragomir Radev. 2021. QMSum: A new benchmark for query-based multi-domain meeting summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5905–5921, Online. Association for Computational Linguistics.

Chenguang Zhu, Yang Liu, Jie Mei, and Michael Zeng. 2021. MediaSum: A large-scale media interview dataset for dialogue summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5927–5934, Online. Association for Computational Linguistics.

A Datasets in SummZoo

The final SummZoo contains following sub-tasks:

MultiNews (Fabbri et al., 2019) is a large-scale multi-document summarization dataset. The task is to generate a summary given multiple news articles.

XSum (Narayan et al., 2018) is an extreme text summarization dataset. Given a news article, the task is to generate a one-sentence summary.

Reddit-TIFU (Kim et al., 2019) is a social post summarization dataset. The task is to generate a short summary for posts from the online discussion forum Reddit.[6] Compared with news text, the text in Reddit-TIFU is less formal and structured.

ArXiv (Cohan et al., 2018) is a long scientific paper summarization dataset collected from ArXiv, including articles of multiple domains, such as physics, computer science, etc.

WikiHow (Koupaee and Wang, 2018) is a large-scale instruction summarization dataset. The task is to generate a short summary given the multiple-step instruction.

SAMSum (Gliwa et al., 2019) is a written conversation summarization dataset for Messenger-style chit-chats. Both dialogue and summary are annotated by experts.

DIALOGSUM (Chen et al., 2021) is a real-life scenario dialogue summarization dataset that covers a wide range of daily life dialogues, including diverse task-oriented dialogues. The testset of DIALOGSUM provides three reference summaries for each dialogue, we report the averaged results.

QMSum (Zhong et al., 2021) is a query-based meeting summarization dataset that is derived from Augmented Multi-party Interaction (AMI) corpus (Kraaij et al., 2005), the International Computer Science Institute (ICSI) (Shriberg et al., 2004) and Committee Meetings. The task is to generate a summary given a meeting and a query.

B Multi-Task Pre-Training Datasets

We use the following datasets for multi-task pre-training:

[6]We categorize it into single document summarization task because the posts of each input are from the same user, centring one event.

Dataset	Raw Size	Sam. Size
CNNDM	287,227	287,227
BillSum	23,455	113,694
PubMed	119,924	119,924
GovReport	19,466	105,114
MediaSum	463,596	100,000
SummScreen	22,588	67,764
XWikis	280,000	100,000
Total	–	893,723

Table 8: Statistics of multi-task pre-training data. We combine 7 summarization tasks. Raw size is the number of input and output pairs of raw datasets. Sam. (Sampled) size is the size of balanced data that are actually used in multi-task pre-training.

CNNDM (Nallapati et al., 2016) is a large news summarization dataset that contains articles and paired human annotated summaries from CNN and Daily Mail.

BillSum (Kornilova and Eidelman, 2019) consists of the US Congressional and California state bills, and summaries written by Legislative Counsel.

PubMed (Cohan et al., 2018) contains large long scientific articles and human labeled abstracts. Compared with ArXiv, which contains data from multiple domains, PubMed dataset focuses on the biomedical field.

GovReport (Huang et al., 2021) consists of long reports and summaries from government research agencies.

MediaSum (Zhu et al., 2021) is an interview summarization dataset that contains $463.6k$ transcripts and summaries from NPR and CNN.

SummScreen (Chen et al., 2022a) consists of long TV series transcripts and human written recaps.

XWikis (Perez-Beltrachini and Lapata, 2021) is a cross-lingual summarization dataset that contains Wikipedia articles and leading paragraphs in multiple languages. We only use the English data that have paired documents and summaries.

To balance the training data size of different datasets, we perform down-sampling on over-sized datasets and up-sampling on low-resource datasets respectively. The statistics of resulting data for pre-training are shown in Table 8.

Task		PEGASUS			BART-PT			MultiBART			UniSumm		
		D_{R1}	D_{R2}	D_{RL}	D_{R1}	D_{R2}	D_{RL}	D_{R1}	D_{R2}	D_{RL}	D_{R1}	D_{R2}	D_{RL}
MultiNews	10	0.37	0.30	0.21	1.04	0.37	0.23	0.68	0.20	0.10	0.33	0.27	0.23
	100	0.20	0.24	0.22	0.11	0.23	0.21	0.26	0.21	0.19	0.19	0.30	0.29
XSum	10	1.45	0.93	1.26	1.60	0.54	1.05	1.65	0.72	1.28	1.21	0.78	1.15
	100	0.37	0.31	0.31	0.27	0.28	0.30	0.11	0.08	0.05	0.27	0.18	0.23
Arxiv	10	0.57	0.09	0.28	1.08	0.54	0.87	0.32	0.36	0.29	0.93	0.31	0.83
	100	0.55	0.17	0.35	0.83	0.38	0.76	0.64	0.19	0.60	0.54	0.18	0.54
WikiHow	10	0.79	0.25	0.42	0.66	0.35	0.56	0.66	0.46	0.48	0.40	0.31	0.48
	100	0.46	0.21	0.31	0.25	0.15	0.22	0.38	0.26	0.31	0.21	0.10	0.15
Reddit	10	0.83	0.28	0.76	1.61	0.57	1.00	1.20	0.49	0.78	1.16	0.64	1.01
	100	0.71	0.31	0.50	0.72	0.39	0.57	0.68	0.43	0.61	0.52	0.26	0.49
DialogSum	10	1.18	0.90	1.13	0.96	0.68	0.65	1.46	1.01	1.02	0.99	0.76	0.80
	100	0.83	1.01	0.85	0.90	1.08	0.83	1.01	1.16	0.95	0.91	1.10	1.00
SAMSum	10	1.61	1.19	1.24	1.58	1.44	1.19	1.91	1.69	1.51	1.07	0.82	0.83
	100	0.47	0.39	0.60	0.29	0.54	0.57	0.40	0.47	0.50	0.47	0.30	0.41
QMSum	10	0.84	0.52	0.60	0.75	0.45	0.36	0.71	0.42	0.20	0.45	0.57	0.30
	100	0.72	0.82	0.72	0.55	0.55	0.41	0.34	0.32	0.24	0.30	0.23	0.17
Average	10	0.96	0.56	0.74	1.16	0.62	0.74	1.07	0.69	0.71	0.82	0.56	0.70
	100	0.54	0.43	0.48	0.49	0.45	0.48	0.48	0.39	0.43	0.43	0.33	0.41

Table 9: Comparison of model robustness towards different few-shot samples. We report the standard deviations of ROUGE scores on 5 sets of few-shot samples provided in SUMMZOO for each task and setting. D_{R1}, D_{R2} and D_{RL} mean the standard deviations of R1, R2 and RL, respectively. Lower standard deviation indicates the model is more robust towards different few-shot samples. The bottom block presents the averaged results of all 8 sub-tasks.

C Implementation Details

We use BART-large (Lewis et al., 2020) to initialize the summarization model of UNISUMM. All experiments are conducted on NVIDIA A100 GPU with PyTorch 1.11. The max input length and target length are set to 2,048 and 400. The hyperparameter choice is based on previous few-shot summarization work (Zhang et al., 2020; Fabbri et al., 2021; Chen and Shuai, 2021) and empirical consideration. For multi-task pre-training, we initialize from BART-large, and train the model on 16 GPUs with 300,000 steps, batch size of 32, learning rate of 1.5e-5, and warm-up with 4,000 steps. For few-shot tuning, we prefix-tune the model on 4 GPUs with 100 and 1000 steps for 10-shot and 100-shot, respectively, with batch size of 32, learning rate of 1.5e-4, and warm-up with 10% of the training steps. For XSum, the training steps are set to 10 and 100 for 10-shot and 100-shot, respectively, while other configurations are unchanged.

D Model Robustness

Table 9 shows the standard deviations of ROUGE-1, ROUGE-2 and ROUGE-L scores on 5 different sets of few-shot samples in SUMMZOO. Overall, UNISUMM shows the least standard deviations on most metrics across tasks in both settings, suggesting it is most robust and stable towards different selections of training samples.

E Human Evaluation

Following Kryscinski et al. (2019, 2020), we conduct human evaluation from 4 dimensions, which can offer a more robust and holistic perspective to understand summarization system (Zhong et al., 2022b):

- *Fluency* evaluates the quality of individually generated sentences, including grammar, word order, etc;
- *Coherence* evaluates the collective quality of generated summaries;
- *Relevance* evaluates the importance of information in the generated summaries;
- *Consistency* evaluates the factual alignment of the generated summary against the input document.

We ask a judge to give scores from 1 to 5 along these 4 dimensions. Higher score indicates better quality. The judge is a postgraduate student, who studied in the United Kingdom and has solid experience in evaluating summarization tasks.

Task		$P_{0.01}$+$L_{0.01}$ R2	$P_{0.05}$+$L_{0.05}$ R2	$P_{0.01}$+$L_{0.05}$ R2
MN	10	15.32	15.00	15.19
	100	15.47	15.82	15.86
XSum	10	6.52	6.41	7.20
	100	11.57	11.30	11.36
Arxiv	10	15.50	15.20	15.38
	100	16.50	16.15	16.42
WH	10	9.48	9.37	9.35
	100	11.81	11.72	11.73
Reddit	10	5.72	5.55	5.60
	100	6.17	6.23	6.17
DS	10	13.39	13.26	13.38
	100	15.71	15.74	15.64
SS	10	18.55	18.38	18.53
	100	20.93	20.96	20.65
QMSum	10	12.06	12.04	12.12
	100	13.40	13.73	13.89
Average	10	12.07	11.90	12.09
	100	13.95	13.96	13.97

Table 10: Results of UNISUMM using different combinations of weight decay rates for multi-task training. $P_{0.01}$ indicates the weight decay rate for prefix parameters is 0.01 and $L_{0.01}$ indicates the weight decay rate for LM parameters is 0.01.

F Case Study

We qualitatively demonstrate the advantages of UNISUMM (100-shot) using cases from MultiNews and QMSum, and present an error analysis using case from WikiHow.

As shown in Table 11 (MultiNews), we see that the UNISUMM generates a summary with similar events and faithful descriptions compared with the gold summary. However, PEGASUS generated summary contains factual errors ("*... was last seen in a package shipped to the us from belgium.*") while the summary generated by UNISUMM ("*... unearthed ... shipment from belgium to newark*") is consistent with the gold summary and input ("*... turned up ... shipped from belgium.*"). This shows that UNISUMM has the ability to collect important information from multiple news reports and generate high-quality summaries, which is a task that the model has never seen during multi-task pre-training.

Also, as shown in Table 12 (QMSum), compared with gold summary, although the summary generated by UNISUMM is longer, it is highly relevant to the query. And UNISUMM properly rephrases the key utterance from the source meeting into an objective description, which suits the characteristic of conversation summarization. In contrast, the summary generated by PEGASUS misses important contents and contains irrelevant sentences compared with UNISUMM and human annotation. This evidence shows that UNISUMM successfully learns important characters of query-based meeting summarization task with only 100 samples.

An error case where UNISUMM fails can be found in Table 14 (WikiHow). UniSumm mistakenly generates "*...matches the text of the letter...*", where the ground truth should be the "*...matches... the one (address)...on the envelope*". Moreoever, the summary generated by UniSumm is a bit repetitive in wording, e.g., serveral repeated phrases "*... on the inside of the letter...*".

We present more cases in Table 13 (ArXiv and DIALOGSUM), Table 14 (XSum) and Table 15 (SAMSum and Reddit). Overall, we find that UNISUMM is capable of generating very fluent, relevant, faithful and human-like summaries on diverse unseen tasks. This verifies UNISUMM's great generalization ability in the few-shot scenario.

G Influence of Weight Decay

In § 3.4, we design a separated weight decay strategy to circumvent negative transfer in multi-task learning. In Table 10, we examine whether the combination of different weight decay rates (d_p for prefixes and d_l for the summarization model) is beneficial. Specifically, we report ROUGE-2 scores on SUMMZOO with different combinations of weight decay rates. We can see that the model performs the best with $d_p = 0.05$ and $d_l = 0.01$. And this asymmetrical weight decay is especially helpful to 10-shot XSum, which is more distinct from pre-training summarization tasks and relies more on general summarization knowledge.

MultiNews	
Input text	paris (ap) the pompidou centre in paris hopes to display a long-vanished picasso painting in may, now that it has been recovered by u.s. customs authorities. this undated photo provided by the united states department of justice, shows a cubist painting entitled the hairdresserby pablo picasso. authorities say the painting worth millions of dollars was ... ||||| a stolen picasso worth millions of dollars was shipped to the u.s. in a package that described it as a $37 "art craft " but it will soon be on its way back to france...
PEGASUS	– a 1911 cubist painting worth millions of dollars has been recovered after being mysteriously transported from france to the us nearly a decade ago. the pablo picasso painting, titled la coiffeuse, was reported missing from a pompidou storeroom in paris in 2001 and was last seen in a package shipped to the us from belgium, the ap reports. federal prosecutors in brooklyn filed papers thursday to seize the painting, which was hidden in a climate-controlled storage facility in queens, ny, and will soon be returned to france, reports the new york times. the center for art recovery in paris says it hopes to have the painting back in may, at a time when the cultural world is reeling from islamic state videos showing the destruction of statues in iraq. "this is a true comfort for us at a time when the cultural world is reeling from the islamic state video showing the destruction of statues in iraq," says alain seban, the director of the pompidou centre in paris. "it is a work of art that represents the genius of pablo picasso and was bequeathed to us in 2001 by the master of france's national museum."
GPT-3.5	– the new york times has a front-page story friday detailing how former fbi chief james comey tried to instruct president trump about the proper boundaries that should exist between them.
UNISUMM	- a century-old picasso painting missing from paris for more than a decade resurfaced in the united states under false pretenses as a $37 holiday-themed "art craft," the ap reports. the 1911 painting, called "la coiffeuse," was unearthed in december in a fedex shipment from belgium to newark. the canvas had been smuggled out of a storeroom of the centre georges pompidou, the paris museum and arts center. officials there thought it was safe and sound until they received a loan request for it in 2001. they searched the storerooms and could not find it. on dec. 17, someone going by "robert" with an address in belgium brought the painting to a climate-controlled warehouse in long island city and was seized, the ap reports. the us attorney for the eastern district of new york filed a civil complaint thursday to forfeit the picasso and return it to france. the canvas, titled "la coiffeuse," was bequeathed to the national museums of france in 1998. the pompidou director alain seban said in a statement friday that he hopes the work can be exhibited again publicly in may. no arrests have been made in the case.
Gold	- a picasso painting that was found to have vanished from a paris museum more than a decade ago has turned up 2014 in the us, in a package shipped from belgium. its papers identified it as a $37 "art craft/toy" and also included the line "joyeux noel," or merry christmas, the new york times reports. a man named "robert" attempted to send the package to a climate-controlled warehouse in queens, new york, in december, but custom officials at the port of newark seized what turned out to be picasso's 1911 la coiffeuse (the hairdresser). french museum officials traveled to new york last month and confirmed the find is indeed the missing picasso work, which the centre georges pompidou realized was missing from its storerooms in 2001 following a loan request; it was then valued at more than $2.5 million. court documents don't specify whether the sender or would-be recipient have been identified, nbc news reports, but "a lost treasure has been found," us attorney loretta lynch said in a statement. she filed a civil complaint yesterday that will have the painting returned to france. the pompidou's director, who called the painting's rediscovery a "true comfort," hopes the painting can go on display at the museum as early as may, the ap reports. (another picasso work was stolen days before this one was found.)

Table 11: Case study (a): Gold and generated summaries by 1-shot `text-davinci-002` (GPT3.5), 100-shot PEGASUS and UNISUMM on MultiNews, respectively. "|||||" seperates different reports in the input.

	QMSum
Input text	**what did barry hughes think about draft changes to cps guidance when talking about the divergence between the law in england and wales?** suzy davies am: thank you. i was going to ask this question later, but i think it fits in . i've forgotten what i was going to ask; this is terrible. oh no, that's right obviously, this act is on a particular trajectory at the moment, and it's due to be introduced before the end of this assembly. do you think it would be better for us as scrutinisers of this act if we could see the draft changes to cps guidance on the public interest test before we make our final decision? barry hughes: i honestly don't think that would necessarily be helpful. i've had some discussions with kwame, who would have an involvement in this. ...
PEGASUS	barry hughes thought it would be better for the members to see the draft changes to cps guidance on the public interest test before making their final decision. he added that they would need to work that up as they went along, and it would be a risk of putting the cart before the horse.
GPT-3.5	Barry Hughes thinks that it would not be helpful for the public to see the draft changes to the CPS guidance on the public interest test before the final decision is made on the bill. He believes that it would be better to take the general principles set out in the code for crown prosecutors and provide more specific details around them that relate to the issues being discussed in the bill.
UNISUMM	barry hughes thought that the draft changes to cps guidance would not necessarily be helpful. he thought that they would simply want to take the present public interest factors set out very clearly in the code for crown prosecutors, and provide a degree of detail around those that relates more specifically to the issues that they were discussing. so, it would be taking principles of generality and according them a degree of specificity. and they'd need to work that up as they go along.
Gold	barry hughes thought draft changes to cps guidance would not necessarily be helpful to the scrutinisers, and simply wanted to take the present interest factors. barry thought they needed to work for the principles of generality and a degree of specificity.

Table 12: Case study (b): Gold and generated summaries by 1-shot `text-davinci-002` (GPT3.5), 100-shot PEGASUS and UNISUMM on QMSum, respectively. The **blod** is the query.

ArXiv	
Input text	rapid progress in the design and manufacture of optical fiber systems is a result of worldwide demand for ultra - high bit - rate optical communications . this explains the growing interest of the soliton community in soliton - based optical fiber communication systems . this area of research was considerably advanced in recent years @ xcite . the most remarkable results include the application of the concept of the dispersion management to _ temporal optical solitons _ and soliton - based optical transmission systems , and the discovery of the so - called _ dispersion managed soliton _ . high - speed optical communications require effective components such as high - performance broadband computer networks that can be developed by employing the concept of the bit - parallel - wavelength (bpw) pulse transmission that offers many of the advantages of both parallel fiber ribbon cable and conventional wavelength - division ...
PEGASUS	the physical properties of multi - frequency solitary waves propagating in optical fibers and waveguides are discussed in the context of the soliton - based optical dispersion management. the emphasis is placed on the role of the spatial dispersion in the propagation of such solitary waves. it is shown that the propagation of such waves in optical fibers and waveguides is affected not only by the parametric difference between the pulse and the beam velocity, but also by the spatial variation of the dispersion profile. as a consequence, we argue that the role of the spatial dispersion in the propagation of solitary waves in optical fibers and waveguides is not limited to the parametric difference between the pulse and the beam velocity, but also includes the role of the spatial dispersion itself.
GPT-3.5	This article discusses the various types of solitary waves that can occur in nonlinear optical systems, with a focus on how to extract information about the propagation characteristics from experimental data. The examples of temporal and spatial solitons are given, along with a discussion of the differences between them. The difficulties associated with experimental measurements are also addressed, along with suggestions for ways to overcome them.
UNISUMM	the aim of the present paper is to present a versatile scheme for the study of spatial and temporal solitary waves . this approach is based on the theory of spatio - temporal wave propagation , where the spatial beam profile along the transverse direction is similar to the temporal pulse profile in fibers . in all such systems , the nonlinear interaction between the waves of two (or more) frequencies is the major physical effect that can support coupled - mode multi - frequency solitary waves . we will illustrate three common examples : multi - wavelength solitary waves in bit - parallel - wavelength optical fiber links , multi - colour spatial solitons due to multistep cascading in optical waveguides with quadratic nonlinearities , and quasiperiodic solitons in the fibonacci superlattices .
Gold	we discuss several novel types of multi - component (temporal and spatial) envelope solitary waves that appear in fiber and waveguide nonlinear optics . in particular , we describe multi - channel solitary waves in bit - parallel - wavelength fiber transmission systems for high performance computer networks , multi - colour parametric spatial solitary waves due to cascaded nonlinearities of quadratic materials , and quasiperiodic envelope solitons due to quasi - phase - matching in fibonacci optical superlattices . 2
DIALOGSUM	
Input text	person1: ms. dawson, i need you to take a dictation for me. person2: yes, sir... person1: this should go out as an intra-office memorandum to all employees by this afternoon. are you ready? person2: yes, sir. go ahead. person1: attention all staff... effective immediately, all office communications are restricted to email correspondence and official memos. the use of instant message programs by employees during working hours is strictly prohibited. ...
PEGASUS	person1 advises dawson to sign an intra-office memorandum restricting office communications to email correspondence and official memos. person2 asks if this applies to internal and external communications. person1 tells them so.
GPT-3.5	person1 needs Ms. Dawson to take a dictation for an intra-office memorandum that is to be distributed to all employees by 4 pm. The memo is to inform all staff that effective immediately, all office communications are restricted to email correspondence and official memos; the use of instant message programs by employees during working hours is strictly prohibited. At first offense, the employee will receive a warning and be placed on probation. The second offense will result in termination.
UNISUMM	person1 asks ms. dawson to take a dictation for person1. person1 restricts office communications to email correspondence and official memos and prohibits the use of instant message programs by employees during working hours.
Gold	ms. dawson helps person1 to write a memo to inform every employee that they have to change the communication method and should not use instant messaging anymore.

Table 13: Case study (c): Gold and generated summaries by 1-shot `text-davinci-002` (GPT3.5), 100-shot PEGASUS and UNISUMM on ArXiv and DIALOGSUM, respectively.

WikiHow	
Input text	before folding the letter, make sure that the address on the inside of the letter matches with the one that you have written or printed on the envelope.this will help avoid any mixups. don't forget to check that you've signed the letter. make sure that the text is facing up and towards you. this is your last chance to proofread and double check that you haven't forgotten anything.for example, have you dated the letter? are there any spelling or grammar mistakes? , take the bottom of the letter and fold it so that it is about half an inch (about 1 cm) from the top of the page.you can place the envelope underneath the letter to use as a guide. make sure that when you have folded the letter up, it is small enough to fit inside the envelope. make sure to line up all the outside edges of the paper before creasing so that you don't end up with a crooked fold. if the fold is crooked, your letter may not fit into the envelope.use a ruler to make the crease nice and sharp. you can do this by holding a ruler sideways and using the thin edge. slide the edge of the ruler along the letter on the creases so they are nice and flat. you will now take the right part of the letter and fold it about one third of the way in.line up the top and bottom edges of the letter and then crease the fold. take the other side of the letter and fold it one third of the way in. this will be the same thing that you did with the right side.line up the top and bottom of the edges so they are perfectly straight before creasing. the last crease you made should go into the bottom of the envelope first. place it so that the folds face the backside of the envelope.this will make it easy for your recipient to find where to begin unfolding the letter.
PEGASUS	take the top of the letter and fold it so that it is about half an inch (about 1 cm) from the top of the page. place the envelope underneath the letter. make sure that the envelope is flat on the inside. crease the folded side of the letter.
GPT-3.5	The document explains how to fold a letter so that it will fit into an envelope. First, make sure that the address on the letter matches the address on the envelope. Next, fold the bottom of the letter up to the top of the page, making sure to crease the fold. Then, fold the right side of the letter in to the middle, and repeat with the left side. Finally, insert the letter into the envelope so that the folds are facing the back.
UNISUMM	check that the address on the inside of the letter matches the text of the letter to make sure that the address on the inside of the letter matches with the one on the page. make sure that the text matches with the one on the inside of the letter. make sure that the envelope is small enough to fit into the envelope.
Gold	check the address. lay the letter on a table. fold the bottom half up. crease the fold. fold the right half of the letter inwards. fold the left half of the letter inwards. turn the letter sideways and insert into the envelope.
XSum	
Input text	the sunday times says the missile veered off course during a test in june last year - weeks before the commons voted to spend 40bn renewing trident . questioned by andrew marr, the pm refused to say four times if she had known about the test ahead of the vote . the snp's nicola sturgeon called for a ' full disclosure ' of what happened . according to the sunday times, an unarmed trident ii d5 missile veered off in the wrong direction towards the us - instead of towards africa - when it was launched from a british submarine off the coast of florida . in july - days after mrs may had become prime minister - mps voted overwhelmingly in favour of replacing trident . during the debate, mrs may told mps it would be ' an act of gross irresponsibility ' for the uk to abandon its nuclear weapons . mps backed its renewal by 472 votes to 117 . however, all 52 snp mps voted against it - as did labour leader jeremy corbyn . when asked on the bbc's andrew marr show whether she had known then that a misfire had happened, mrs may said: ' i have absolute faith in our trident missiles . ' when i made that speech in the house of commons, what we were talking about was whether or not we should renew our trident . ' she was asked a further three times - but did not answer the questions . the ministry of defence did not give details of the test process but said it had been a success . scottish first minister, mrs sturgeon - a long-standing opponent of trident, whose submarines are based at faslane, on the river clyde - said the apparent misfire was a ' hugely serious issue ' . she tweeted: ' there should be full disclosure of what happened, who knew what/when, and why the house of commons wasn't told . ' meanwhile, mr corbyn said the reports called for ' a serious discussion ' . ...
PEGASUS	the prime minister has been accused of " side-stepping " questions about a submarine-launched nuclear-capable missile which misfired during a test.
GPT-3.5	trident is the uk's nuclear weapons system, consisting of submarines, missiles and warheads. it is operated by the royal navy and has been in use since the 1990s. reports that a missile test in june 2016 went wrong have raised questions about the system's effectiveness and whether it was deliberately kept quiet ahead of a key parliamentary vote on its renewal.
UNISUMM	a nuclear missile launched a british submarine failed during a test, according to a report in the sunday times.
Gold	theresa may is coming under pressure to say whether she knew about a reported misfire of the uk's nuclear weapons system before a crucial commons vote.

Table 14: Case study (d): Gold and generated summaries by 1-shot `text-davinci-002` (GPT3.5), 100-shot PEGASUS and UNISUMM on WikiHow and XSum, respectively.

SAMSum	
Input text	cora: have you heard how much fuss british media made about meet and greet with james charles in birmingham? ellie: no...! what happened? cora: well, there was a meet and greet with james charles in one of the malls in birmingham and about 8000 fans showed up for it. cora: it cause a gridlock around the mall and - of course - british media had to make some (quite negative) comments on it. ellie: they came for sister james?! >:(ellie: i sister snapped!! :p :d cora: haha :d cora: you shouldn't watch so much youtube, you're getting weirder and weirder. :d ellie: sister shut up :p so, what did they say? cora: ;) : * daily mail" was surprised that a meet and greet with a "virtually unknown" youtuber gathered 8000 people. :p cora: a host from lbc tried to find an answer to an unanswerable question: "who is james charles?" ...
PEGASUS	ellie's meet and greet with james charles in birmingham got some negative comments from british media. cora thinks it's ridiculous that a man can wear makeup.
GPT-3.5	cora and ellie were discussing the meet and greet with james charles in birmingham. 8000 fans showed up and it caused a gridlock. british media was surprised and some hosts couldn't wrap their heads around the fact that a guy can wear makeup.
UNISUMM	about 8000 fans showed up for the meet and greet with james charles in birmingham.
Gold	there was a meet-and-greet with james charles in birmingham which gathered 8000 people.
Reddit	
Input Text	tifu making lunch. made myself a lush sandwich for lunch today. smoked ham, cheddar cheese, a dash of bbq sauce on thick cut wholemeal bread. popped it in a small tupperware box, for freshness. i found some bn type biscuits in the cupboard, so i pop them in the box on top of my sandwiches. think nothing of it. eventually lunch time comes around. i discover, to my horror, that the biscuits have absorbed the moisture from the sandwich. and now i have a dried out sandwich and soggy biscuits
PEGASUS	made a luch sandwich for lunch, only to have it ruined by soggy biscuits. i'm completely fucked.
GPT-3.5	made a sandwich and it turned out terrible because the biscuits absorbed all the moisture.
UNISUMM	made a big sandwich, with cheddar cheese, bbq sauce and bn type biscuits. popped them in a tupperware box, and discover they've dried out
Gold	don't put your sandwich and biscuits in same box, unless you like dry bread and soggy biscuits.

Table 15: Case study (e): Gold and generated summaries by 1-shot `text-davinci-002` (GPT3.5), 100-shot PEGASUS and UNISUMM on SAMSum and Reddit, respectively.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Limitations

☑ A2. Did you discuss any potential risks of your work?
Ethics Statement

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract and Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 3, Section 4, Section 5 and Appendix C.

☑ B1. Did you cite the creators of artifacts you used?
Section 3, Section 4, Section 5 and Appendix C.

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Ethics Statement

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Section 3, Section 4, Section 5, Appendix A, Appendix B and Appendix C.

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Ethics Statement

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Section 3, Section 4, Section 5, Appendix A, Appendix B and Appendix C.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Appendix A and Appendix B.

C ☑ Did you run computational experiments?

Section 5, Section 6 and Appendix C

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Appendix C

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Appendix C

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 5 and Section 6

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Section 3, Section 4, Section 5, Appendix A, Appendix B and Appendix C.

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Section 7

☑ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Section 7 and Appendix E

☑ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Ethics Statement

☑ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Section 7 and Appendix E

☑ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Section 7 and Appendix E

☑ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Appendix E