AdaptSum: Towards Low-Resource Domain Adaptation for Abstractive Summarization

Tiezheng Yu*, **Zihan Liu***, **Pascale Fung**
Center for Artificial Intelligence Research (CAiRE)
Department of Electronic and Computer Engineering
The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong
{tyuah,zliucr}@connect.ust.hk, pascale@ece.ust.hk

Abstract

State-of-the-art abstractive summarization models generally rely on extensive labeled data, which lowers their generalization ability on domains where such data are not available. In this paper, we present a study of domain adaptation for the abstractive summarization task across six diverse target domains in a low-resource setting. Specifically, we investigate the second phase of pre-training on large-scale generative models under three different settings: 1) source domain pre-training; 2) domain-adaptive pre-training; and 3) task-adaptive pre-training. Experiments show that the effectiveness of pre-training is correlated with the similarity between the pre-training data and the target domain task. Moreover, we find that continuing pre-training could lead to the pre-trained model's catastrophic forgetting, and a learning method with less forgetting can alleviate this issue. Furthermore, results illustrate that a huge gap still exists between the low-resource and high-resource settings, which highlights the need for more advanced domain adaptation methods for the abstractive summarization task.[1]

1 Introduction

Abstractive summarization models aim to extract essential information from long documents and to generate short, concise and readable text. Recently, neural abstractive summarization models have achieved remarkable performance (Gehrmann et al., 2018; Paulus et al., 2018), and large-scale generative pre-training (Lewis et al., 2019; Raffel et al., 2019) has shown itself to be surprisingly effective at generation tasks, including abstractive summarization. However, these models generally require large numbers of human-annotated summaries to achieve state-of-the-art performance, which makes them not scalable to low-resource domains where only a few labeled data are available.

Domain adaptation methods have naturally arisen to tackle the low-resource issue and enable models to quickly adapt to target domain tasks. Yet, despite their practicality, very few studies have used domain adaptation methods on the low-resource scenario for the abstractive summarization task. To address this research gap, we present **AdaptSum**, the first benchmark to simulate the low-resource domain **Adapt**ation setting for abstractive **Sum**marization systems with a combination of existing datasets across six diverse domains (dialog (Gliwa et al., 2019), email (Zhang and Tetreault, 2019), movie review (Wang and Ling, 2016), debate (Wang and Ling, 2016), social media (Kim et al., 2019), and science (Yasunaga et al., 2019)), and for each domain, we reduce the number of training samples to a small quantity so as to create a low-resource scenario.

Recently, conducting a second pre-training step on large-scale language models (e.g., BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019a)) has proven to be effective for domain adaptation tasks (Lee et al., 2020; Gururangan et al., 2020). However, the current methods incorporating such a step are mainly focused on classification or classification-based (e.g., named entity recognition) tasks, leaving a research gap in exploring their use for generation tasks. In this paper, we systematically investigate adding a second phase of pre-training on large-scale generative models under three settings: 1) source domain pre-training (SDPT) based on a labeled source domain summarization dataset; 2) domain-adaptive pre-training (DAPT) based on an unlabeled substantial domain-related corpus; and 3) task-adaptive pre-training (TAPT) based on an unlabeled small-scale task-related corpus. The second phase of pre-training could cause the catastrophic forgetting in the pre-trained model. Thus, we propose to apply

* Equal contributions. Listing order is random.

[1] The code and data are released at: https://github.com/TysonYu/AdaptSum

RecAdam (Chen et al., 2020) into the pre-training process to alleviate this issue and further improve the adaptation performance.

Experimental results show that SDPT and TAPT can generally improve on the performance of the fine-tuning method, while the effectiveness of DAPT is correlated to the similarity between the pre-training data and the target domain task data. Different from previous insights into adaptive pre-training on classification tasks (Gururangan et al., 2020), we find that in the summarization task, DAPT could make the adaptation performance worse, even though the pre-training corpus is collected from domain-related sources. Furthermore, we show that RecAdam can further boost the performance of the second pre-training step by effectively maintaining the pre-trained model's knowledge gained in the first phase of pre-training.

Our contributions are summarized as follows:

- We introduce a low-resource domain adaptation scenario for the abstractive summarization task to move towards the fast adaptation of summarization systems.
- To the best of our knowledge, we are the first to systematically study the domain- and task-adaptative pre-training for a low-resource generation task.
- Our work highlights the research questions and challenges in the low-resource abstractive summarization task, which we hope will catalyze research in this area.

2 Related Work

2.1 Abstractive Summarization

Abstractive summarization aims to generate short, concise and readable text that captures the core meaning of the input documents. Neural networks have achieved remarkable results for the abstractive summarization due to the emergence of Seq2Seq models (Sutskever et al., 2014) and attention mechanisms (Bahdanau et al., 2014). See et al. (2017), Paulus et al. (2017) and Gehrmann et al. (2018) applied a pointer network to solve the out-of-vocabulary issue. Further, See et al. (2017) used a coverage mechanism (Tu et al., 2016) to keep track of the already summarized content, which discourages repetition, while Paulus et al. (2017) and Chen and Bansal (2018) combined reinforcement learning into an end2end setting. Recently, pre-trained language models (Peters et al., 2018; Radford et al., 2018; Devlin et al., 2019; Dong et al., 2019; Lewis et al., 2019) have achieved impressive gains in a wide variety of natural language tasks. Many studies on the use of pre-trained language models in the abstractive summarization task (Liu and Lapata, 2019; Yan et al., 2020; Su et al., 2020; Yu et al., 2020) have been undertaken and have achieved the state-of-the-art performance.

Domain	Unlabeled Corpus		Labeled data		
	# Tokens	Size	Train	Valid	Test
Dialog	44.96M	212MB	300	818	819
Email	117.54M	705MB	300	1960	1906
Movie R.	11.36M	62MB	300	500	2931
Debate	122.99M	693MB	300	956	1003
Social M.	153.30M	786MB	300	1000	1000
Science	41.73M	291MB	100	350	497

Table 1: Data statistics of AdaptSum for the unlabeled corpus and labeled summarization data across the six domains ("R." and "M." are the abbreviations for Review and Media, respectively).

2.2 Domain Adaptation

Domain adaption for natural language processing and computer vision tasks is widely studied (Blitzer et al., 2007; Mansour et al., 2008; Daumé III, 2009; Sandu et al., 2010; Foster et al., 2010; Wang and Cardie, 2013; Sun et al., 2016; Liu et al., 2019b, 2020b; Gururangan et al., 2020; Winata et al., 2020; Jadon, 2020; Yin, 2020; Liu et al., 2020a,d). However, little has been done to investigate domain adaption for the abstractive summarization task. Hua and Wang (2017) first studied the adaptation of neural summarization models and showed that the models were able to select salient information from the source domain data. Wang et al. (2019) investigated the domain shift problem for the extractive summarization task. Recently, Magooda and Litman (2020) studied cross-domain transfer between two entirely different domains and introduced data synthesis methods. To the best of our knowledge, we are the first to systematically study the domain- and task-adaptative pre-training based on the pre-trained generative model in the low-resource abstractive summarization task across multiple diverse domains.

3 AdaptSum

The goal of AdaptSum is to provide an accessible benchmark for the evaluation of low-resource domain adaptation for abstractive summarization on a diverse set of domains. The vocabulary overlaps between domains are shown in Figure 1. AdaptSum consists of six diverse target domains and the corresponding unlabeled domain-related corpora for DAPT. We provide the data statistics of all domains in Table 1, and the details are as follows.

Dialog Gliwa et al. (2019) introduced a human-annotated abstractive chat dialog summarization dataset. The unlabeled dialog corpus from different sources, namely, Reddit conversations,[2] personalized dialogs (Zhang et al., 2018), empathetic dialogs (Rashkin et al., 2019), and Wizard of Wikipedia dialogs (Dinan et al., 2019).

Email Zhang and Tetreault (2019) introduced an abstractive business and personal email summarization dataset which consists of email and subject pairs. We collect the unlabeled email corpus from the Enron Email Dataset.[3]

Movie Review Wang and Ling (2016) introduced a human-annotated abstractive movie review summarization dataset. We collect the unlabeled corpus for this domain from IDMB Movie Review (Maas et al., 2011).

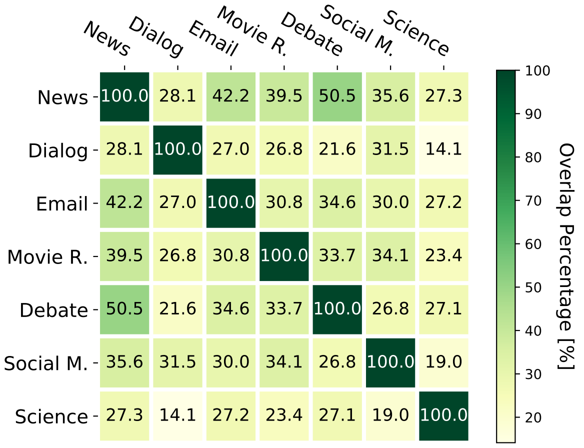
Debate Wang and Ling (2016) introduced an abstractive debate summarization dataset which consists of arguments and the debate topic pairs. The unlabeled corpus is from Ajjour et al. (2019).

Social Media Kim et al. (2019) introduced an abstractive summarization dataset of Reddit TIFU posts, where the summary for each post come from its title. We collect the unlabeled corpus directly from Reddit TIFU.[4]

Science Yasunaga et al. (2019) introduced a human-annotated abstractive summarization dataset on computational linguistics. We collect the unlabeled domain corpus from the ACL anthology (Bird et al., 2008).

4 Methodology

In this section, we will first introduce the three different settings that we investigate for a second pre-training step. Then, we will discuss how we cope with the catastrophic forgetting issue in the second phase of pre-training.



	News	Dialog	Email	Movie R.	Debate	Social M.	Science
News	100.0	28.1	42.2	39.5	50.5	35.6	27.3
Dialog	28.1	100.0	27.0	26.8	21.6	31.5	14.1
Email	42.2	27.0	100.0	30.8	34.6	30.0	27.2
Movie R.	39.5	26.8	30.8	100.0	33.7	34.1	23.4
Debate	50.5	21.6	34.6	33.7	100.0	26.8	27.1
Social M.	35.6	31.5	30.0	34.1	26.8	100.0	19.0
Science	27.3	14.1	27.2	23.4	27.1	19.0	100.0

Figure 1: Vocabulary overlaps of the summarization validation set between domains. The News domain is the source domain and the other six domains are low-resource target domains. Vocabularies for each domain are created by considering the top 10K most frequent words (excluding stopwords). We observe that the vocabulary overlaps between domains are generally small, which illustrates that the overlaps between domains are comparably small and the chosen domains are diverse.

4.1 A Second Phase of Pre-Training

We conduct a second pre-training phase based on a pre-trained generative model, BART (Lewis et al., 2019), on three different settings. Then, we fine-tune it to the summarization task in the target domains. The three settings are described as follows.

Source Domain Pre-Training (SDPT) Inspired by the cross-domain setting (Jia et al., 2019; Liu et al., 2020c,d), we leverage substantial training samples from a source (News) domain (XSum (Narayan et al., 2018)), to aid in the fast adaptation in target domains. We choose the News domain as the source domain because it is a rich-resource domain in the summarization task, and from Figure 1, the similarity between this domain and target domains is generally low which increases the challenge of the domain adaptation.

Our method to conduct SDPT is straightforward. We continue pre-training BART using the source domain summarization data. The objective function for this pre-training is not the sentence reconstruction, as in the original pre-training of BART. Instead, we utilize the supervisions from the source domain summarization data to train BART on the summarization task. The purpose of this

[2] https://github.com/PolyAI-LDN/conversational-datasets/tree/master/reddit

[3] https://www.cs.cmu.edu/~./enron/

[4] https://convokit.cornell.edu/documentation/subreddit.html

pre-training is to inject the task knowledge into the pre-trained language model so that the model can quickly adapt to the same task in target domains.

Domain-Adaptive Pre-Training (DAPT) We leverage an unlabeled domain-related corpus to continue pre-training BART using its original pre-training objective function (corrupting documents and then optimizing a reconstruction loss—the cross-entropy between the decoder's output and the original document). The intuition behind this method is to introduce the domain knowledge into the pre-trained language model so as to enable its fast adaptation to the target domains.

Task-Adaptive Pre-Training (TAPT) The size of the domain-related corpus for DAPT is usually enormous, which results in two potential drawbacks. First, such a large corpus might not be always available, especially for the low-resource domains. Second, pre-training on such a large corpus is time-consuming and requires excessive computational resources. Therefore, investigating pre-training on a smaller unlabeled corpus is a practical and beneficial research direction. TAPT refers to pre-training on a set of the unlabeled documents in the target domain's summarization task. Compared to DAPT, TAPT uses a much smaller but far more task-relevant pre-training corpus since it directly uses the input documents from summarization task. This setting makes TAPT much less expensive to run and independent of the collection of the large domain-related corpus.

4.2 Recall and Learn

Although the second pre-training step allows the pre-trained model to learn the task or domain knowledge, it might lead to the catastrophic forgetting issue and cause the pre-trained model to partly lose the language understanding ability that it gains in the first pre-training step. To alleviate this issue, we expect the pre-trained model to recall the previously learned knowledge during the process of learning new knowledge. A straightforward way to achieve this goal is to borrow the idea of continual learning methods (Kirkpatrick et al., 2017; Lopez-Paz and Ranzato, 2017; Chen et al., 2020). In this paper, we adopt RecAdam from Chen et al. (2020) for the second phase of pre-training to weaken the catastrophic forgetting issue. The reason for choosing RecAdam is twofold: 1) it does not require the first step pre-training data from the pre-trained model, which is usually not available; 2) it is the most recent approach that is being successfully applied to natural language processing tasks. The RecAdam is introduced as follows.

Based on the Adam optimizer (Kingma and Ba, 2015), RecAdam reconstructs the objective function to allow it to gradually shift to the target task:

$$Loss = \lambda(t) \cdot Loss_T + (1 - \lambda(t)) \cdot Loss_S, \quad (1)$$

$$\lambda(t) = \frac{1}{1 + \exp(-k \cdot (t - t_0))}, \quad (2)$$

where k and t_0 are the hyper-parameters controlling the annealing rate and time steps, $Loss_T$ represents the target task objective function, and $Loss_S$ is used to simulate the first pre-training step of the pre-trained model. $Loss_S$ can be simplified as:

$$Loss_S = \frac{1}{2}\gamma \sum_i (\theta_i - \theta_i^*)^2, \quad (3)$$

where $\frac{1}{2}\gamma$ is the coefficient of the quadratic penalty, θ is the parameters of the model, and θ^* (fixed) is the original parameters of the pre-trained model.

Although RecAdam has shown its effectiveness in fine-tuning BERT-like models (e.g., BERT (Devlin et al., 2019) and ALBERT (Lan et al., 2020)) to the GLUE benchmark (Wang et al., 2018), exploring the effectiveness of RecAdam in the second phase of pre-training for generative pre-trained models is not trivial. First, the second pre-training step of a language model is a completely different task compared to fine-tuning to downstream tasks. Second, a generative model (e.g., BART) is structurally different from BERT-like models. Third, the corpus sizes for SDPT and DAPT are generally much larger than the sizes of GLUE tasks, which could affect the learning process.

5 Experimental Setup

Training Details We evaluate all of our models on AdaptSum. For the dialog and email domains, we use the standard splits of (Gliwa et al., 2019; Zhang and Tetreault, 2019), while for movie review, debate, social media and science domains, we split the whole dataset into training, validation and test sets by ourselves since the original works do not specify how to split these datasets or the published datasets do not contain the split training, validation and test sets. Since the dataset sizes are limited for science, movie review and dialog domains, the maximum training samples for these domains are

Models	Dialog	Email	Movie R.	Debate	Social M.	Science	Average
BART Fine-tuning	39.95	24.71	25.13	24.48	21.76	72.76	34.80
SDPT	42.84	25.16	25.45	25.61	22.43	**73.09**	35.76
w/ RecAdam	**45.23**	**26.97**	**26.06**	25.17	**23.25**	72.60	**36.55**
DAPT	41.22	26.50	24.25	**26.71**	22.95	71.88	35.59
w/ RecAdam	40.05	25.66	25.78	25.01	21.51	72.23	35.04
TAPT	40.15	25.30	25.27	24.59	22.81	73.08	35.20
w/ RecAdam	41.34	25.73	25.65	24.70	23.01	72.80	35.54

Table 2: ROUGE-1 scores on different pre-training methods compared to the baseline BART over all domains.

Corpus	Dialog	Email	Movie R.	Debate	Social M.	Science	Average
DAPT	212MB	705MB	62MB	693MB	786MB	291MB	458.2MB
TAPT	7.9MB	14MB	3.3MB	2.4MB	74MB	384KB	17.0MB

Table 3: Corpus size comparisons between DAPT and TAPT.

100, 300, and 300, respectively, while for dialog, email, and social media domains, the maximum training samples for them are 14732, 14436, and 60354, respectively, and we select 300 samples for each domain to construct a low-resource setting. We truncate the input documents into 1024 tokens due to the limitation of the maximum input length for BART. For all the experiments, we use the BART-base version to implement our models. We use a mini-batch size of 4 with a gradient accumulation for 10 iterations. We use Adam optimizer with momentum $\beta_1 = 0.9$, $\beta_2 = 0.998$ and noam decay with warm up steps of 1000. In the decoding stage, we use beam search with a beam size of 4. The decoding process will not stop until an end-of-sequence (EOS) token is emitted or the length of the generated summary reaches to 256 tokens. As for the hyperparameters of RecAdam, we select the best $t0$ and k in $\{500, 600, 700, 800, 900, 1,000\}$ and $\{1e-2, 1e-3, 1e-4, 1e-5, 1e-6\}$, respectively, for the annealing coefficient $\lambda(t)$ (Eq. 2).

Baseline As our baseline, we use an off-the-shelf BART model (Lewis et al., 2019) and perform supervised fine-tuning of its parameters for the summarization task in each domain. BART serves as a good baseline since it provides the state-of-the-art performance in the summarization task. And, as a single generative language model, it can be easily adapted to different target domains.

Evaluation Metrics We use ROUGE (Lin and Hovy, 2003) to measure the quality of the summary produced in our experiments. Following the previous work (Nema et al., 2017), we report ROUGE F1 (ROUGE-1) on the AdaptSum dataset.[5]

6 Results & Analysis

6.1 Main Results

From Table 2, we can see that SDPT is able to generally improve the summarization performance of the fine-tuning method for all domains. This is because SDPT teaches the model how to do the task using large numbers of annotated examples, which enables the model to adapt to target domains faster than the fine-tuning method, and SDPT is able to outperform both DAPT and TAPT in terms of the averaged ROUGE-1 score. The enormous unlabeled corpus makes DAPT quite effective in certain domains, such as email, debate and social media, with close to or more than 2 ROUGE-1 scores improvements over the fine-tuning baseline. As we can see from Table 3, although TAPT uses a far smaller pre-training corpus than DAPT, the performance of TAPT is on par with that of DAPT, which accords with the results in Gururangan et al. (2020), where the experiments are conducted for domain adaptation in classification tasks. Additionally, adding RecAdam into the second phase of pre-training can generally further boost the adaptation performance for SDPT and TAPT, while it only boost the performance on the movie review and science domains for DAPT. We conjecture that a relatively large corpus can potentially weaken the effectiveness of RecAdam, and we observe that

[5] We use `pyrouge` to compute all ROUGE scores, with parameters "-c 95 -2 4 -U -r 1000 -n 4 -w 1.2 -a". The full results of all the models with ROUGE-2 and ROUGE-L are reported in the Appendix.

Domains	DAPT Corpus	TAPT Corpus
Dialog	37.56 (-7.04)	44.60
Email	51.87 (-5.93)	57.80
Movie R.	46.63 (**-14.59**)	61.22
Debate	53.49 (-8.99)	62.48
Social M.	48.10 (-3.82)	51.92
Science	36.94 (**-20.90**)	57.84
Average	45.44 (-10.54)	55.98

Table 4: Vocabulary overlaps (%) between the pre-training corpus (for DAPT or TAPT) and the validation set of the summarization task for each domain. The numbers in the brackets denote the vocabulary overlap differences between the two pre-training corpora, and the bold numbers denote the large discrepancies.

the corpus used for DAPT is comparably small for movie review and science domains, and the number of data samples for XSum (204k) is also much smaller than those of DAPT corpora in many domains (e.g., email), which have more than 1M sentences. According to Eq. 1, extensive training data could result in a comparatively large $Loss_s$ (the model's parameters tend to be greatly modified) which lead to an unstable loss and a negative effect to the pre-training process. In addition, we find that RecAdam is originally shown to be effective at fine-tuning to the downstream GLUE tasks (Chen et al., 2020), the sizes of which are much smaller than the datasets used for SDPT and DAPT.

6.2 How Pre-training Data Affects DAPT

According to prior experiments on domain adaptation for classification or classification-based tasks (Beltagy et al., 2019; Lee et al., 2020; Gururangan et al., 2020), DAPT improves the performance for all domains on the fine-tuning baseline. However, as we can see from Table 2, DAPT causes the performance to drop for the movie review and science domains in the summarization task, while TAPT boosts the performance for all the domains. To further investigate the reasons, we aim to analyze the similarity (e.g., vocabulary overlap) between the pre-training corpus for DAPT and the summarization task in the target domain, which we represent with the target domain validation set of the summarization task to represent. We notice that it is difficult to justify how much overlap is large enough for DAPT to be considered as effective. Hence, we add the TAPT corpus, which is directly related to the target domain's summarization task, as an upper bound for the comparison.

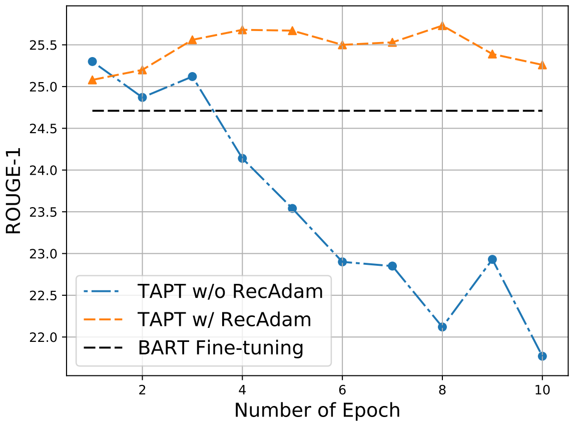


Figure 2: TAPT performance over different pre-training epoch numbers in the email domain in terms of using and not using RecAdam.

Table 4 illustrates the vocabulary overlaps for DAPT and TAPT for each domain.[6] We find large discrepancies between the DAPT corpus and TAPT corpus on the movie review and science domains, which indicates that the domain-related corpora in these two domains are not quite related to the task domains, and pre-training on a domain-unrelated or less related corpus can lead to a performance drop compared to the fine-tuning method. Given that the corpus construction is done by looking for the domain-related sources (as mentioned in Section 3), the experimental results point out that collecting a domain-related corpus for DAPT in the summarization task is not straightforward. *Thus, we leave exploring how to construct an effective corpus for DAPT for future work.*

6.3 Catastrophic Forgetting Issue

We speculate that the second phase of pre-training will result in the catastrophic forgetting for the pre-trained model, which could hurt the adaptation performance. Figure 2 illustrates that the performance of TAPT without RecAdam keeps dropping as the pre-training continues, and it starts to perform worse than the fine-tuning method after three epochs' pre-training, while the performance of TAPT with RecAdam remains stable at around a 25.5 ROUGE-1 score. We conjecture that excessive pre-training makes the pre-trained model overfit to the pre-training data and partially lose its language

[6] To ensure the comparison between DAPT and TAPT is fair, we sample partial data from the DAPT corpus to make its size comparable to the TAPT corpus and create vocabularies for each based on the top 5K most frequent words (excluding stopwords). The vocabulary for the validation set of the summarization task is also created in the same way.

	Source Domains		**Target Domains**					
	XSum	**CNN/DM**	**Dialog**	**Email**	**Movie R.**	**Debate**	**Social M.**	**Science**
Document	354.16	676.03	91.64	124.47	2112.97	196.69	229.31	633.03
Summary	21.13	57.91	20.28	4.10	21.28	11.07	6.31	150.01

Table 5: Averaged length of the input documents and output summaries for the source and target domains.

Domains	**BART**	**SDPT**	**DAPT**	**SDPT+DAPT**
Dialog	39.95	**42.84**	41.22	42.27
Email	24.71	25.16	**26.50**	23.71
Movie R.	25.13	**25.45**	24.25	22.20
Debate	24.48	25.61	**26.71**	25.16
Social M.	21.76	22.43	**22.95**	22.03
Science	72.76	**73.09**	71.88	71.56
Average	34.80	**35.76**	35.59	34.49

Table 6: ROUGE-1 results for SDPT+DAPT compared to the SDPT, DAPT and BART fine-tuning.

Domains	**BART**	**SDPT (XSum)**	**SDPT (CNN)**
Dialog	39.95	42.84	43.13
Email	24.71	25.16	23.81
Movie R.	25.13	25.45	24.51
Debate	24.48	25.61	23.98
Social M.	21.76	22.43	22.56
Science	72.76	73.09	72.41
Average	34.80	35.76	35.07

Table 7: ROUGE-1 results for SDPT based on the XSum and CNN/DM (denoted as CNN in the table) datasets.

understanding and generation ability. However, the model is required to possess both language ability and domain knowledge for better performance in the domain adaptation task. RecAdam helps the pre-trained model preserve its original language ability while continuing pre-training on a new corpus, which boosts the effectiveness of pre-training. However, as we can see from Table 2, RecAdam fails to improve the performance on DAPT using large corpora. We speculate that the catastrophic forgetting issue does not do much harm to the performance of DAPT because pre-training on the large corpus enables the pre-trained model to possess a good language understanding ability in the target domain even though it could lead to partial forgetting in previous domains, and RecAdam makes DAPT stay somewhere in the middle (not forgetting much the previous learned knowledge, but not learning well in the target domain, either). *It indicates that more advanced learning methods are needed for coping with the second pre-training phase on a large corpus.*

6.4 Incorporating SDPT and DAPT

Intuitively, incorporating both the summarization task and target domain knowledge into the pre-trained model could further boost the domain adaptation performance in the summarization task. Therefore, we propose to combine SDPT and DAPT in the second pre-training step. Since SDPT and DAPT use different objective functions, jointly learning these two tasks will make BART confused about what to generate (summarization or sentence reconstruction) given the input sequences. To cope with this issue, we use two BART models (one for SDPT and one for DAPT) and share their encoders in this joint pre-training process to learn the knowledge from both the task and domain. Then, we use the BART model for SDPT to fine-tune to the summarization task in the target domain.

As shown in Table 6, the experimental results are contradictory to the intuition. We find that SDPT+DAPT can not further improve upon the performance of SDPT and DAPT. For the dialog and social media domains, the performances of SDPT+DAPT stay between those of SDPT and DAPT, while for the science, movie review and email domains, the performances of SDPT+DAPT are even lower than that of the BART fine-tuning. We conjecture that SDPT and DAPT are two completely different tasks, and jointly pre-training based on them could confuse the model about the knowledge that it learns. *However, integrating the task and domain knowledge is still a promising direction for domain adaptation. We leave how to incorporate SDPT and DAPT for future work.*

6.5 Different Source Domain Data for SDPT

To explore how different source domain data can affect the performance of SDPT, we use another News domain dataset, CNN/Daily Mail (DM) dataset (Hermann et al., 2015; Nallapati et al., 2016), as the labeled summarization data for SDPT. As we can see from Table 7, SDPT based on CNN/DM only achieves marginal improvements upon the BART fine-tuning baseline in terms of

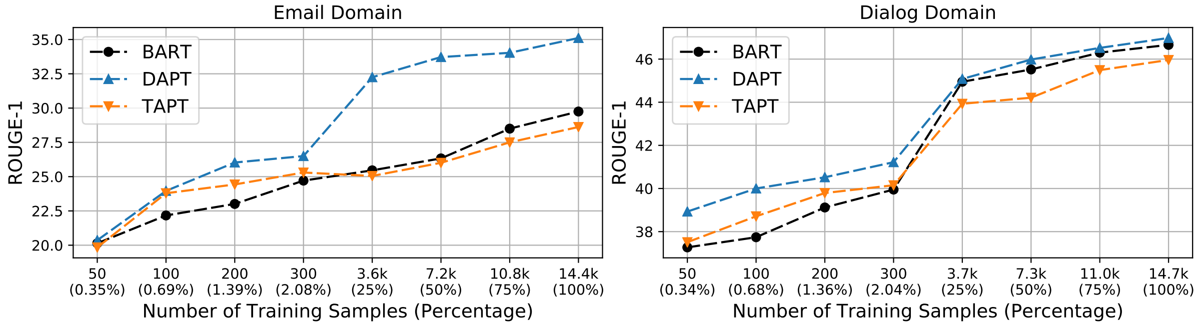


Figure 3: ROUGE-1 results of BART fine-tuning, DAPT and SDPT over different numbers of training data for email (left) and dialog (right) domains. We consider both low-resource settings (50, 100, 200 and 300 (∼2%) samples), medium-resource settings (25% and 50% samples), and high-resource settings (75% and 100% samples).

the averaged score, and for all the domains, it generally performs worse or similar compared to SDPT based on XSum. Since both of them are from the News domain but the number of training samples in CNN/DM (287k) is higher than that in XSum (204k), pre-training on CNN/DM should have achieved better performance than pre-training on XSum. To further analyze the reason, we calculate the averaged length of input documents and output summaries for the source and target domains. From Table 5, we find that the averaged length of XSum is much shorter than that of CNN/DM in terms of both document and summary, and surprisingly, SDPT based on XSum can outperform SDPT based on CNN/DM in domains with short length document and summary (e.g., debate and email) as well as the domains with long length document or summary (e.g., movie review and science). Hence, we conjecture that pre-training with relatively short document and summary is more effective for SDPT. Another reason can be attributed to the fact that the summaries of the CNN/DM tend to copy the content in the input documents, while XSum has larger amounts of novel tokens in the summaries. Therefore, we conjecture that XSum enables model learn a more powerful summarization ability, which helps it to better adapt to low-resource target domains. *We leave investigating the effectiveness of different source domain datasets in SDPT for future work.*

6.6 Performance vs. Training Sample Size

We investigate how well models perform in an extremely low-resource scenario (e.g., 50 training samples) and the performance discrepancies among different levels of resources. The performance over different numbers of training samples is illustrated in Figure 3. We find that BART fine-tuning with the 25% data samples significantly outperforms that with ∼2% data samples in the dialog domain, but such improvements are not remarkable in the email domain. We conjecture that the input and output lengths for the email domain are relatively short compared to the dialog domain (according to Table 5), making the domain adaptation easier.

Interestingly, DAPT outperforms other models in the medium-resource and high-resource settings in the email domain but not in the dialog domain. We speculate the reasons are twofold. First, based on the vocabulary overlaps from Table 4, the email corpus is more effective for DAPT than the dialog domain. Second, email corpus is much larger than the dialog corpus from Table 3. However, the performance of DAPT using a high-quality corpus will be still limited by the low-resource scenario, and it needs large enough training samples to achieve remarkable improvements. Moreover, the performance of TAPT is better than BART fine-tuning in the low-resource setting, while it becomes worse in the medium-resource and high-resource settings. We conjecture that training with more data will aggravate the catastrophic forgetting caused by TAPT, which leads to the worse performance.

Surprisingly, the performance of DAPT with medium-resource is close to that with high-resource, which can be attributed to the combination of the powerful adaptation ability of the large pre-trained generative model and the effectiveness of the second phase of pre-training. However, there is still a large performance gap for all the models between the low-resource and high-resource settings and all the models perform badly when there is only 50 training samples, *which highlights the needs for more advanced domain adaptation models for the summarization task.*

7 Conclusion and Future Work

In this paper, we present AdaptSum, the first benchmark to simulate the low-resource setting for the abstractive summarization task with a combination of existing datasets across six diverse domains. We systematically study three different methods for a second phase of pre-training (i.e., SDPT, DAPT and TAPT), and propose to leverage RecAdam to alleviate the catastrophic forgetting issue caused by the continuing pre-training. Experiments show that SDPT and TAPT can generally improve on the performance of the fine-tuning method, while the effectiveness of DAPT depends on the similarity between the pre-training data and the target domain task data, which is different from the insights into DAPT for classification tasks. Further analysis illustrates that RecAdam successfully alleviates the catastrophic forgetting issue for TAPT and further boost its performance.

Finally, our work highlights several research challenges in low-resource domain adaptation for the abstractive summarization task: (1) How to construct an effective corpus for DAPT; (2) How to better cope with the catastrophic forgetting issue for the second pre-training phase on a large corpus; (3) How to effectively integrate the task and domain knowledge (i.e., incorporate SDPT and DAPT); (4) How to choose better source domain datasets for conducting SDPT; (5) How to build a more powerful domain adaptation models for the extremely low-resource summarization task. We hope that the proposed dataset and the highlighted research directions will accelerate the studies in this area.

Acknowledgments

We want to thank the anonymous reviewers for their constructive feedback. This work is partially funded by ITF/319/16FP, ITS/353/19FP and MRP/055/18 of the Innovation Technology Commission, the Hong Kong SAR Government.

References

Yamen Ajjour, Henning Wachsmuth, Johannes Kiesel, Martin Potthast, Matthias Hagen, and Benno Stein. 2019. Data acquisition for argument search: The args. me corpus. In *Joint German/Austrian Conference on Artificial Intelligence (Künstliche Intelligenz)*, pages 48–59. Springer.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. 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 3606–3611.

Steven Bird, Robert Dale, Bonnie J. Dorr, Bryan Gibson, Mark T. Joseph, Min-Yen Kan, Dongwon Lee, Brett Powley, Dragomir R. Radev, and Yee Fan Tan. 2008. The ACL Anthology Reference Corpus: A Reference Dataset for Bibliographic Research in Computational Linguistics. In *Proc. of the 6th International Conference on Language Resources and Evaluation Conference (LREC'08)*, pages 1755–1759.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th annual meeting of the association of computational linguistics*, pages 440–447.

Sanyuan Chen, Yutai Hou, Yiming Cui, Wanxiang Che, Ting Liu, and Xiangzhan Yu. 2020. Recall and learn: Fine-tuning deep pretrained language models with less forgetting. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7870–7881, Online. Association for Computational Linguistics.

Yen-Chun Chen and Mohit Bansal. 2018. Fast abstractive summarization with reinforce-selected sentence rewriting. *arXiv preprint arXiv:1805.11080*.

Hal Daumé III. 2009. Frustratingly easy domain adaptation. *arXiv preprint arXiv:0907.1815*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of wikipedia: Knowledge-powered conversational agents. In *International Conference on Learning Representations*.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. In *Advances in Neural Information Processing Systems*, pages 13063–13075.

George Foster, Cyril Goutte, and Roland Kuhn. 2010. Discriminative instance weighting for domain adaptation in statistical machine translation. In *Proceedings of the 2010 conference on empirical methods in natural language processing*, pages 451–459.

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

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.

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 ACL*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in neural information processing systems*, pages 1693–1701.

Xinyu Hua and Lu Wang. 2017. A pilot study of domain adaptation effect for neural abstractive summarization. *arXiv preprint arXiv:1707.07062*.

Shruti Jadon. 2020. An overview of deep learning architectures in few-shot learning domain. *arXiv preprint arXiv:2008.06365*.

Chen Jia, Xiaobo Liang, and Yue Zhang. 2019. Cross-domain ner using cross-domain language modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2464–2474.

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.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR (Poster)*.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

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

Chin-Yew Lin and Eduard Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 150–157.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. *arXiv preprint arXiv:1908.08345*.

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

Zihan Liu, Jamin Shin, Yan Xu, Genta Indra Winata, Peng Xu, Andrea Madotto, and Pascale Fung. 2019b. Zero-shot cross-lingual dialogue systems with transferable latent variables. 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 1297–1303.

Zihan Liu, Genta Indra Winata, and Pascale Fung. 2020a. Zero-resource cross-domain named entity recognition. In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 1–6.

Zihan Liu, Genta Indra Winata, Zhaojiang Lin, Peng Xu, and Pascale Fung. 2020b. Attention-informed mixed-language training for zero-shot cross-lingual task-oriented dialogue systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8433–8440.

Zihan Liu, Genta Indra Winata, Peng Xu, and Pascale Fung. 2020c. Coach: A coarse-to-fine approach for cross-domain slot filling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 19–25, Online. Association for Computational Linguistics.

Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. 2020d. Crossner: Evaluating cross-domain named entity recognition. *arXiv preprint arXiv:2012.04373*.

David Lopez-Paz and Marc'Aurelio Ranzato. 2017. Gradient episodic memory for continual learning. In *Advances in neural information processing systems*, pages 6467–6476.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150.

Ahmed Magooda and Diane Litman. 2020. Abstractive summarization for low resource data using domain transfer and data synthesis. *arXiv preprint arXiv:2002.03407*.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. 2008. Domain adaptation with multiple sources. *Advances in neural information processing systems*, 21:1041–1048.

Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*.

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.

Preksha Nema, Mitesh M Khapra, Anirban Laha, and Balaraman Ravindran. 2017. Diversity driven attention model for query-based abstractive summarization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1063–1072.

Romain Paulus, Caiming Xiong, and Richard Socher. 2017. A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304*.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *International Conference on Learning Representations*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

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

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5370–5381.

Oana Sandu, Giuseppe Carenini, Gabriel Murray, and Raymond Ng. 2010. Domain adaptation to summarize human conversations. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*, pages 16–22.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.

Dan Su, Yan Xu, Tiezheng Yu, Farhad Bin Siddique, Elham J Barezi, and Pascale Fung. 2020. Caire-covid: A question answering and multi-document summarization system for covid-19 research. *arXiv preprint arXiv:2005.03975*.

Baochen Sun, Jiashi Feng, and Kate Saenko. 2016. Return of frustratingly easy domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. *arXiv preprint arXiv:1601.04811*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355.

Danqing Wang, Pengfei Liu, Ming Zhong, Jie Fu, Xipeng Qiu, and Xuanjing Huang. 2019. Exploring domain shift in extractive text summarization. *arXiv preprint arXiv:1908.11664*.

Lu Wang and Claire Cardie. 2013. Domain-independent abstract generation for focused meeting summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1395–1405.

Lu Wang and Wang Ling. 2016. Neural network-based abstract generation for opinions and arguments. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 47–57.

Genta Indra Winata, Samuel Cahyawijaya, Zihan Liu, Zhaojiang Lin, Andrea Madotto, Peng Xu, and Pascale Fung. 2020. Learning fast adaptation on cross-accented speech recognition. *arXiv preprint arXiv:2003.01901*.

Yu Yan, Weizhen Qi, Yeyun Gong, Dayiheng Liu, Nan Duan, Jiusheng Chen, Ruofei Zhang, and Ming Zhou. 2020. Prophetnet: Predicting future n-gram for sequence-to-sequence pre-training. *arXiv preprint arXiv:2001.04063*.

Michihiro Yasunaga, Jungo Kasai, Rui Zhang, Alexander R Fabbri, Irene Li, Dan Friedman, and Dragomir R Radev. 2019. Scisummnet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7386–7393.

Wenpeng Yin. 2020. Meta-learning for few-shot natural language processing: A survey. *arXiv preprint arXiv:2007.09604*.

Tiezheng Yu, Dan Su, Wenliang Dai, and Pascale Fung. 2020. Dimsum@ laysumm 20: Bart-based approach for scientific document summarization. *arXiv preprint arXiv:2010.09252*.

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.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2204–2213.

A Full Results of All Models

The full results of all models are shown in Table 8.

B Training Details

Our model contains $\sim$139.4 million parameters and we train all models on one GTX 1080 Ti. We train all the models for 50 epochs in around three hours. We manually tune the hyperparameter values.

Domains	ROUGE Scores	BART Fine-tuning	SDPT	SDPT w/ RecAdam	DAPT	DAPT w/ RecAdam	TAPT	TAPT w/ RecAdam
Dialog	ROUGE-1 F1	39.95	42.84	**45.23**	41.22	40.05	40.15	41.34
	ROUGE-2 F1	17.50	17.51	**19.43**	17.88	17.62	16.99	17.88
	ROUGE-L F1	31.64	33.79	**35.37**	32.40	31.36	31.21	32.31
Email	ROUGE-1 F1	24.71	25.16	**26.97**	26.50	25.66	25.30	25.73
	ROUGE-2 F1	11.71	12.2	**13.44**	13.14	12.89	12.03	12.69
	ROUGE-L F1	24.15	24.28	**25.98**	25.61	25.14	24.63	25.32
Movie R.	ROUGE-1 F1	25.13	25.45	**26.06**	24.25	25.78	25.27	25.65
	ROUGE-2 F1	9.22	9.49	**10.27**	9.06	9.84	9.24	9.13
	ROUGE-L F1	20.04	20.11	**20.91**	19.56	20.69	20.09	20.45
Debate	ROUGE-1 F1	24.48	25.61	25.17	**26.71**	25.01	24.59	24.70
	ROUGE-2 F1	8.21	8.48	8.38	**9.14**	8.42	8.13	8.43
	ROUGE-L F1	21.96	22.86	22.39	**23.64**	22.17	22.04	22.25
Social M.	ROUGE-1 F1	21.76	22.43	**23.25**	22.95	21.51	22.81	23.01
	ROUGE-2 F1	8.11	9.06	9.01	**9.66**	8.25	8.96	8.49
	ROUGE-L F1	21.03	21.03	**22.18**	21.93	20.69	22.06	21.95
Science	ROUGE-1 F1	72.76	**73.09**	72.60	71.88	72.23	73.08	72.80
	ROUGE-2 F1	64.66	**65.15**	63.79	63.73	63.32	65.04	64.26
	ROUGE-L F1	68.40	68.62	68.06	67.34	67.62	**68.81**	68.41

Table 8: Full results of all models.