Separating Context and Pattern: Learning Disentangled Sentence Representations for Low-Resource Extractive Summarization

Ruifeng Yuan[1], **Shichao Sun**[1], **Zili Wang**[2], **Ziqiang Cao**[3], **Wenjie Li**[1]
[1]The Hong Kong Polytechnic University, [2]Xiaohongshu Inc, [3]Soochow University
csryuan@comp.polyu.edu.hk, bruce.sun@connect.polyu.hk
wangzili@xiaohongshu.com, zqcao@suda.edu.cn, cswjli@comp.polyu.edu.hk

Abstract

Extractive summarization aims to select a set of salient sentences from the source document to form a summary. Context information has been considered one of the key factors for this task. Meanwhile, there also exist other pattern factors that can identify sentence importance, such as sentence position or certain n-gram tokens. However, such pattern information is only effective in specific datasets or domains and can not be generalized like the context information when there only exists limited data. In this case, current extractive summarization models may suffer from a performance drop when transferring to a new dataset. In this paper, we attempt to apply disentangled representation learning on extractive summarization, and separate the two key factors for the task, context and pattern, for a better generalization ability in the low-resource setting. To achieve this, we propose two groups of losses for encoding and disentangling sentence representations into context representations and pattern representations. In this case, we can either use only the context information in the zero-shot setting or fine-tune the pattern information in the few-shot setting. Experimental results on three summarization datasets from different domains show the effectiveness of our proposed approach.

1 Introduction

The glob of text summarization is to generate a concise highlight of a source document, which covers the crucial information conveyed in the source text. In this paper, we focus on extractive summarization. It aims to produce summaries by selecting and combining the salient sentences that are directly taken from the source text.

It is widely agreed that extractive summarization is mainly based on context information to select the important sentences. Meanwhile, there also exist other factors that can be used to identify these sentences, such as sentence position or certain n-gram tokens. As shown in Figure 1, in the news summarization dataset, lead sentences always have a much higher possibility to become crucial sentences. Meanwhile, Table 1 shows that sentences with certain n-gram tokens like "in this paper" or "we find that" are also considered to be important in science paper summarization. Here, we collectively called these factors pattern information, since they are context-independent and can decide the sentence importance solely by themselves. However, as we displayed in Figure 1 and Table 1, pattern information varies from dataset to dataset. In this case, such information is only effective in its corresponding dataset or domain and can not be generalized like the context information. Although both context information and pattern information are

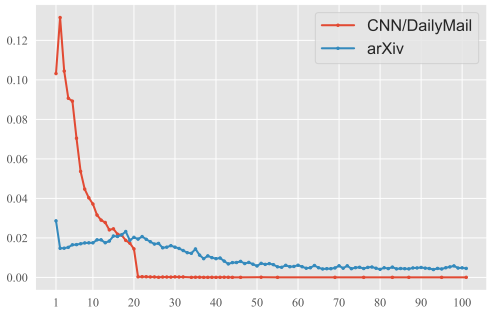

Figure 1: Comparison of position distribution of oracle sentences in news summarization dataset CNN/DailyMail and science paper summarization dataset arXiv. The X-axis refers to 1 to 100 sentence position and the Y-axis represents its proportion.

CNNDM	Num	arXiv	Num
(cnn) –	21k	in this paper	11k
according to the	3.5k	as a function	6.4k
the first time	2.4k	in the case	4.8k
the end of	1.3k	we find that	3.7k

Table 1: Examples about the high-frequency n-grams in oracle sentences from CNN/DailyMail and arXiv.

Findings of the Association for Computational Linguistics: ACL 2023, pages 7575–7586
July 9-14, 2023

crucial for the task, it is hard to tell whether the improvement of the current extractive summarization models stems from a better understanding of the context information or overfitting the pattern information on specific data. Hence, the existing models may fail to achieve good performance when transferring to other domains or datasets with limited data due to the intermingling of domain-specific pattern information.

In this paper, we aim to apply disentangled representation learning to extractive summarization, and separate the two key factors for the task, context information and pattern information, for a better generalization ability in low-resource settings (zero-shot and few-shot). Our model is built on a pretraining-based extractive summarization model (Liu and Lapata, 2019) that uses a BERT to encode each sentence with its context to the latent representation. We would like the latent representation to be disentangled with respect to the context and pattern information. Following the previous works (John et al., 2018; Cheng et al., 2020), we combine the multitask objectives and adversarial objectives/mutual information (MI) minimizing objectives to accomplish this. The multitask objectives aim to encourage the two latent spaces to learn its corresponding information. For the context information, we propose to approximate it by predicting the high-frequency non-stop word appearing in a sentence and its context. For the pattern information, we divide it into two parts: the position pattern feature and the n-gram pattern feature. The former one can be transferred into a sentence position predicting problem, while the latter one is approximated by predicting whether the target sentence contains any high-frequency n-gram patterns. Then we try two commonly used disentangled representation learning approaches, adversarial objectives/MI minimizing objectives, to further ensure the independence between the two latent spaces.

After the model is trained on a source dataset, it can be transferred to a target dataset for low-resource extractive summarization. In the zero-shot setting, we only utilize the context representation to do the extractive summarization. In the few-shot setting, we choose to fine-tune the pattern-related parameters with a few training instances to automatically select useful patterns for the target dataset.

To evaluate our proposed model, we conduct the experiments on three datasets from different domains: CNN/DaliyMail from the news summarization domain, arXiv from the science article summarization domain, and QMSum from the dialogue summarization domain. These experiments suggest the effectiveness of our model by disentangling context and pattern information.

2 Related Work

2.1 Text Summarization

Extractive summarization is an important sub-topic for text summarization. Early works (Nallapati et al., 2017; Narayan et al., 2018; Zhou et al., 2018; Zhang et al., 2018) formulated it as a sentence binary classification problem and further extend it with different techniques. With the development of the pretrained model, using a transformer-based pretrained model as encoder (Liu and Lapata, 2019; Bae et al., 2019; Zhang et al., 2019) leads to a huge improvement in the task. Recently, MATCHSUM (Zhong et al., 2020) has achieved a state-of-the-art performance by combining contrastive learning with extractive summarization. These models mainly focus on improving the performance on a certain dataset or domain. Research on low-resource text summarization is also increasing. AdaptSum (Yu et al., 2021) propose a pre-train and then fine-tune strategy for low-resource domain adaptation for abstractive summarization. Other researchers (Fabbri et al., 2020) present a similar idea but further enhance it with a data augmentation method using the large corpus from Wikipedia. (Zhao et al., 2022) combines domain words and a prompt-based language model to achieve zero-shot domain adaption in dialogue abstractive summarization. In this work, we aim to explore the low-resource extractive summarization by disentangling context and pattern information.

2.2 Disentanglement Representation Learning

Disentanglement representation has first been explored in computer vision to disentangle features such as color or rotation. Recently, a growing amount of work has been proposed to investigate learning disentangled representations in NLP tasks. Early works (Hu et al., 2017; Shen et al., 2017; John et al., 2018) follow a similar idea, and applied disentanglement representation learning on style/sentiment transferring. Later, researchers further extend its application to different topics such as cross-lingual transfer (Wu et al., 2022), negation and uncertainty learning (Vasilakes et al., 2022),

and fair classification(Park et al., 2021). Generally, there are mainly three types of approaches for disentanglement representation learning. A common approach (John et al., 2018) is to add an adversary that competes against the encoder trying to avoid learning certain types of attribute. Another approach (Cheng et al., 2020; Colombo et al., 2021) is to adopt the mutual information theory, and attempt to minimize the mutual information upper bound between two disentangle representations. Recently, some researchers (Colombo et al., 2022) propose a simpler approach by adding a set of regulizers to achieve disentanglement representation learning. Similar to cross-lingual transfer, in this work, we also aim to adopt disentanglement representation learning to domain transferring, but in the context of extractive summarization.

3 Model

3.1 Problem Statement

In this work, we disentangle the sentence representation for extractive summarization into two parts: context representation and pattern representation. To achieve this, we need to satisfy the following requirements for an effective disentanglement.

- The context and pattern representation need to have the ability to predict sentence importance and contribute to the extractive summarization.
- The context and pattern representation should be predictive of the corresponding ground-truth information. For example, the pattern representation of a sentence can predict its pattern feature such as its position.
- The context and pattern representation should lie in independent vector space, and one representation can not predict the corresponding ground-truth information of the other one.

3.2 Extractive Summarization Model

Given an input document containing n sentences $x = \{s_1, s_2, .., s_n\}$, we adopt a BERT to generate contextualized representations for each sentence. Since the output of BERT is grounded to tokens, we use a similar strategy with (Liu and Lapata, 2019) to modify the input sequence of BERT. We insert a [cls] token at the beginning of each sentence and use the embedding of the [cls] token to represent its corresponding sentence. Considering our glob is to disentangle it to context and pattern representation, we add two additional multilayer perceptrons (MLP) that map the sentence representations generated by BERT to context representations c and pattern representations p. Here, we collectively called the BERT and two MLP mappers encoder E. Then a sigmoid classifier F_{ext} takes the concatenation of both representations as input to predict a score y_i^e for sentence s_i, and the loss of the whole model is the binary classification loss of y_i^e against gold label t_i^e. Note that the gold label refers to the one-hot distribution of the oracle sentences (the sentence set that has the highest similarity with the reference summary). The loss is shown in the following:

$$y_i^e = F_{ext}(c_i; p_i) \tag{1}$$

$$l_{ext} = -\frac{1}{n}\sum_{i}^{n} t_i^e log(y_i^e) + (1-t_i^e) log(1-y_i^e) \tag{2}$$

This classification loss serves as our primary training objective for extractive summarization. Meanwhile, to better utilize the context representation and pattern representation in the low-resource setting, we expect the two disentangled representations can do extractive summarization independently. Hence, we add two similar classifiers that directly take context representation or pattern representation as input, and their losses are denoted as $l_{ext(c)}$ and $l_{ext(p)}$. Note that the gradients of the two classifiers are detached from the main model.

3.3 Learning Context Representation

The context representation c is expected to do extractive summarization using the context information. In addition to the extractive summarization loss, we add a multitask objective to ensure the context information is contained in it. The question that lies ahead is to define what "context" actually refers to. A widely accepted idea is that the effective context information in extractive summarization is salient words/phrases that repeat multiple times in the context. Inspired by this, given a sentence s_i, we propose to approximate the context information by predicting the non-stop words existing in both s_i and its adjacent sentences. The distribution of these words on the vocabulary is considered as the context feature t_i^c for s_i.

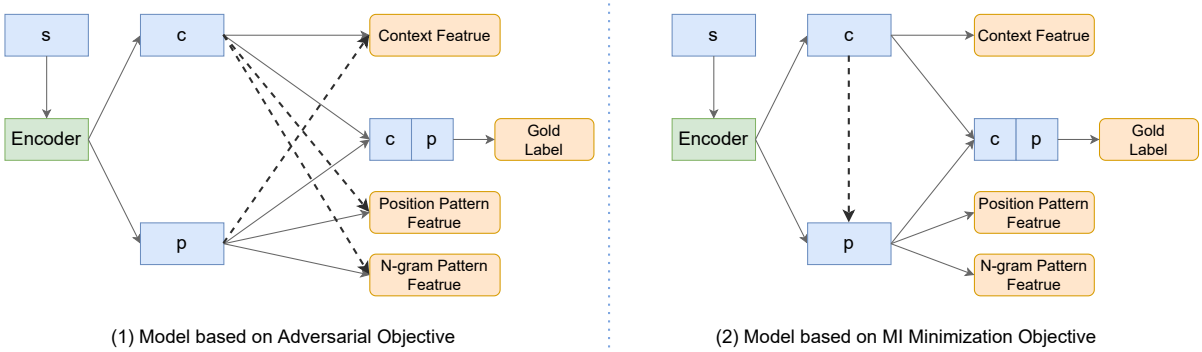

Figure 2: The framework of our proposed model. Part (1) shows the model based on the adversarial objective, while part (2) displays the one based on the MI minimization objective. The blue blocks refer to different sentence representations, the green blocks stand for the model components and the yellow blocks represent the target features. The solid lines represent normal classification loss, and the dashed lines stand for the discriminator loss plus the adversary loss.

We build a two-layer MLP classifier $F_{mul(c)}$ on the context representation c to predict the context feature, and the classifier is trained with cross-entropy loss against the ground-truth distribution:

$$l_{mul(c)} = -\frac{1}{n}\sum_{i}^{n}\sum_{j\in voc} t_{ij}^{c} log(y_{ij}^{c}) \quad (3)$$

where the voc stands for the vocabulary and $y_i^c = F_{mul(c)}(c_i)$ is the predicted context feature.

3.4 Learning Pattern Representation

The pattern representation p needs to predict both sentence importance and pattern-related features. In this paper, we mainly focus on the two types of pattern, position pattern and n-gram pattern, that contribute the most to extractive summarization. Position pattern refers to the position of the sentence in the document, which plays an important role in the news article summarization. We add a multitask objective that predicts the position of a sentence. In this case, the position pattern feature t_i^o is a one-hot vector with a length that is the same as the sentence number. N-gram pattern is another crucial factor that influences sentence importance, which represents the expressions/phrases that are commonly used for summaries. Inspired by (Salkar et al., 2022), We count the frequencies of all n-grams that appear in the oracle sentences and select the top 500 as the n-gram pattern set. The glob of pattern representation is to predict whether a sentence contains any pattern from the pattern set, which is a binary classification problem.

Similarly, we also use two MLP classifiers on the pattern representation p to predict the pattern related feature:

$$l_{mul(p)} = -\frac{1}{n}\sum_{i}^{n} t_i^p log(y_i^p) + (1-t_i^p) log(1-y_i^p) \quad (4)$$

$$l_{mul(o)} = -\frac{1}{n}\sum_{i}^{n}\sum_{j}^{n} t_{ij}^{o} log(y_{ij}^{o}) \quad (5)$$

where $y_i^p = F_{mul(p)}(p_i)$ is the predicted n-gram pattern feature and $y_i^o = F_{mul(o)}(p_i)$ is the predicted position pattern feature.

3.5 Learning Disentangled Representation

Although the multitask objectives assist the model to learn context and pattern information in different latent spaces, they are not effective enough to ensure the independence between c and p. As shown in the Figure 2, we adopt two commonly used objectives for learning disentangled representation in this paper.

Adversarial Objective Considering one representation should be predictive of their corresponding information only, following (John et al., 2018), we add adversarial classifiers that try to predict the information related to the other one on both latent spaces, and the model is forced to structure the latent spaces such that the outputs of these adversarial classifiers are non-predictive. The adversarial objective is composed of two parts. The first part is the adversarial classifiers on each latent space for each type of non-target information. The second part is the adversarial loss aiming to maximize the entropy of the predicted distribution of the adversarial classifiers.

Taking the adversarial objective on the pattern space for example, we train a two-layer MLP classifier, context discriminator $F_{dis(c)}$, to predict whether it contains any context information. One thing that is worth noticing is that the gradients of these classifiers are not back-propagated to the encoder. In this case, the training of the context discriminator will not influence the encoder. Similar to equation (3) and (5), a cross-entropy loss is shown as follow, but with different input and parameters:

$$l_{dis(c)} = -\frac{1}{n}\sum_{i}^{n}\sum_{j\in voc} t_{ij}^{c} log(y_{ij}^{c}) \quad (6)$$

where $y_i^c = F_{dis(c)}(p_i)$ refers to the predicted context feature using pattern representation.

Then an adversarial loss is used to maximize the entropy of the output of context discriminator. Here, we only train the encoder with such adversarial loss and the parameters of the context discriminator are excluded.

$$l_{adv(c)} = -\frac{1}{n}\sum_{i}^{n}\sum_{j\in voc} y_{ij}^{c} log(y_{ij}^{c}) \quad (7)$$

We also impose the n-gram pattern discriminator and position pattern discriminator to disentangle the pattern information from the context space. These two adversarial objectives follow nearly the same way as the mentioned one and their corresponding loss are denoted as $l_{dis(p)}$, $l_{dis(o)}$, $l_{adv(p)}$ and $l_{adv(o)}$.

MI Minimization Objective Mutual information (MI) is a natural measure of the independence between two variables. Inspired by the previous works (Cheng et al., 2020), minimizing the upper-bound estimate of the mutual information (MI) between two latent spaces is an effective way to disentangle them. Following the Contrastive Learning Upper-Bound (CLUB) estimate of the MI (Cheng et al., 2020), we firstly train a neural network M that aims to estimate pattern representation by taking context representation as input:

$$l_{map} = \frac{1}{n}\sum_{i}^{n} kl(M(c_i), p_i) \quad (8)$$

where kl stands for the Kullback–Leibler divergence. Just like the discriminator in the adversarial objective, we fix the parameters of the encoder when we train the neural network M with this loss.

We minimize the Mutual information between the two latent spaces by minimizing the following equation:

$$l_{mi} = \frac{1}{n}\sum_{i}^{n} kl(M(p_i), c_i) - kl(M(p_i), c_k) \quad (9)$$

where k is selected uniformly from indices $\{1, ..., n\}$. Here, the optimization is only performed with parameters of the encoder E.

3.6 Training Strategy

The loss of our model mainly consists of two parts, the losses that update the discriminator (for MI Objective, it is M) and the main loss (all the other losses). In the training process, for each batch, we first optimize the discriminator by $l_{dis(c)}$, $l_{dis(p)}$ and $l_{dis(o)}$ with a weight λ_{dis} (for MI Objective, it is l_{map}), and then optimize the encoder and all other classifiers with the main loss. The main loss L_{all} for our model comprises three types of terms: the extractive summarization objectives, the context/pattern feature learning objectives and adversarial objectives (for MI Objective, it is l_{mi}), given by

$$\begin{aligned} l_{all} = l_{ext} + l_{ext(c)} + l_{ext(p)} + \\ \lambda_{mul} l_{mul(c)} - \lambda_{adv} l_{adv(c)} + \\ \lambda_{mul} l_{mul(p)} - \lambda_{adv} l_{adv(p)} + \\ \lambda_{mul} l_{mul(o)} - \lambda_{adv} l_{adv(o)} \end{aligned} \quad (10)$$

The checkpoint selection strategy and hyperparameter searching are also crucial for model training. Considering the glob of our model is to effectively utilize the context information in the target dataset rather than achieve the best performance on the source dataset, we follow two rules: (1) The disentanglement is successful (based on the training log); (2) We select the checkpoint with the best performance when using context representation on the validation set. In the experiment, the weights are $\lambda_{mul} = 1, \lambda_{adv} = 1, \lambda_{dis} = 3$

3.7 Application in Low-Resource Setting

After we train the model on a source dataset, we can transfer it to a target dataset with limited data. Considering the pattern information in the source dataset may be misleading in a target dataset, we use the context representation to do the extractive summarization in the zero-shot setting. As for the few-shot setting, the data samples from the target

Dataset	Type	Domain	Size	Source length	Target length
QMsum	dialogue	meeting	1257/272/281	9070	70
arXiv	document	science	202914/6436/6440	6030	273
CNN/DailyMail	document	news	287227/13368/11490	766	53

Table 2: The statistics and comparison of the datasets.

dataset provide the model a chance to accomplish a quick adjustment on its pattern information. In this case, we choose to fine-tune the pattern-related parameters with the given samples to select useful patterns for the target dataset.

4 Experiment

4.1 Experiment Details

Dataset: We evaluate our proposed methods in three English datasets from different domains. The detailed information and comparison are shown in Table 2. **arXiv** (Cohan et al., 2018) collects academic articles from arXiv.org as source documents and uses the abstracts of these articles as the target summaries. **QMSum** (Zhong et al., 2021) is one of the benchmark datasets for dialogue summarization. Considering the QMSum dataset contains both data samples for normal text summarization and query-focused summarization, we only use the data samples that contain no query. Meanwhile, the number of training data in QMSum is relatively small, so we only use it for testing. **CNN/DailyMail** (Nallapati et al., 2016) is the classic dataset for news summarization. It is also known for suffering from lead bias, where the summaries that consist of the lead three sentences can achieve a relatively good performance.

Model Details: In this work, we adopt BERT-base as the encoder of our model. Our implementation is based on Transformers from Hugging Face. In the training, the learning rate is set to 2e-5, and the batch size is set to 16. We conduct the validation for every 2000 steps and train the model for a maximum of 30000 steps. We truncate all the input documents to 500 tokens. For the long-input summarization dataset such as arXiv and QMSum, we split the original document into multiple chunks and generate extractive summarization scores for the sentences in each chunk independently. In all experiments, we select 3 sentences for CNN/DM and 6 sentences for arXiv and QMSum. Following previous works, we also adopt the trigram blocking trick during inference.

Evaluation Metric: We adopt Rouge as our evaluation metric (Lin, 2004) including Rouge-1 (R-1), Rouge-2 (R-2), and Rouge-L (R-L) as evaluation metrics. In practice, we use a python wrapper py-rouge to apply the classic Rouge 1.5.5.

To arXiv	R-1	R-2	R-L
Lead*	33.66	8.94	22.19
TextRank*	24.38	10.57	22.18
LexRank*	33.85	10.73	28.99
AdaptSum	36.28	9.17	32.26
Our_adv	**37.03**	**9.64**	**33.03**
Our_mi	36.89	9.44	32.75
BERT(full)	41.04	13.92	36.61
To QMSum	R-1	R-2	R-L
Lead-5*	12.84	1.69	9.17
TextRank*	16.27	2.69	15.41
AdaptSum	26.41	4.67	23.80
Our_adv	**27.27**	**5.11**	**24.91**
Our_mi	26.71	4.49	24.18

Table 3: The results of models trained on CNN/DM in zero-shot setting.

4.2 Comparison

We compare our method with some commonly used baselines and previous state-of-the-art methods designed for low-resource text summarization. There are three types of methods: unsupervised baselines, comparable unsupervised models based on domain transferring or pretraining, and other reference models that are not directly comparable.

Unsupervised Baselines Lead-n aims to select the lead sentences in the document as the summaries, and it always plays an important role in the news summarization dataset that heavily relies on the position pattern information such as CNN/DailyMail. We also show the result of two strong unsupervised baselines TextRank (Mihalcea and Tarau, 2004) and LexRank (Erkan and Radev, 2004).

Comparable Models AdaptSum (Yu et al., 2021) focuses on one-to-one domain adaption in text summarization. It proposes a Source Domain Pre-Training (SDPT) strategy that first fine-tunes a pre-trained model on the source domain and then ap-

To CNN/DM	R-1	R-2	R-L
Lead*	40.49	17.66	36.75
TextRank*	33.85	13.61	30.14
LexRank*	34.68	12.82	31.12
AdaptSum	37.21	15.07	33.64
Our_adv	**38.37**	**15.81**	**34.64**
Our_mi	38.05	15.74	34.37
BERT(full)	42.83	19.82	39.13
To QMSum	R-1	R-2	R-L
Lead-5*	12.84	1.69	9.17
TextRank*	16.27	2.69	15.41
AdaptSum	**28.28**	**4.78**	**25.28**
Our_adv	28.01	4.74	24.94
Our_mi	27.63	4.66	25.13

Table 4: The results of models trained on arXiv in zero-shot setting.

plies it to the target domain. Another research (Fabbri et al., 2020) also proposes a similar method with it and further extends with a data augmentation method. However, this data augmentation method requires the pattern information from the target dataset and is not comparable with our model.

Other Reference Models We display the result of BERTSum (Liu and Lapata, 2019) training on the full target dataset, which can be considered as the upper bound of our model.

4.3 Experiment Results

Zero-shot application We first evaluate the performance of our model in the zero-shot setting in Table 3 and Table 4, where the information of the target dataset is totally unknown. Here, we display the two variants of the model, Our_adv using the adversarial objective and Our_mi adopting the MI minimization objective. Based on the results, we have the following observation. Firstly, Our_adv achieves the best result in most cases. This indicates the effectiveness of context information in the zero-shot setting. Meanwhile, we also observe that Our_mi obtains a lower performance compared to Our_adv. Further investigation of the training process shows that using the MI minimization objective is more difficult to disentangle pattern and context information. We think the reason is that the two types of information are not naturally disentangled and are optimized by the same extractive summarization objectives. In this case, the model requires more clear guidance to achieve the disentanglement.

arXiv to CNN/DM	R-1	R-2	R-L
Both	37.71	15.28	33.98
Context	38.37	15.81	34.64
Pattern	36.65	14.39	32.95

Table 5: The results on CNN/DM when using context/pattern representation.

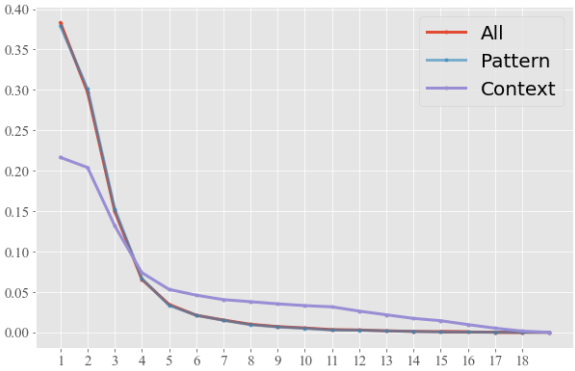

Figure 3: The predicted sentences position distribution on arXiv when using context/pattern representation.

Analysis of context and pattern information To understand the influence of both context and pattern information on the target dataset, we compare the performance of using context representation, using pattern representation, and using both representations in Table 5. Considering the huge gap in the pattern between the two datasets, it is not surprising that using the pattern representation achieves the worst result. Meanwhile, its misleading information also pulls down the results of using both representations. We also display the position distribution of extracted sentences on arXiv using the model trained on CNN/DM in Figure 3. Since CNN/DM is known for its lead bias, the pattern latent space learned on it inevitably tend to select the lead sentences. This trend further dominates the situation when using both representations. As for using context representation alone, the lead bias is relatively weaker.

Few-shot application Directly using the pattern information in an unsuitable dataset leads to a decrease in the model performance. However, this does not mean the pattern representation is completely useless. In the few-shot setting, we can obtain some information from the target dataset and fine-tune the pattern latent space. To simulate this situation, for each target dataset, we build its few-shot version by randomly taking 50 data samples from its original training set and splitting it into 25

	arXiv to CNN/DM			CNN/DM to arXiv		
	Rouge-1	Rouge-2	Rouge-L	Rouge-1	Rouge-2	Rouge-L
BERT	37.36	15.21	33.86	32.55	7.68	28.92
AdaptSum	38.21	15.91	34.60	39.12	11.25	34.78
Our_adv	39.27	16.56	35.47	39.39	11.35	34.97

Table 6: The results on arXiv and CNN/DM in few-shot setting.

arXiv to CNNDM	R-1	R-2	R-L
Our_adv	38.37	15.81	34.64
–adv loss	37.72	15.44	34.05
–aux loss	37.11	14.75	33.44

Table 7: The ablation study in the zero-shot setting.

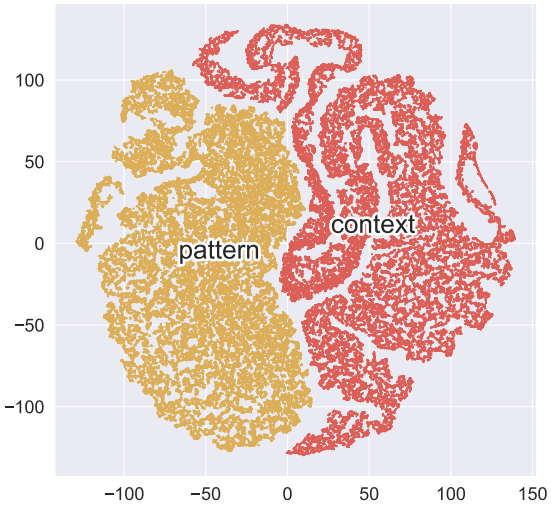

Figure 4: Visualization of context and pattern representations.

training data and 25 validation data. Here, despite our proposed model and AdaptSum, we also show the result of directly fine-tuning a BERTSum model on the limited data. In Table 6, the performance of all models is improved with the help of the limited data, while the gap between Our_adv and AdaptSum still exists. This shows our model is capable of selecting the effective pattern information for the target dataset and preserving its advantages on context information.

Ablation study We further conduct an ablation study. Firstly, we remove the adversary objectives from our model (–adv loss), which means the model can only learn the disentangled representation by approximating context/pattern features. Then we further remove the multitask objectives (–aux loss). In this case, the main difference between this model and the AdaptSum is that our classifier contains more parameters. Here we compare the result of only using context representations in the zero-shot setting. As shown in Table 7, we find that removing the adversary objectives leads to a clear performance drop. This suggests that using the adversary objectives alone is far enough to disentangle the context and pattern information. We also find that the result of model "–aux loss" is similar to the result of AdaptSum in Table 4, which shows the improvement of our model is not brought by the additional parameters.

4.4 Visualization

To have a more direct observation, we visualize the context and pattern representations by using the t-SNE algorithm (Van der Maaten and Hinton, 2008) to reduce them to two dimensions in Figure 4. These representations are taken from 1000 randomly sampled examples from CNN/DM using the model trained on arXiv. Each point refers to a context/pattern representation of a sentence from the source document. The figure shows that the context latent space and pattern latent space are well separated into two parts, which supports the effectiveness of our model in disentangling context and pattern information.

5 Conclusion

In this paper, we propose a novel extractive summarization model that aims to improve the generalization ability in low-resource setting. It disentangles the sentence representation to context and pattern representation and utilize the context information to reduce the influence of domain-specific pattern information during model transferring. The experiment suggests the ability of our model in the disentanglement, and it also supports the claim that the context information tends to have better generalization ability facing the dataset from a different domain. In the future, we plan to extend this idea by learning a more generalized context latent space from multiple summarization datasets.

Limitations

Firstly, we adopt two types of representative pattern information, position pattern, and n-gram pattern, but it does not mean they cover all effective pattern information. In this case, the way to efficiently include all types of pattern information is still an important problem. Secondly, we do not put too much effort into investigating the influence of different feature forms (pattern feature and context feature) for the multitask objectives. Thirdly, due to the limitation of time and paper length, we only evaluate our method in three representative domains. Other domains such as review summarization (Reddit (Völske et al., 2017)) and legislation document summarization (BillSum (Kornilova and Eidelman, 2019)) are also worth exploring.

Ethics Statement

Our experimental datasets, CNN/DailyMail, arXiv, and QMSum, are well-established and publicly available. Datasets construction and annotation are consistent with the intellectual property and privacy rights of the original authors. The scientific artifacts we used are available for research with permissive licenses, including ROUGE and Transformers from HuggingFace. The use of these artifacts is consistent with their intended use. The task of our work is a classic NLP task, text summarization. Considering all the datasets are public available, we think there are no potential risks for this work.

Acknowledgements

The work described in this paper was supported by Research Grants Council of Hong Kong (PolyU/15203617 and PolyU/5210919), National Natural Science Foundation of China (61672445, 62076212, 62106165).

References

Sanghwan Bae, Taeuk Kim, Jihoon Kim, and Sang-goo Lee. 2019. Summary level training of sentence rewriting for abstractive summarization. *arXiv preprint arXiv:1909.08752*.

Pengyu Cheng, Martin Renqiang Min, Dinghan Shen, Christopher Malon, Yizhe Zhang, Yitong Li, and Lawrence Carin. 2020. Improving disentangled text representation learning with information-theoretic guidance. *arXiv preprint arXiv:2006.00693*.

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. *arXiv preprint arXiv:1804.05685*.

Pierre Colombo, Chloe Clavel, and Pablo Piantanida. 2021. A novel estimator of mutual information for learning to disentangle textual representations. *arXiv preprint arXiv:2105.02685*.

Pierre Colombo, Guillaume Staerman, Nathan Noiry, and Pablo Piantanida. 2022. Learning disentangled textual representations via statistical measures of similarity. *arXiv preprint arXiv:2205.03589*.

Günes Erkan and Dragomir R Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, 22:457–479.

Alexander R Fabbri, Simeng Han, Haoyuan Li, Haoran Li, Marjan Ghazvininejad, Shafiq Joty, Dragomir Radev, and Yashar Mehdad. 2020. Improving zero and few-shot abstractive summarization with intermediate fine-tuning and data augmentation. *arXiv preprint arXiv:2010.12836*.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. In *International conference on machine learning*, pages 1587–1596. PMLR.

Vineet John, Lili Mou, Hareesh Bahuleyan, and Olga Vechtomova. 2018. Disentangled representation learning for non-parallel text style transfer. *arXiv preprint arXiv:1808.04339*.

Anastassia Kornilova and Vlad Eidelman. 2019. Billsum: A corpus for automatic summarization of us legislation. *arXiv preprint arXiv:1910.00523*.

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

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

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Thirty-First AAAI Conference on Artificial Intelligence*.

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. Ranking sentences for extractive summarization with reinforcement learning. *arXiv preprint arXiv:1802.08636*.

Sungho Park, Sunhee Hwang, Dohyung Kim, and Hyeran Byun. 2021. Learning disentangled representation for fair facial attribute classification via fairness-aware information alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 2403–2411.

Nikita Salkar, Thomas Trikalinos, Byron C Wallace, and Ani Nenkova. 2022. Self-repetition in abstractive neural summarizers. *arXiv preprint arXiv:2210.08145*.

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. *Advances in neural information processing systems*, 30.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Jake Vasilakes, Chrysoula Zerva, Makoto Miwa, and Sophia Ananiadou. 2022. Learning disentangled representations of negation and uncertainty. *arXiv preprint arXiv:2204.00511*.

Michael Völske, Martin Potthast, Shahbaz Syed, and Benno Stein. 2017. Tl; dr: Mining reddit to learn automatic summarization. In *Proceedings of the Workshop on New Frontiers in Summarization*, pages 59–63.

Shaojuan Wu, Xiaowang Zhang, Deyi Xiong, Shizhan Chen, Zhiqiang Zhuang, Zhiyong Feng, et al. 2022. Learning disentangled semantic representations for zero-shot cross-lingual transfer in multilingual machine reading comprehension. *arXiv preprint arXiv:2204.00996*.

Tiezheng Yu, Zihan Liu, and Pascale Fung. 2021. Adaptsum: Towards low-resource domain adaptation for abstractive summarization. *arXiv preprint arXiv:2103.11332*.

Xingxing Zhang, Mirella Lapata, Furu Wei, and Ming Zhou. 2018. Neural latent extractive document summarization. *arXiv preprint arXiv:1808.07187*.

Xingxing Zhang, Furu Wei, and Ming Zhou. 2019. Hibert: Document level pre-training of hierarchical bidirectional transformers for document summarization. *arXiv preprint arXiv:1905.06566*.

Lulu Zhao, Fujia Zheng, Weihao Zeng, Keqing He, Weiran Xu, Huixing Jiang, Wei Wu, and Yanan Wu. 2022. Domain-oriented prefix-tuning: Towards efficient and generalizable fine-tuning for zero-shot dialogue summarization. *arXiv preprint arXiv:2204.04362*.

Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2020. Extractive summarization as text matching. *arXiv preprint arXiv:2004.08795*.

Ming Zhong, Da Yin, Tao Yu, Ahmad Zaidi, Mutethia Mutuma, Rahul Jha, Ahmed Hassan Awadallah, Asli Celikyilmaz, Yang Liu, Xipeng Qiu, et al. 2021. Qmsum: A new benchmark for query-based multi-domain meeting summarization. *arXiv preprint arXiv:2104.05938*.

Qingyu Zhou, Nan Yang, Furu Wei, Shaohan Huang, Ming Zhou, and Tiejun Zhao. 2018. Neural document summarization by jointly learning to score and select sentences. *arXiv preprint arXiv:1807.02305*.

ACL 2023 Responsible NLP Checklist

A For every submission:

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

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

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

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

B ☑ Did you use or create scientific artifacts?

Section 4.1 the used scientific artifacts are public datasets

☑ B1. Did you cite the creators of artifacts you used?
Section 4.1

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. the used scientific artifacts are public datasets

☑ 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 4.1

☑ 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?
Section Ethics Statement, all the data used in this paper are from public datasets.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Section 4.1

☑ 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.
Section 4.1

C ☑ Did you run computational experiments?

Section 4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 4.1

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?
Section 4.1, Section 3.6

☒ 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?
The experiments are conducted based on single run.

☑ 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 4.1

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

Left blank.

☐ 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.?
No response.

☐ 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)?
Not applicable. Left blank.

☐ 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?
Not applicable. Left blank.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.