A Variational Hierarchical Model for Neural Cross-Lingual Summarization

Yunlong Liang[1*], **Fandong Meng**[2], **Chulun Zhou**[2,3], **Jinan Xu**[1†],
Yufeng Chen[1], **Jinsong Su**[3] and **Jie Zhou**[2]
[1]Beijing Key Lab of Traffic Data Analysis and Mining, Beijing Jiaotong University, China
[2]Pattern Recognition Center, WeChat AI, Tencent Inc, China
[3]School of Informatics, Xiamen University
{yunlongliang,jaxu,chenyf}@bjtu.edu.cn clzhou@stu.xmu.edu.cn
jssu@xmu.edu.cn {fandongmeng,withtomzhou}@tencent.com

Abstract

The goal of the cross-lingual summarization (CLS) is to convert a document in one language (*e.g.*, English) to a summary in another one (*e.g.*, Chinese). Essentially, the CLS task is the combination of machine translation (MT) and monolingual summarization (MS), and thus there exists the hierarchical relationship between MT&MS and CLS. Existing studies on CLS mainly focus on utilizing pipeline methods or jointly training an end-to-end model through an auxiliary MT or MS objective. However, it is very challenging for the model to directly conduct CLS as it requires both the abilities to translate and summarize. To address this issue, we propose a hierarchical model for the CLS task, based on the conditional variational auto-encoder. The hierarchical model contains two kinds of latent variables at the local and global levels, respectively. At the local level, there are two latent variables, one for translation and the other for summarization. As for the global level, there is another latent variable for cross-lingual summarization conditioned on the two local-level variables. Experiments on two language directions (English⇔Chinese) verify the effectiveness and superiority of the proposed approach. In addition, we show that our model is able to generate better cross-lingual summaries than comparison models in the few-shot setting.

1 Introduction

The cross-lingual summarization (CLS) aims to summarize a document in source language (*e.g.*, English) into a different language (*e.g.*, Chinese), which can be seen as a combination of machine translation (MT) and monolingual summarization (MS) to some extent (Orăsan and Chiorean, 2008; Zhu et al., 2019). The CLS can help people effectively master the core points of an article in a foreign language. Under the background of globalization, it becomes more important and is now coming into widespread use in real life.

Many researches have been devoted to dealing with this task. To our knowledge, they mainly fall into two categories, *i.e.*, pipeline and end-to-end learning methods. (*i*) The first category is pipeline-based, adopting either translation-summarization (Leuski et al., 2003; Ouyang et al., 2019) or summarization-translation (Wan et al., 2010; Orăsan and Chiorean, 2008) paradigm. Although being intuitive and straightforward, they generally suffer from error propagation problem. (*ii*) The second category aims to train an end-to-end model for CLS (Zhu et al., 2019, 2020). For instance, Zhu et al. (2020) focus on using a pre-constructed probabilistic bilingual lexicon to improve the CLS model. Furthermore, some researches resort to multi-task learning (Takase and Okazaki, 2020; Bai et al., 2021a; Zhu et al., 2019; Cao et al., 2020a,b). Zhu et al. (2019) separately introduce MT and MS to improve CLS. Cao et al. (2020a,b) design several additional training objectives (*e.g.*, MS, back-translation, and reconstruction) to enhance the CLS model. And Xu et al. (2020) utilize a mixed-lingual pre-training method with several auxiliary tasks for CLS.

As pointed out by Cao et al. (2020a), it is challenging for the model to directly conduct CLS as it requires both the abilities to translate and summarize. Although some methods have used the related tasks (*e.g.*, MT and MS) to help the CLS, the hierarchical relationship between MT&MS and CLS are not well modeled, which can explicitly enhance the CLS task. Apparently, how to effectively model the hierarchical relationship to exploit MT and MS is one of the core issues, especially when the CLS data are limited.[1] In many other related NLP tasks (Park et al., 2018; Serban et al., 2017;

[*] Work was done when Liang and Zhou were interning at Pattern Recognition Center, WeChat AI, Tencent Inc, China.

[†] Jinan Xu is the corresponding author.

[1] Generally, it is difficult to acquire the CLS dataset (Zhu et al., 2020; Ayana et al., 2018; Duan et al., 2019).

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 2088 - 2099
May 22-27, 2022

Shen et al., 2019, 2021), the Conditional Variational Auto-Encoder (CVAE) (Sohn et al., 2015) has shown its superiority in learning hierarchical structure with hierarchical latent variables, which is often leveraged to capture the semantic connection between the utterance and the corresponding context of conversations. Inspired by these work, we attempt to adapt CVAE to model the hierarchical relationship between MT&MS and CLS.

Therefore, we propose a Variational Hierarchical Model to exploit translation and summarization simultaneously, named VHM, for CLS task in an end-to-end framework. VHM employs hierarchical latent variables based on CVAE to learn the hierarchical relationship between MT&MS and CLS. Specifically, the VHM contains two kinds of latent variables at the local and global levels, respectively. Firstly, we introduce two local variables for translation and summarization, respectively. The two local variables are constrained to reconstruct the translation and source-language summary. Then, we use the global variable to explicitly exploit the two local variables for better CLS, which is constrained to reconstruct the target-language summary. This makes sure the global variable captures its relationship with the two local variables without any loss, preventing error propagation. For inference, we use the local and global variables to assist the cross-lingual summarization process.

We validate our proposed training framework on the datasets of different language pairs (Zhu et al., 2019): Zh2EnSum (Chinese⇒English) and En2ZhSum (English⇒Chinese). Experiments show that our model achieves consistent improvements on two language directions in terms of both automatic metrics and human evaluation, demonstrating its effectiveness and generalizability. Few-shot evaluation further suggests that the local and global variables enable our model to generate a satisfactory cross-lingual summaries compared to existing related methods.

Our main contributions are as follows[2]:

- We are the first that builds a variational hierarchical model via conditional variational auto-encoders that introduce a global variable to combine the local ones for translation and summarization at the same time for CLS.
- Our model gains consistent and significant performance and remarkably outperforms the most previous state-of-the-art methods after using mBART (Liu et al., 2020).
- Under the few-shot setting, our model still achieves better performance than existing approaches. Particularly, the fewer the data are, the greater the improvement we gain.

2 Background

Machine Translation (MT). Given an input sequence in the source language $X_{mt}{=}\{x_i\}_{i=1}^{|X_{mt}|}$, the goal of the neural MT model is to produce its translation in the target language $Y_{mt}{=}\{y_i\}_{i=1}^{|Y_{mt}|}$. The conditional distribution of the model is:

$$p_\theta(Y_{mt}|X_{mt}) = \prod_{t=1}^{|Y_{mt}|} p_\theta(y_t|X_{mt}, y_{1:t-1}),$$

where θ are model parameters and $y_{1:t-1}$ is the partial translation.

Monolingual Summarization (MS). Given an input article in the source language $X_{ms}^{src}{=}\{x_i^{src}\}_{i=1}^{|X_{ms}^{src}|}$ and the corresponding summarization in the same language $X_{ms}^{tgt}{=}\{x_i^{tgt}\}_{i=1}^{|X_{ms}^{tgt}|}$, the monolingual summarization is formalized as:

$$p_\theta(X_{ms}^{tgt}|X_{ms}^{src}) = \prod_{t=1}^{|X_{ms}^{tgt}|} p_\theta(x_t^{tgt}|X_{ms}^{src}, x_{1:t-1}^{tgt}).$$

Cross-Lingual Summarization (CLS). In CLS, we aim to learn a model that can generate a summary in the target language $Y_{cls}{=}\{y_i\}_{i=1}^{|Y_{cls}|}$ for a given article in the source language $X_{cls}{=}\{x_i\}_{i=1}^{|X_{cls}|}$. Formally, it is as follows:

$$p_\theta(Y_{cls}|X_{cls}) = \prod_{t=1}^{|Y_{cls}|} p_\theta(y_t|X_{cls}, y_{1:t-1}).$$

Conditional Variational Auto-Encoder (CVAE). The CVAE (Sohn et al., 2015) consists of one prior network and one recognition (posterior) network, where the latter takes charge of guiding the learning of prior network via Kullback–Leibler (KL) divergence (Kingma and Welling, 2013). For example, the variational neural MT model (Zhang et al., 2016a; Su et al., 2018a; McCarthy et al., 2020; Su et al., 2018c), which introduces a random latent variable $\mathbf{z}$ into the neural MT conditional distribution:

$$p_\theta(Y_{mt}|X_{mt}) = \int_{\mathbf{z}} p_\theta(Y_{mt}|X_{mt}, \mathbf{z}) \cdot p_\theta(\mathbf{z}|X_{mt}) d\mathbf{z}. \tag{1}$$

Given a source sentence X, a latent variable $\mathbf{z}$ is

[2] The code is publicly available at: `https://github.com/XL2248/VHM`

firstly sampled by the prior network from the encoder, and then the target sentence is generated by the decoder: $Y_{mt} \sim p_\theta(Y_{mt}|X_{mt}, \mathbf{z})$, where $\mathbf{z} \sim p_\theta(\mathbf{z}|X_{mt})$.

As it is hard to marginalize Eq. 1, the CVAE training objective is a variational lower bound of the conditional log-likelihood:

$$\begin{aligned}\mathcal{L}(\theta, \phi; X_{mt}, Y_{mt}) &= -\mathrm{KL}(q_\phi(\mathbf{z}'|X_{mt}, Y_{mt}) \| p_\theta(\mathbf{z}|X_{mt})) \\ &+ \mathbb{E}_{q_\phi(\mathbf{z}'|X_{mt}, Y_{mt})}[\log p_\theta(Y_{mt}|\mathbf{z}, X_{mt})] \\ &\leq \log p(Y_{mt}|X_{mt}),\end{aligned}$$

where ϕ are parameters of the CVAE.

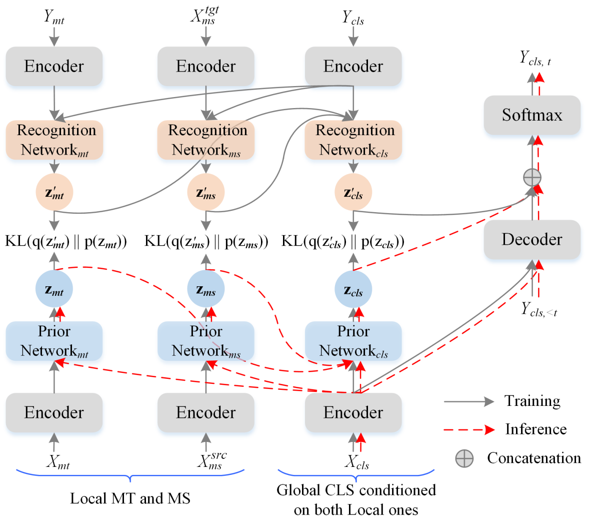

Figure 1: Overview of the proposed VHM framework. The local variables $\mathbf{z}_{mt}$, $\mathbf{z}_{ms}$ are tailored for translation and summarization, respectively. Then the global one $\mathbf{z}_{cls}$ is for cross-lingual summarization, where the $\mathbf{z}_{cls}$ not only conditions on the input but also $\mathbf{z}_{mt}$ and $\mathbf{z}_{ms}$. The solid grey lines indicate training process responsible for generating $\{\mathbf{z}'_{mt}, \mathbf{z}'_{ms}, \mathbf{z}'_{cls}\}$ from the corresponding posterior distribution predicted by recognition networks, which guide the learning of prior networks. The dashed red lines indicate inference process for generating $\{\mathbf{z}_{mt}, \mathbf{z}_{ms}, \mathbf{z}_{cls}\}$ from the corresponding prior distributions predicted by prior networks. The encoder is shared by all tasks with a bilingual vocabulary.

3 Methodology

Fig. 1 demonstrates an overview of our model, consisting of four components: *encoder*, *variational hierarchical modules*, *decoder*, *training and inference*. Specifically, we aim to explicitly exploit the MT and MS for CLS simultaneously. Therefore, we firstly use the *encoder* (§ 3.1) to prepare the representation for the *variational hierarchical module* (§ 3.2), which aims to learn the two local variables for the global variable in CLS. Then, we introduce the global variable into the *decoder* (§ 3.3). Finally, we elaborate the process of our *training and inference* (§ 3.4).

3.1 Encoder

Our model is based on transformer (Vaswani et al., 2017) framework. As shown in Fig. 1, the encoder takes six types of inputs, $\{X_{mt}, X_{ms}^{src}, X_{cls}, Y_{mt}, X_{ms}^{tgt}, Y_{cls}\}$, among which Y_{mt}, X_{ms}^{tgt}, and Y_{cls} are only for training recognition networks. Taking X_{mt} for example, the encoder maps the input X_{mt} into a sequence of continuous representations whose size varies with respect to the source sequence length. Specifically, the encoder consists of N_e stacked layers and each layer includes two sub-layers:[3] a multi-head self-attention (SelfAtt) sub-layer and a position-wise feed-forward network (FFN) sub-layer:

$$\begin{aligned}\mathbf{s}_e^\ell &= \mathrm{SelfAtt}(\mathbf{h}_e^{\ell-1}) + \mathbf{h}_e^{\ell-1}, \\ \mathbf{h}_e^\ell &= \mathrm{FFN}(\mathbf{s}_e^\ell) + \mathbf{s}_e^\ell,\end{aligned}$$

where $\mathbf{h}_e^\ell$ denotes the state of the ℓ-th encoder layer and $\mathbf{h}_e^0$ denotes the initialized embedding.

Through the encoder, we prepare the representations of $\{X_{mt}, X_{ms}^{src}, X_{cls}\}$ for training prior networks, encoder and decoder. Taking X_{mt} for example, we follow Zhang et al. (2016a) and apply *mean-pooling* over the output $\mathbf{h}_e^{N_e, X_{mt}}$ of the N_e-th encoder layer:

$$\mathbf{h}_{X_{mt}} = \frac{1}{|X_{mt}|}\sum_{i=1}^{|X_{mt}|}(\mathbf{h}_{e,i}^{N_e, X_{mt}}).$$

Similarly, we obtain $\mathbf{h}_{X_{ms}^{src}}$ and $\mathbf{h}_{X_{cls}}$.

For training recognition networks, we obtain the representations of $\{Y_{mt}, X_{ms}^{tgt}, Y_{cls}\}$, taking Y_{mt} for example, and calculate it as follows:

$$\mathbf{h}_{Y_{mt}} = \frac{1}{|Y_{mt}|}\sum_{i=1}^{|Y_{mt}|}(\mathbf{h}_{e,i}^{N_e, Y_{mt}}).$$

Similarly, we obtain $\mathbf{h}_{X_{ms}^{tgt}}$ and $\mathbf{h}_{Y_{cls}}$.

3.2 Variational Hierarchical Modules

Firstly, we design two local latent variational modules to learn the translation distribution in MT pairs and summarization distribution in MS pairs, respectively. Then, conditioned on them, we introduce a global latent variational module to explicitly exploit them.

[3] The layer normalization is omitted for simplicity and you may refer to (Vaswani et al., 2017) for more details.

3.2.1 Local: Translation and Summarization

Translation. To capture the translation of the paired sentences, we introduce a local variable $\mathbf{z}_{mt}$ that is responsible for generating the target information. Inspired by Wang and Wan (2019), we use isotropic Gaussian distribution as the prior distribution of $\mathbf{z}_{mt}$: $p_\theta(\mathbf{z}_{mt}|X_{mt}) \sim \mathcal{N}(\boldsymbol{\mu}_{mt}, \boldsymbol{\sigma}^2_{mt}\mathbf{I})$, where $\mathbf{I}$ denotes the identity matrix and we have

$$\begin{aligned} \boldsymbol{\mu}_{mt} &= \mathrm{MLP}^{mt}_\theta(\mathbf{h}_{X_{mt}}), \\ \boldsymbol{\sigma}_{mt} &= \mathrm{Softplus}(\mathrm{MLP}^{mt}_\theta(\mathbf{h}_{X_{mt}})), \end{aligned} \tag{2}$$

where $\mathrm{MLP}(\cdot)$ and $\mathrm{Softplus}(\cdot)$ are multi-layer perceptron and approximation of ReLU function, respectively.

At training, the posterior distribution conditions on both source input and the target reference, which provides translation information. Therefore, the prior network can learn a tailored translation distribution by approaching the recognition network via KL divergence (Kingma and Welling, 2013): $q_\phi(\mathbf{z}'_{mt}|X_{mt}, Y_{mt}) \sim \mathcal{N}(\boldsymbol{\mu}'_{mt}, \boldsymbol{\sigma}'^2_{mt}\mathbf{I})$, where $\boldsymbol{\mu}'_{mt}$ and $\boldsymbol{\sigma}'_{mt}$ are calculated as:

$$\begin{aligned} \boldsymbol{\mu}'_{mt} &= \mathrm{MLP}^{mt}_\phi(\mathbf{h}_{X_{mt}}; \mathbf{h}_{Y_{mt}}), \\ \boldsymbol{\sigma}'_{mt} &= \mathrm{Softplus}(\mathrm{MLP}^{mt}_\phi(\mathbf{h}_{X_{mt}}; \mathbf{h}_{Y_{mt}})), \end{aligned} \tag{3}$$

where $(\cdot;\cdot)$ indicates concatenation operation.

Summarization. To capture the summarization in MS pairs, we introduce another local variable $\mathbf{z}_{ms}$, which takes charge of generating the source-language summary. Similar to $\mathbf{z}_{mt}$, we define its prior distribution as: $p_\theta(\mathbf{z}_{ms}|X^{src}_{ms}) \sim \mathcal{N}(\boldsymbol{\mu}_{ms}, \boldsymbol{\sigma}^2_{ms}\mathbf{I})$, where $\boldsymbol{\mu}_{ms}$ and $\boldsymbol{\sigma}_{ms}$ are calculated as:

$$\begin{aligned} \boldsymbol{\mu}_{ms} &= \mathrm{MLP}^{ms}_\theta(\mathbf{h}_{X^{src}_{ms}}), \\ \boldsymbol{\sigma}_{ms} &= \mathrm{Softplus}(\mathrm{MLP}^{ms}_\theta(\mathbf{h}_{X^{src}_{ms}})). \end{aligned} \tag{4}$$

At training, the posterior distribution conditions on both the source input and the source-language summary that contains the summarization clue, and thus is responsible for guiding the learning of the prior distribution. Specifically, we define the posterior distribution as: $q_\phi(\mathbf{z}'_{ms}|X^{src}_{ms}, X^{tgt}_{ms}) \sim \mathcal{N}(\boldsymbol{\mu}'_{ms}, \boldsymbol{\sigma}'^2_{ms}\mathbf{I})$, where $\boldsymbol{\mu}'_{ms}$ and $\boldsymbol{\sigma}'_{ms}$ are calculated as:

$$\begin{aligned} \boldsymbol{\mu}'_{ms} &= \mathrm{MLP}^{ms}_\phi(\mathbf{h}_{X^{src}_{ms}}; \mathbf{h}_{X^{tgt}_{ms}}), \\ \boldsymbol{\sigma}'_{ms} &= \mathrm{Softplus}(\mathrm{MLP}^{ms}_\phi(\mathbf{h}_{X^{src}_{ms}}; \mathbf{h}_{X^{tgt}_{ms}})). \end{aligned} \tag{5}$$

3.2.2 Global: CLS

After obtaining $\mathbf{z}_{mt}$ and $\mathbf{z}_{ms}$, we introduce the global variable $\mathbf{z}_{cls}$ that aims to generate a target-language summary, where the $\mathbf{z}_{cls}$ can simultaneously exploit the local variables for CLS. Specifically, we firstly encode the source input X_{cls} and condition on both two local variables $\mathbf{z}_{mt}$ and $\mathbf{z}_{ms}$, and then sample $\mathbf{z}_{cls}$. We define its prior distribution as: $p_\theta(\mathbf{z}_{cls}|X_{cls}, \mathbf{z}_{mt}, \mathbf{z}_{ms}) \sim \mathcal{N}(\boldsymbol{\mu}_{cls}, \boldsymbol{\sigma}^2_{cls}\mathbf{I})$, where $\boldsymbol{\mu}_{cls}$ and $\boldsymbol{\sigma}_{cls}$ are calculated as:

$$\begin{aligned} \boldsymbol{\mu}_{cls} &= \mathrm{MLP}^{cls}_\theta(\mathbf{h}_{X_{cls}}; \mathbf{z}_{mt}; \mathbf{z}_{ms}), \\ \boldsymbol{\sigma}_{cls} &= \mathrm{Softplus}(\mathrm{MLP}^{cls}_\theta(\mathbf{h}_{X_{cls}}; \mathbf{z}_{mt}; \mathbf{z}_{ms})). \end{aligned} \tag{6}$$

At training, the posterior distribution conditions on the local variables, the CLS input, and the cross-lingual summary that contains combination information of translation and summarization. Therefore, the posterior distribution can teach the prior distribution. Specifically, we define the posterior distribution as: $q_\phi(\mathbf{z}'_{cls}|X_{cls}, \mathbf{z}_{mt}, \mathbf{z}_{ms}, Y_{cls}) \sim \mathcal{N}(\boldsymbol{\mu}'_{cls}, \boldsymbol{\sigma}'^2_{cls}\mathbf{I})$, where $\boldsymbol{\mu}'_{cls}$ and $\boldsymbol{\sigma}'_{cls}$ are calculated as:

$$\begin{aligned} \boldsymbol{\mu}'_{cls} &= \mathrm{MLP}^{cls}_\phi(\mathbf{h}_{X_{cls}}; \mathbf{z}_{mt}; \mathbf{z}_{ms}; \mathbf{h}_{Y_{cls}}), \\ \boldsymbol{\sigma}'_{cls} &= \mathrm{Softplus}(\mathrm{MLP}^{cls}_\phi(\mathbf{h}_{X_{cls}}; \mathbf{z}_{mt}; \mathbf{z}_{ms}; \mathbf{h}_{Y_{cls}})). \end{aligned} \tag{7}$$

3.3 Decoder

The decoder adopts a similar structure to the encoder, and each of N_d decoder layers includes an additional cross-attention sub-layer (CrossAtt):

$$\begin{aligned} \mathbf{s}^\ell_d &= \mathrm{SelfAtt}(\mathbf{h}^{\ell-1}_d) + \mathbf{h}^{\ell-1}_d, \\ \mathbf{c}^\ell_d &= \mathrm{CrossAtt}(\mathbf{s}^\ell_d, \mathbf{h}^{N_e}_e) + \mathbf{s}^\ell_d, \\ \mathbf{h}^\ell_d &= \mathrm{FFN}(\mathbf{c}^\ell_d) + \mathbf{c}^\ell_d, \end{aligned}$$

where $\mathbf{h}^\ell_d$ denotes the state of the ℓ-th decoder layer.

As shown in Fig. 1, we firstly obtain the local two variables either from the posterior distribution predicted by recognition networks (training process as the solid grey lines) or from prior distribution predicted by prior networks (inference process as the dashed red lines). Then, conditioned on the local two variables, we generate the global variable ($\mathbf{z}'_{cls}$/$\mathbf{z}_{cls}$) via posterior (training) or prior (inference) network. Finally, we incorporate $\mathbf{z}^{(\prime)}_{cls}$[4] into the state of the top layer of the decoder with a projection layer:

$$\mathbf{o}_t = \mathrm{Tanh}(\mathbf{W}_p[\mathbf{h}^{N_d}_{d,t}; \mathbf{z}^{(\prime)}_{cls}] + \mathbf{b}_p), \tag{8}$$

where $\mathbf{W}_p$ and $\mathbf{b}_p$ are training parameters, $\mathbf{h}^{N_d}_{d,t}$ is the hidden state at time-step t of the N_d-th decoder layer. Then, $\mathbf{o}_t$ is fed into a linear transformation and softmax layer to predict the probability distri-

[4] Here, we use $\mathbf{z}'_{cls}$ when training and $\mathbf{z}_{cls}$ during inference, as similar to Eq. 8.

bution of the next target token:

$$\mathbf{p}_t = \mathrm{Softmax}(\mathbf{W}_o \mathbf{o}_t + \mathbf{b}_o),$$

where $\mathbf{W}_o$ and $\mathbf{b}_o$ are training parameters.

3.4 Training and Inference

The model is trained to maximize the conditional log-likelihood, due to the intractable marginal likelihood, which is converted to the following variational lower bound that needs to be maximized in the training process:

$$\begin{aligned}
&\mathcal{J}(\theta, \phi; X_{cls}, X_{mt}, X_{ms}^{src}, Y_{cls}, Y_{mt}, X_{ms}^{tgt}) = \\
&- \mathrm{KL}(q_\phi(\mathbf{z}'_{mt}|X_{mt}, Y_{mt}) \| p_\theta(\mathbf{z}_{mt}|X_{mt})) \\
&- \mathrm{KL}(q_\phi(\mathbf{z}'_{ms}|X_{ms}^{src}, X_{ms}^{tgt}) \| p_\theta(\mathbf{z}_{ms}|X_{ms}^{src})) \\
&- \mathrm{KL}(q_\phi(\mathbf{z}'_{cls}|X_{cls}, \mathbf{z}_{mt}, \mathbf{z}_{ms}, Y_{cls}) \| p_\theta(\mathbf{z}_{cls}|X_{cls}, \mathbf{z}_{mt}, \mathbf{z}_{ms})) \\
&+ \mathbb{E}_{q_\phi}[\mathrm{log} p_\theta(Y_{mt}|X_{mt}, \mathbf{z}_{mt})] \\
&+ \mathbb{E}_{q_\phi}[\mathrm{log} p_\theta(X_{ms}^{tgt}|X_{ms}^{src}, \mathbf{z}_{ms})] \\
&+ \mathbb{E}_{q_\phi}[\mathrm{log} p_\theta(Y_{cls}|X_{cls}, \mathbf{z}_{cls}, \mathbf{z}_{mt}, \mathbf{z}_{ms})],
\end{aligned}$$

where the variational lower bound includes the reconstruction terms and KL divergence terms based on three hierarchical variables. We use the reparameterization trick (Kingma and Welling, 2013) to estimate the gradients of the prior and recognition networks (Zhao et al., 2017).

During inference, firstly, the prior networks of MT and MS generate the local variables. Then, conditioned on them, the global variable is produced by prior network of CLS. Finally, only the global variable is fed into the decoder, which corresponds to red dashed arrows in Fig. 1.

4 Experiments

4.1 Datasets and Metrics

Datasets. We evaluate the proposed approach on Zh2EnSum and En2ZhSum datasets released by (Zhu et al., 2019).[5] The Zh2EnSum and En2ZhSum are originally from (Hu et al., 2015) and (Hermann et al., 2015; Zhu et al., 2018), respectively. Both the Chinese-to-English and English-to-Chinese test sets are manually corrected. The involved training data in our experiments are listed in Tab. 1.

Zh2EnSum	D1	CLS	Zh2EnSum	1,693,713
	D2	MS	LCSTS	1,693,713
	D3	MT	LDC	2.08M
En2ZhSum	D4	CLS	En2ZhSum	364,687
	D5	MS	ENSUM	364,687
	D3	MT	LDC	2.08M

Table 1: Involved training data. LCSTS (Hu et al., 2015) is a Chinese summarization dataset. LDC corpora includes LDC2000T50, LDC2002L27, LDC2002T01, LDC2002E18, LDC2003E07, LDC2003E14, LDC2003T17, and LDC2004T07. ENSUM consists of CNN/Dailymail (Hermann et al., 2015) and MSMO (Zhu et al., 2018).

Models	Zh2EnSum		
	Size (M)	Train (S)	Data
ATS-A	137.60	30	D1&D3
MS-CLS	211.41	48	D1&D2
MT-CLS	208.84	63	D1&D3
MT-MS-CLS	114.90	24	D1&D2&D3
VHM	117.40	27	D1&D2&D3

Table 2: Model details. Size (M): number of trainable parameters; Train (S) denotes how many seconds required for each model to train the 100-batch cross-lingual summarization task of the same batch size (3072). Data: Training Data, as listed in Tab. 1.

Models	En2ZhSum		
	Size (M)	Train (S)	Data
ATS-A	115.05	25	D4&D3
MS-CLS	190.23	65	D4&D5
MT-CLS	148.16	72	D4&D3
MT-MS-CLS	155.50	32	D4&D5&D3
VHM	158.00	36	D4&D5&D3

Table 3: Model details. Size (M): number of trainable parameters; Train (S) denotes how many seconds required for each model to train the 100-batch cross-lingual summarization task of the same batch size (3072). Data: Training Data, as listed in Tab. 1.

Zh2EnSum. It is a Chinese-to-English summarization dataset, which has 1,699,713 Chinese short texts (104 Chinese characters on average) paired with Chinese (18 Chinese characters on average) and English short summaries (14 tokens on average). The dataset is split into 1,693,713 training pairs, 3,000 validation pairs, and 3,000 test pairs. The involved training data used in multi-task learning, model size, training time, are listed in Tab. 2.

En2ZhSum. It is an English-to-Chinese summarization dataset, which has 370,687 English documents (755 tokens on average) paired with multi-sentence English (55 tokens on average) and Chinese summaries (96 Chinese characters on average). The dataset is split into 364,687 training pairs, 3,000 validation pairs, and 3,000 test pairs. The involved training data used in multi-task learning, model size, training time, are listed in Tab. 3.

Metrics. Following Zhu et al. (2020), 1) we evaluate all models with the standard ROUGE metric (Lin, 2004), reporting the F1 scores for ROUGE-1, ROUGE-2, and ROUGE-L. All ROUGE scores

[5] https://github.com/ZNLP/NCLS-Corpora

M#	Models	Zh2EnSum				En2ZhSum		
		RG1	RG2	RGL	MVS	RG1	RG2	RGL
M1	GETran (Zhu et al., 2019)	24.34	9.14	20.13	0.64	28.19	11.40	25.77
M2	GLTran (Zhu et al., 2019)	35.45	16.86	31.28	16.90	32.17	13.85	29.43
M3	TNCLS (Zhu et al., 2019)	38.85	21.93	35.05	19.43	36.82	18.72	33.20
M4	ATS-A (Zhu et al., 2020)	40.68	24.12	36.97	22.15	40.47	22.21	36.89
M5	MS-CLS (Zhu et al., 2019)	40.34	22.65	36.39	21.09	38.25	20.20	34.76
M6	MT-CLS (Zhu et al., 2019)	40.25	22.58	36.21	21.06	40.23	22.32	36.59
M7	MS-CLS-Rec (Cao et al., 2020a)	40.97	23.20	36.96	NA	38.12	16.76	33.86
M8	MS-CLS*	40.44	22.19	36.32	21.01	38.26	20.07	34.49
M9	MT-CLS*	40.05	21.72	35.74	20.96	40.14	22.36	36.45
M10	MT-MS-CLS (Ours)	40.65	24.02	36.69	22.17	40.34	22.35	36.44
M11	VHM (Ours)	$41.36^{\dagger\dagger}$	$24.64^{\dagger}$	$37.15^{\dagger}$	$22.55^{\dagger}$	$40.98^{\dagger\dagger}$	$23.07^{\dagger\dagger}$	$37.12^{\dagger}$
M12	mBART (Liu et al., 2020)	43.61	25.14	38.79	23.47	41.55	23.27	37.22
M13	MLPT (Xu et al., 2020)	43.50	25.41	29.66	NA	41.62	23.35	37.26
M14	VHM + mBART (Ours)	**43.97**†	**25.61**†	**39.19**†	**23.88**	**41.95**†	**23.54**†	**37.67**†

Table 4: ROUGE F1 scores (%) and MoverScore scores (%) on Zh2EnSum test set, and ROUGE F1 scores (%) on En2ZhSum test set. RG and MVS refer to ROUGE and MoverScore, respectively. The "*" denotes results by running their released code. The "NA" indicates no such result in the original paper. "$\dagger$" and "$\dagger\dagger$" indicate that statistically significant better (M11 vs. M4 and M14 vs. M12) with t-test $p < 0.05$ and $p < 0.01$, respectively. "VHM + mBART" means that we use mBART weights as model initialization of our VHM.

are reported by the 95% confidence interval measured by the official script;[6] 2) we also evaluate the quality of English summaries in Zh2EnSum with MoverScore (Zhao et al., 2019).

4.2 Implementation Details

In this paper, we train all models using standard transformer (Vaswani et al., 2017) in *Base* setting. For other hyper-parameters, we mainly follow the setting described in Zhu et al. (2019, 2020) for fair comparison. For more details, please refer to Appendix A.

4.3 Comparison Models

Pipeline Models. TETran (Zhu et al., 2019). It first translates the original article into the target language by Google Translator[7] and then summarizes the translated text via LexRank (Erkan and Radev, 2004). TLTran (Zhu et al., 2019). It first summarizes the original article via a transformer-based monolingual summarization model and then translates the summary into the target language by Google Translator.

End-to-End Models. TNCLS (Zhu et al., 2019). It directly uses the de-facto transformer (Vaswani et al., 2017) to train an end-to-end CLS system. ATS-A (Zhu et al., 2020).[8] It is an efficient model to attend the pre-constructed probabilistic bilingual lexicon to enhance the CLS. MS-CLS (Zhu et al., 2019). It simultaneously performs summarization generation for both CLS and MS tasks and calculates the total losses. MT-CLS (Zhu et al., 2019).[9] It alternatively trains CLS and MT tasks. MS-CLS-Rec (Cao et al., 2020a). It jointly trains MS and CLS systems with a reconstruction loss to mutually map the source and target representations. mBART (Liu et al., 2020). We use mBART ($mbart.cc25$) as model initialization to fine-tune the CLS task. MLPT (Mixed-Lingual Pre-training) (Xu et al., 2020). It applies mixed-lingual pretraining that leverages six related tasks, covering both cross-lingual tasks such as translation and monolingual tasks like masked language models. MT-MS-CLS. It is our strong baseline, which is implemented by alternatively training CLS, MT, and MS. Here, we keep the dataset used for MT and MS consistent with Zhu et al. (2019) for fair comparison.

4.4 Main Results

Overall, we separate the models into three parts in Tab. 4: the pipeline, end-to-end, and multi-task

[6] The parameter for ROUGE script here is "-c 95 -r 1000 -n 2 -a"

[7] https://translate.google.com/

[8] https://github.com/ZNLP/ATSum

[9] https://github.com/ZNLP/NCLS-Corpora

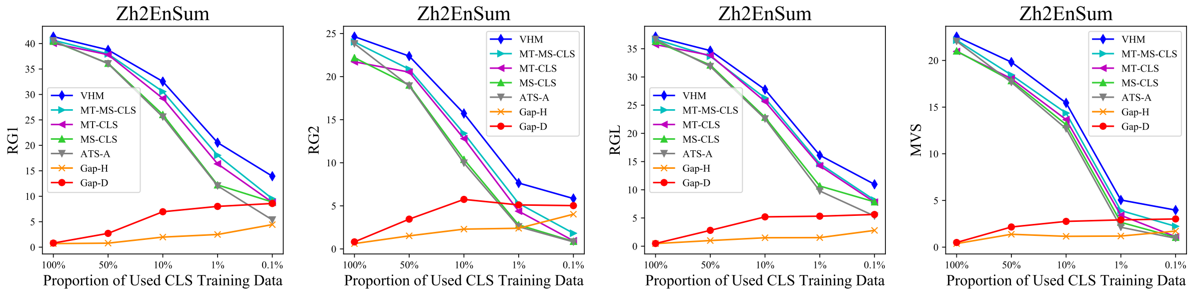

Figure 2: ROUGE F1 scores (%) and MoverScore scores (%) on Zh2EnSum test set in few-shot setting. **x%** means that the **x%** CLS training dataset is used, *e.g.*, **0.1%** represents that **0.1%** training dataset (about 1.7k instances) is used for training. The performance "Gap-H" (orange line) between "VHM" and "MT-MS-CLS" grows steadily with the decreasing of used CLS training data, which is similar to the performance "Gap-D" (red line) between "VHM" and "ATS-A".

settings. In each part, we show the results of existing studies and our re-implemented baselines and our approach, *i.e.*, the VHM, on Zh2EnSum and En2ZhSum test sets.

Results on Zh2EnSum. Compared against the pipeline and end-to-end methods, VHM substantially outperforms all of them (*e.g.*, the previous best model "ATS-A") by a large margin with 0.68/0.52/0.18/0.4↑ scores on RG1/RG2/RGL/MVS, respectively. Under the multi-task setting, compared to the existing best model "MS-CLS-Rec", our VHM also consistently boosts the performance in three metrics (*i.e.*, 0.39↑, 1.44↑, and 0.19↑ ROUGE scores on RG1/RG2/RGL, respectively), showing its effectiveness. Our VHM also significantly surpasses our strong baseline "MT-MS-CLS" by 0.71/0.62/0.46/0.38↑ scores on RG1/RG2/RGL/MVS, respectively, demonstrating the superiority of our model again.

After using mBART as model initialization, our VHM achieves the state-of-the-art results on all metrics.

Results on En2ZhSum. Compared against the pipeline, end-to-end and multi-task methods, our VHM presents remarkable ROUGE improvements over the existing best model "ATS-A" by a large margin, about 0.51/0.86/0.23↑ ROUGE gains on RG1/RG2/RGL, respectively. These results suggest that VHM consistently performs well in different language directions.

Our approach still notably surpasses our strong baseline "MT-MS-CLS" in terms of all metrics, which shows the generalizability and superiority of our model again.

4.5 Few-Shot Results

Due to the difficulty of acquiring the cross-lingual summarization dataset (Zhu et al., 2019), we conduct such experiments to investigate the model performance when the CLS training dataset is limited, *i.e.*, few-shot experiments. Specifically, we randomly choose 0.1%, 1%, 10%, and 50% CLS training datasets to conduct experiments. The results are shown in Fig. 2 and Fig. 3.

Results on Zh2EnSum. Fig. 2 shows that VHM significantly surpasses all comparison models under each setting. Particularly, under the 0.1% setting, our model still achieves best performances than all baselines, suggesting that our variational hierarchical model works well in the few-shot setting as well. Besides, we find that the performance gap between comparison models and VHM is growing when the used CLS training data become fewer. It is because relatively larger proportion of translation and summarization data are used, the influence from MT and MS becomes greater, effectively strengthening the CLS model. Particularly, the performance "Gap-H" between MT-MS-CLS and VHM is also growing, where both models utilize the same data. This shows that the hierarchical relationship between MT&MS and CLS makes substantial contributions to the VHM model in terms of four metrics. Consequently, our VHM achieves a comparably stable performance.

Results on En2ZhSum. From Fig. 3, we observe the similar findings on Zh2EnSum. This shows that VHM significantly outperforms all comparison models under each setting, showing the generalizability and superiority of our model again in the few-shot setting.

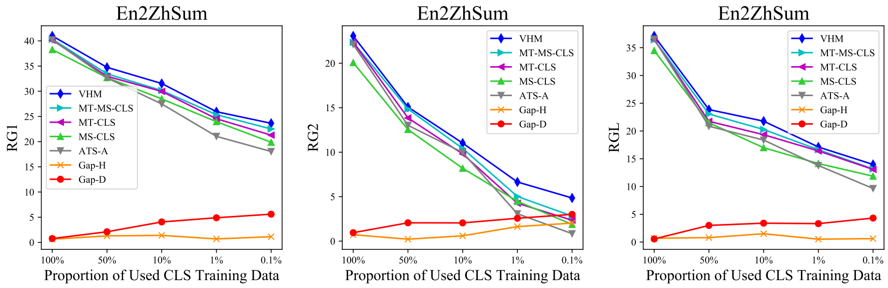

Figure 3: Rouge F1 scores (%) on the test set when using different CLS training data. The performance "Gap-H" (orange line) between "VHM" and "MT-MS-CLS" grows steadily with the decreasing of used CLS training data on ROUGE-2, which is similar to the performance "Gap-D" (red line) between "VHM" and "ATS-A".

#	Models	Zh2EnSum	En2ZhSum
		RG1/RG2/RGL/MVS	RG1/RG2/RGL
0	VHM	41.36/24.64/37.15/22.55	40.98/23.07/37.12
1	– $\mathbf{z}_{mt}$	40.75/23.47/36.48/22.18	40.35/22.48/36.55
2	– $\mathbf{z}_{ms}$	40.69/23.34/36.35/22.12	40.57/22.79/36.71
3	– $\mathbf{z}_{mt}$&$\mathbf{z}_{ms}$	40.45/22.97/36.03/22.36	39.98/21.91/36.33
4	– $\mathbf{z}_{cls}$	39.77/22.41/34.87/21.62	39.76/21.69/35.99
5	– hierarchy	40.47/22.64/34.96/21.78	39.67/21.79/35.87

Table 5: Ablation results (in the full setting). Row 1 denotes that we remove the local variable $\mathbf{z}_{mt}$, and sample $\mathbf{z}_{cls}$ from the source input and another local variable $\mathbf{z}_{ms}$, similarly for row 2. Row 3 denotes that we remove both local variables $\mathbf{z}_{mt}$ and $\mathbf{z}_{ms}$ and sample $\mathbf{z}_{cls}$ only from the source input. Row 4 means that we remove the global variable $\mathbf{z}_{cls}$ and directly attend the local variables $\mathbf{z}_{mt}$ and $\mathbf{z}_{ms}$ in Eq. 8. Row 5 represents that we keep three latent variables but remove the hierarchical relation between $\mathbf{z}_{cls}$ and $\mathbf{z}_{mt}$&$\mathbf{z}_{ms}$.

5 Analysis

5.1 Ablation Study

We conduct ablation studies to investigate how well the local and global variables of our VHM works. When removing variables listed in Tab. 5, we have the following findings.

(1) Rows 1∼3 vs. row 0 shows that the model performs worse, especially when removing the two local ones (row 3), due to missing the explicit translation or summarization or both information provided by the local variables, which is important to CLS. Besides, row 3 indicates that directly attending to $\mathbf{z}_{cls}$ leads to poor performances, showing the necessity of the hierarchical structure, *i.e.*, using the global variable to exploit the local ones.

(2) Rows 4∼5 vs. row 0 shows that directly attending the local translation and summarization cannot achieve good results due to lacking of the global combination of them, showing that it is very necessary for designing the variational hierarchical model, *i.e.*, using a global variable to well exploit and combine the local ones.

5.2 Human Evaluation

Following Zhu et al. (2019, 2020), we conduct human evaluation on 25 random samples from each of the Zh2EnSum and En2ZhSum test set. We compare the summaries generated by our methods (MT-MS-CLS and VHM) with the summaries generated by ATS-A, MS-CLS, and MT-CLS in the full setting and few-shot setting (0.1%), respectively. We invite three graduate students to compare the generated summaries with human-corrected references, and assess each summary from three independent perspectives:

1. How **informative** (*i.e.*, IF) the summary is?
2. How **concise** (*i.e.*, CC) the summary is?
3. How **fluent**, grammatical (*i.e.*, FL) the summary is?

Each property is assessed with a score from 1 (worst) to 5 (best). The average results are presented in Tab. 6 and Tab. 7.

Tab. 6 shows the results in the full setting. We find that our VHM outperforms all comparison models from three aspects in both language directions, which further demonstrates the effectiveness and superiority of our model.

Tab. 7 shows the results in the few-shot setting, where only 0.1% CLS training data are used in all models. We find that our VHM still performs best than all other models from three perspectives in both datasets, suggesting its generalizability and effectiveness again under different settings.

6 Related Work

Cross-Lingual Summarization. Conventional cross-lingual summarization methods mainly focus on incorporating bilingual information into

Models	Zh2EnSum			En2ZhSum		
	IF	CC	FL	IF	CC	FL
ATS-A	3.44	4.16	3.98	3.12	3.31	3.28
MS-CLS	3.12	4.08	3.76	3.04	3.22	3.12
MT-CLS	3.36	4.24	4.14	3.18	3.46	3.36
MT-MS-CLS	3.42	4.46	4.22	3.24	3.48	3.42
VHM	**3.56**	**4.54**	**4.38**	**3.36**	**3.54**	**3.48**

Table 6: Human evaluation results in the full setting. IF, CC and FL denote **informative**, **concise**, and **fluent** respectively.

Models	Zh2EnSum			En2ZhSum		
	IF	CC	FL	IF	CC	FL
ATS-A	2.26	2.96	2.82	2.04	2.58	2.68
MS-CLS	2.24	2.84	2.78	2.02	2.52	2.64
MT-CLS	2.38	3.02	2.88	2.18	2.74	2.76
MT-MS-CLS	2.54	3.08	2.92	2.24	2.88	2.82
VHM	**2.68**	**3.16**	**3.08**	**2.56**	**3.06**	**2.88**

Table 7: Human evaluation results in the few-shot setting (0.1%).

the pipeline methods (Leuski et al., 2003; Ouyang et al., 2019; Orăsan and Chiorean, 2008; Wan et al., 2010; Wan, 2011; Yao et al., 2015; Zhang et al., 2016b), *i.e.*, translation and then summarization or summarization and then translation. Due to the difficulty of acquiring cross-lingual summarization dataset, some previous researches focus on constructing datasets (Ladhak et al., 2020; Scialom et al., 2020; Yela-Bello et al., 2021; Zhu et al., 2019; Hasan et al., 2021; Perez-Beltrachini and Lapata, 2021; Varab and Schluter, 2021), mixed-lingual pre-training (Xu et al., 2020), knowledge distillation (Nguyen and Tuan, 2021), contrastive learning (Wang et al., 2021) or zero-shot approaches (Ayana et al., 2018; Duan et al., 2019; Dou et al., 2020), *i.e.*, using machine translation (MT) or monolingual summarization (MS) or both to train the CLS system. Among them, Zhu et al. (2019) propose to use roundtrip translation strategy to obtain large-scale CLS datasets and then present two multi-task learning methods for CLS. Based on this dataset, Zhu et al. (2020) leverage an end-to-end model to attend the pre-constructed probabilistic bilingual lexicon to improve CLS. To further enhance CLS, some studies resort to shared decoder (Bai et al., 2021a), more pseudo training data (Takase and Okazaki, 2020), or more related task training (Cao et al., 2020b,a; Bai et al., 2021b). Wang et al. (2022) concentrate on building a benchmark dataset for CLS on dialogue field. Different from them, we propose a variational hierarchical model that introduces a global variable to simultaneously exploit and combine the local translation variable in MT pairs and local summarization variable in MS pais for CLS, achieving better results.

Conditional Variational Auto-Encoder. CVAE has verified its superiority in many fields (Sohn et al., 2015; Liang et al., 2021a; Zhang et al., 2016a; Su et al., 2018b). For instance, in dialogue, Shen et al. (2019), Park et al. (2018) and Serban et al. (2017) extend CVAE to capture the semantic connection between the utterance and the corresponding context with hierarchical latent variables. Although the CVAE has been widely used in NLP tasks, its adaption and utilization to cross-lingual summarization for modeling hierarchical relationship are non-trivial, and to the best of our knowledge, has never been investigated before in CLS.

Multi-Task Learning. Conventional multi-task learning (MTL) (Caruana, 1997), which trains the model on multiple related tasks to promote the representation learning and generalization performance, has been successfully used in NLP fields (Collobert and Weston, 2008; Deng et al., 2013; Liang et al., 2021d,c,b). In the CLS, conventional MTL has been explored to incorporate additional training data (MS, MT) into models (Zhu et al., 2019; Takase and Okazaki, 2020; Cao et al., 2020a). In this work, we instead focus on how to connect the relation between the auxiliary tasks at training to make the most of them for better CLS.

7 Conclusion

In this paper, we propose to enhance the CLS model by simultaneously exploiting MT and MS. Given the hierarchical relationship between MT&MS and CLS, we propose a variational hierarchical model to explicitly exploit and combine them in CLS process. Experiments on Zh2EnSum and En2ZhSum show that our model significantly improves the quality of cross-lingual summaries in terms of automatic metrics and human evaluations. Particularly, our model in the few-shot setting still works better, suggesting its superiority and generalizability.

Acknowledgements

The research work descried in this paper has been supported by the National Key R&D Program of China (2020AAA0108001) and the National Nature Science Foundation of China (No. 61976015, 61976016, 61876198 and 61370130). Liang is supported by 2021 Tencent Rhino-Bird Research Elite Training Program. The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve this paper.

References

Ayana, shi-qi Shen, Yun Chen, Cheng Yang, Zhi-yuan Liu, and Maosong Sun. 2018. Zero-shot cross-lingual neural headline generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP)*, 26(12):2319–2327.

Yu Bai, Yang Gao, and Heyan Huang. 2021a. Cross-lingual abstractive summarization with limited parallel resources. In *Proceedings of ACL-IJCNLP*, pages 6910–6924, Online. Association for Computational Linguistics.

Yu Bai, Heyan Huang, Kai Fan, Yang Gao, Zewen Chi, and Boxing Chen. 2021b. Bridging the gap: Cross-lingual summarization with compression rate. *CoRR*, abs/2110.07936.

Yue Cao, Hui Liu, and Xiaojun Wan. 2020a. Jointly learning to align and summarize for neural cross-lingual summarization. In *Proceedings of ACL*, pages 6220–6231, Online. Association for Computational Linguistics.

Yue Cao, Xiaojun Wan, Jinge Yao, and Dian Yu. 2020b. Multisumm: Towards a unified model for multi-lingual abstractive summarization. In *Proceedings of AAAI*, volume 34, pages 11–18.

Rich Caruana. 1997. Multitask learning. In *Machine Learning*, pages 41–75.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of ICML*, page 160–167.

Li Deng, Geoffrey E. Hinton, and Brian Kingsbury. 2013. New types of deep neural network learning for speech recognition and related applications: an overview. *2013 IEEE ICASSP*, pages 8599–8603.

Zi-Yi Dou, Sachin Kumar, and Yulia Tsvetkov. 2020. A deep reinforced model for zero-shot cross-lingual summarization with bilingual semantic similarity rewards. In *Proceedings of the Fourth Workshop on Neural Generation and Translation*, pages 60–68, Online. Association for Computational Linguistics.

Xiangyu Duan, Mingming Yin, Min Zhang, Boxing Chen, and Weihua Luo. 2019. Zero-shot cross-lingual abstractive sentence summarization through teaching generation and attention. In *Proceedings of ACL*, pages 3162–3172, Florence, Italy. Association for Computational Linguistics.

Günes Erkan and Dragomir R Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research (JAIR)*, 22:457–479.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of AISTATS*.

Tahmid Hasan, Abhik Bhattacharjee, Wasi Uddin Ahmad, Yuan-Fang Li, Yong-Bin Kang, and Rifat Shahriyar. 2021. Crosssum: Beyond english-centric cross-lingual abstractive text summarization for 1500+ language pairs. *CoRR*, abs/2112.08804.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Proceedings of NIPS*, pages 1693–1701.

Baotian Hu, Qingcai Chen, and Fangze Zhu. 2015. LC-STS: A large scale Chinese short text summarization dataset. In *Proceedings of EMNLP*, pages 1967–1972, Lisbon, Portugal. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of ICLR*.

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Faisal Ladhak, Esin Durmus, Claire Cardie, and Kathleen McKeown. 2020. WikiLingua: A new benchmark dataset for cross-lingual abstractive summarization. In *Findings of EMNLP*, pages 4034–4048, Online. Association for Computational Linguistics.

Anton Leuski, Chin-Yew Lin, Liang Zhou, Ulrich Germann, Franz Josef Och, and Eduard Hovy. 2003. Cross-lingual c*st*rd: English access to hindi information. *ACM Transactions on Asian Language Information Processing*, 2(3):245–269.

Yunlong Liang, Fandong Meng, Yufeng Chen, Jinan Xu, and Jie Zhou. 2021a. Modeling bilingual conversational characteristics for neural chat translation. In *Proceedings of ACL*, pages 5711–5724.

Yunlong Liang, Fandong Meng, Jinchao Zhang, Yufeng Chen, Jinan Xu, and Jie Zhou. 2021b. A dependency syntactic knowledge augmented interactive architecture for end-to-end aspect-based sentiment analysis. *Neurocomputing*.

Yunlong Liang, Fandong Meng, Jinchao Zhang, Yufeng Chen, Jinan Xu, and Jie Zhou. 2021c. An iterative multi-knowledge transfer network for aspect-based sentiment analysis. In *Findings of EMNLP*, pages 1768–1780.

Yunlong Liang, Chulun Zhou, Fandong Meng, Jinan Xu, Yufeng Chen, Jinsong Su, and Jie Zhou. 2021d. Towards making the most of dialogue characteristics for neural chat translation. In *Proceedings of EMNLP*, pages 67–79.

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

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Arya D. McCarthy, Xian Li, Jiatao Gu, and Ning Dong. 2020. Addressing posterior collapse with mutual information for improved variational neural machine translation. In *Proceedings of ACL*, pages 8512–8525.

Thong Nguyen and Luu Anh Tuan. 2021. Improving neural cross-lingual summarization via employing optimal transport distance for knowledge distillation. *CoRR*, abs/2112.03473.

Constantin Orăsan and Oana Andreea Chiorean. 2008. Evaluation of a cross-lingual Romanian-English multi-document summariser. In *Proceedings of LREC*, Marrakech, Morocco. European Language Resources Association (ELRA).

Jessica Ouyang, Boya Song, and Kathy McKeown. 2019. A robust abstractive system for cross-lingual summarization. In *Proceedings of NAACL*, pages 2025–2031, Minneapolis, Minnesota. Association for Computational Linguistics.

Yookoon Park, Jaemin Cho, and Gunhee Kim. 2018. A hierarchical latent structure for variational conversation modeling. In *Proceedings of NAACL*, pages 1792–1801, New Orleans, Louisiana. Association for Computational Linguistics.

Laura Perez-Beltrachini and Mirella Lapata. 2021. Models and datasets for cross-lingual summarisation. In *Proceedings of EMNLP*, pages 9408–9423, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. 2020. MLSUM: The multilingual summarization corpus. In *Proceedings of EMNLP*, pages 8051–8067, Online. Association for Computational Linguistics.

Iulian Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *Proceedings of AAAI*.

Lei Shen, Yang Feng, and Haolan Zhan. 2019. Modeling semantic relationship in multi-turn conversations with hierarchical latent variables. In *Proceedings of ACL*, pages 5497–5502, Florence, Italy. Association for Computational Linguistics.

Lei Shen, Fandong Meng, Jinchao Zhang, Yang Feng, and Jie Zhou. 2021. GTM: A generative triple-wise model for conversational question generation. In *Proceedings of ACL*, pages 3495–3506, Online. Association for Computational Linguistics.

Kihyuk Sohn, Honglak Lee, and Xinchen Yan. 2015. Learning structured output representation using deep conditional generative models. In *Proceedings of NIPS*, pages 3483–3491.

Jinsong Su, Shan Wu, Deyi Xiong, Yaojie Lu, Xianpei Han, and Biao Zhang. 2018a. Variational recurrent neural machine translation. In *Proceedings of AAAI*.

Jinsong Su, Shan Wu, Biao Zhang, Changxing Wu, Yue Qin, and Deyi Xiong. 2018b. A neural generative autoencoder for bilingual word embeddings. *Information Sciences*, 424:287–300.

Jinsong Su, Jiali Zeng, Deyi Xiong, Yang Liu, Mingxuan Wang, and Jun Xie. 2018c. A hierarchy-to-sequence attentional neural machine translation model. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(3):623–632.

Sho Takase and Naoaki Okazaki. 2020. Multi-task learning for cross-lingual abstractive summarization.

Daniel Varab and Natalie Schluter. 2021. MassiveSumm: a very large-scale, very multilingual, news summarisation dataset. In *Proceedings of EMNLP*, pages 10150–10161, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of NIPS*, pages 5998–6008.

Xiaojun Wan. 2011. Using bilingual information for cross-language document summarization. In *Proceedings of ACL*, pages 1546–1555, Portland, Oregon, USA. Association for Computational Linguistics.

Xiaojun Wan, Huiying Li, and Jianguo Xiao. 2010. Cross-language document summarization based on machine translation quality prediction. In *Proceedings of ACL*, pages 917–926, Uppsala, Sweden. Association for Computational Linguistics.

Danqing Wang, Jiaze Chen, Hao Zhou, Xipeng Qiu, and Lei Li. 2021. Contrastive aligned joint learning for multilingual summarization. In *Findings of ACL-IJCNLP*, pages 2739–2750, Online. Association for Computational Linguistics.

Jiaan Wang, Fandong Meng, Ziyao Lu, Duo Zheng, Zhixu Li, Jianfeng Qu, and Jie Zhou. 2022. Clidsum: A benchmark dataset for cross-lingual dialogue summarization. *arXiv preprint arXiv:2202.05599*.

Tianming Wang and Xiaojun Wan. 2019. T-cvae: Transformer-based conditioned variational autoencoder for story completion. In *Proceedings of IJCAI*, pages 5233–5239.

Ruochen Xu, Chenguang Zhu, Yu Shi, Michael Zeng, and Xuedong Huang. 2020. Mixed-lingual pre-training for cross-lingual summarization. In *Proceedings of AACL*, pages 536–541, Suzhou, China. Association for Computational Linguistics.

Jin-ge Yao, Xiaojun Wan, and Jianguo Xiao. 2015. Phrase-based compressive cross-language summarization. In *Proceedings of EMNLP*, pages 118–127, Lisbon, Portugal. Association for Computational Linguistics.

Jenny Paola Yela-Bello, Ewan Oglethorpe, and Navid Rekabsaz. 2021. MultiHumES: Multilingual humanitarian dataset for extractive summarization. In *Proceedings of EACL*, pages 1713–1717, Online. Association for Computational Linguistics.

Biao Zhang, Deyi Xiong, Jinsong Su, Hong Duan, and Min Zhang. 2016a. Variational neural machine translation. In *Proceedings of EMNLP*, pages 521–530.

Jiajun Zhang, Yu Zhou, and Chengqing Zong. 2016b. Abstractive cross-language summarization via translation model enhanced predicate argument structure fusing. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(10):1842–1853.

Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *Proceedings of ACL*, pages 654–664.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. 2019. MoverScore: Text generation evaluating with contextualized embeddings and earth mover distance. In *Proceedings of EMNLP-IJCNLP*, pages 563–578, Hong Kong, China. Association for Computational Linguistics.

Junnan Zhu, Haoran Li, Tianshang Liu, Yu Zhou, Jiajun Zhang, and Chengqing Zong. 2018. MSMO: Multimodal summarization with multimodal output. In *Proceedings of EMNLP*, pages 4154–4164, Brussels, Belgium. Association for Computational Linguistics.

Junnan Zhu, Qian Wang, Yining Wang, Yu Zhou, Jiajun Zhang, Shaonan Wang, and Chengqing Zong. 2019. NCLS: Neural cross-lingual summarization. In *Proceedings of EMNLP-IJCNLP*, pages 3054–3064, Hong Kong, China. Association for Computational Linguistics.

Junnan Zhu, Yu Zhou, Jiajun Zhang, and Chengqing Zong. 2020. Attend, translate and summarize: An efficient method for neural cross-lingual summarization. In *Proceedings of ACL*, pages 1309–1321, Online. Association for Computational Linguistics.

Appendix

A Implementation Details

In this paper, we train all models using standard transformer (Vaswani et al., 2017) in *Base* setting. For other hyper-parameters, we mainly follow the setting described in (Zhu et al., 2019, 2020) for fair comparison. Specifically, the segmentation granularity is "subword to subword" for Zh2EnSum, and "word to word" for En2ZhSum. All the parameters are initialized via Xavier initialization method (Glorot and Bengio, 2010). We train our models using standard transformer (Vaswani et al., 2017) in *Base* setting, which contains a 6-layer encoder (*i.e.*, N_e) and a 6-layer decoder (*i.e.*, N_d) with 512-dimensional hidden representations. And all latent variables have a dimension of 128. Each mini-batch contains a set of document-summary pairs with roughly 4,096 source and 4,096 target tokens. We apply Adam optimizer (Kingma and Ba, 2015) with β_1 = 0.9, β_2 = 0.998. Following Zhu et al. (2019), we train each task for about 800,000 iterations in all multi-task models (reaching convergence). To alleviate the degeneration problem of the variational framework, we apply KL annealing. The KL multiplier λ gradually increases from 0 to 1 over 400, 000 steps. All our methods without mBART as model initialization are trained and tested on a single NVIDIA Tesla V100 GPU. We use 8 NVIDIA Tesla V100 GPU to train our models when using mBART as model initialization, where the number of token on each GPU is set to 2,048 and the training step is set to 400, 000.

During inference, we use beam search with a beam size 4 and length penalty 0.6.