NCLS: Neural Cross-Lingual Summarization

Junnan Zhu[1,2], **Qian Wang**[1,2], **Yining Wang**[1,2],
Yu Zhou[1,2]*, **Jiajun Zhang**[1,2], **Shaonan Wang**[1,2], and **Chengqing Zong**[1,2,3]
[1] National Laboratory of Pattern Recognition, Institute of Automation, CAS, Beijing, China
[2] University of Chinese Academy of Sciences, Beijing, China
[3] CAS Center for Excellence in Brain Science and Intelligence Technology, Beijing, China
{junnan.zhu, yzhou, jjzhang, cqzong}@nlpr.ia.ac.cn

Abstract

Cross-lingual summarization (CLS) is the task to produce a summary in one particular language for a source document in a different language. Existing methods simply divide this task into two steps: summarization and translation, leading to the problem of error propagation. To handle that, we present an end-to-end CLS framework, which we refer to as Neural Cross-Lingual Summarization (NCLS), for the first time. Moreover, we propose to further improve NCLS by incorporating two related tasks, monolingual summarization and machine translation, into the training process of CLS under multi-task learning. Due to the lack of supervised CLS data, we propose a round-trip translation strategy to acquire two high-quality large-scale CLS datasets based on existing monolingual summarization datasets. Experimental results have shown that our NCLS achieves remarkable improvement over traditional pipeline methods on both English-to-Chinese and Chinese-to-English CLS human-corrected test sets. In addition, NCLS with multi-task learning can further significantly improve the quality of generated summaries. We make our dataset and code publicly available here: http://www.nlpr.ia.ac.cn/cip/dataset.htm.

1 Introduction

Given a document in one source language, cross-lingual summarization aims to produce a summary in a different target language, which can help people efficiently acquire the gist of an article in a foreign language. Traditional approaches to CLS are based on the pipeline paradigm, which either first translates the original document into target language and then summarizes the translated document (Leuski et al., 2003) or first summarizes the original document and then translates the summary into target language (Lim et al., 2004; Orasan and Chiorean, 2008; Wan et al., 2010). However, the current machine translation (MT) is not perfect, which results in the error propagation problem. Although end-to-end deep learning has made great progress in natural language processing, no one has yet applied it to CLS due to the lack of large-scale supervised dataset.

The input and output of CLS are in two different languages, which makes the data acquisition much more difficult than monolingual summarization (MS). To the best of our knowledge, no one has studied how to automatically build a high-quality large-scale CLS dataset. Therefore, in this work, we introduce a novel approach to directly address the lack of data. Specifically, we propose a simple yet effective round-trip translation strategy to obtain cross-lingual document-summary pairs from existing monolingual summarization datasets (Hermann et al., 2015; Zhu et al., 2018; Hu et al., 2015). More details can be found in Section 2 below.

Based on the dataset that we have constructed, we propose end-to-end models on cross-lingual summarization, which we refer to as Neural Cross-Lingual Summarization (NCLS). Furthermore, we consider improving CLS with two related tasks: MS and MT. We incorporate the training process of MS and MT into that of CLS under the multi-task learning framework (Caruana, 1997). Experimental results demonstrate that NCLS achieves remarkable improvement over traditional pipeline paradigm. In addition, both MS and MT can significantly help to produce better summaries.

Our main contributions are as follows:

- We propose a novel round-trip translation strategy to acquire large-scale CLS datasets from existing large-scale MS datasets. We have constructed a 370K English-to-Chinese

*Corresponding author.

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 3054–3064,
Hong Kong, China, November 3–7, 2019.

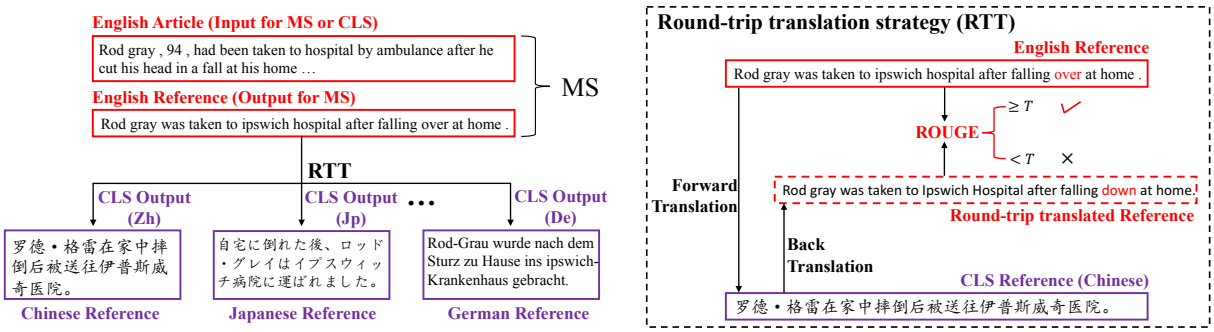

Figure 1: Overview of CLS corpora construction. Our method can be extended to many other language pairs and we focus on En2Zh and Zh2En in this paper. During RTT, we filter the sample in which ROUGE F1 score between the original reference and the round-trip translated reference is below a preset threshold T.

(En2Zh) CLS corpus and a 1.69M Chinese-to-English (Zh2En) CLS corpus.

- To train the CLS systems in an end-to-end manner, we present neural cross-lingual summarization. Furthermore, we propose to improve NCLS by incorporating MT and MS into CLS training process under multi-task learning. To the best of our knowledge, this is the first work to present an end-to-end CLS framework that trained on parallel corpora.

- Experimental results demonstrate that NCLS can achieve +4.87 ROUGE-2 on En2Zh and +5.07 ROUGE-2 on Zh2En over traditional pipeline paradigm. In addition, NCLS with multi-task learning can further achieve +3.60 ROUGE-2 on En2Zh and +0.72 ROUGE-2 on Zh2En. Our methods can be regarded as a benchmark for further NCLS studying.

2 Dataset Construction

Existing large-scale monolingual summarization datasets are automatically collected from the internet. CNN/Dailymail (Hermann et al., 2015) dataset has been collected from *CNN* and *DailyMail* websites, where the article and news highlights are treated as the input and output respectively. Similar to Hermann et al. (2015), Zhu et al. (2018) have constructed a multimodal summarization dataset MSMO where the text input and output are similar to that in CNN/Dailymail. We refer to the union set of CNN/Dailymail and MSMO as ENSUM[1]. Hu et al. (2015) introduce a large-scale corpus of Chinese short text summarization (LCSTS[2]) dataset constructed from the Chinese microblogging website *Sina Weibo*. In this section, we introduce how to construct the En2Zh and Zh2En CLS datasets based on ENSUM and LCSTS respectively.

Round-trip translation strategy. Round-trip translation[3] (RTT) is the process of translating a text into another language (forward translation), then translating the result back into the original language (back translation), using MT service[4]. Inspired by Lample et al. (2018), we propose to adopt the round-trip translation to acquire CLS dataset from MS dataset. The process of constructing our corpora is shown in Figure 1.

Taking the construction of En2Zh corpus as an example, given a document-summary pair $(D_{\text{en}}, S_{\text{en}})$, we first translate the summary S_{en} into Chinese S_{zh} and then back into English S'_{en}. The En2Zh document-summary pair $(D_{\text{en}}, S_{\text{zh}})$, which satisfies ROUGE-1$(S_{\text{en}}, S'_{\text{en}}) \geqslant T_1$ and ROUGE-2$(S_{\text{en}}, S'_{\text{en}}) \geqslant T_2$ (T_1 is set to 0.45 for English and 0.6 for Chinese respectively, and T_2 is set to 0.2 here[5]), will be regarded as a positive pair. Otherwise, the pair will be filtered. Note that there are multiple sentences in S_{en} in ENSUM, we apply the RTT to filter low-quality translated reference sentence by sentence. Once more than two-thirds of the sentences in the summary in a sample are retained, we will keep the sample. This process helps to ensure that the final compression ratio in our task does not differ too much from the actual compression ratio. Similar process is used on constructing Zh2En corpus. The ROUGE scores between Chinese sentences are calculated using Chinese characters as segmentation units.

[1]It contains 626,634 English summarization pairs.
[2]It contains 2,400,591 Chinese summarization pairs.
[3]https://en.wikipedia.org/wiki/Round-trip_translation
[4]http://www.anylangtech.com
[5]The values are obtained by conducting a manual estimation on some samples randomly selected from two corpora.

En2ZhSum	train	valid	test	Zh2EnSum	train	valid	test
#Documents	364,687	3,000	3,000	#Documents	1,693,713	3,000	3,000
#AvgWords (S)	755.09	759.55	744.84	#AvgChars (S)	103.59	103.56	140.06
#AvgEnWords (R)	55.21	55.28	54.76	#AvgZhChars (R)	17.94	18.00	18.08
#AvgZhChars (R)	95.96	96.05	95.33	#AvgEnWords (R)	13.70	13.74	13.84
#AvgSentsWords	19.62	19.63	19.61	#AvgSentsChars	52.73	52.41	53.38
#AvgSents	40.62	41.08	40.25	#AvgSents	2.32	2.33	2.30

Table 1: Corpus statistics. **#AvgWords (S)** is the average number of English words in the source document. Each reference has a bilingual version since each reference in CLS corpus is translated from the corresponding reference in the MS corpus. **#AvgEnWords (R)** means the average number of words in English reference and **#AvgZhChars (R)** denotes the average number of characters in Chinese reference. **#AvgSentsWords** (**#AvgSentsChars**) indicates the average number of words (characters) in a sentence in the source document. **#AvgSents** refers to the average number of sentences in the source document.

Corpus Statistics. After conducting the round-trip translation strategy, we have obtained 370,759 En2Zh CLS pairs from ENSUM and 1,699,713 Zh2En CLS pairs from LCSTS. The statistics of En2Zh corpus (En2ZhSum) and Zh2En corpus (Zh2EnSum) are presented in Table 1. In order to evaluate various CLS methods more reliably, we recruit 10 volunteers to correct the reference in the test sets in two constructed corpora.

3 Approach

The traditional approaches (Section 3.1) intuitively treat CLS as a pipeline process which leads to error propagation. To handle that, we present the neural cross-lingual summarization methods (Section 3.2), which train CLS in an end-to-end manner for the first time. Due to the strong relationship between CLS, MS, and MT tasks, we propose to incorporate MS and MT into CLS training under multi-task learning (Section 3.3).

3.1 Baseline Pipeline Methods

In general, traditional CLS is composed of summarization step and translation step. The different order of these two steps leads to the following two strategies. Take En2Zh CLS as an example.

Early Translation (ETran). This strategy first translates the English document to Chinese document with machine translation. Then a Chinese summary is generated by a summarization model.

Late Translation (LTran). This strategy first summarizes the English document to a short English summary and then translates it into Chinese.

3.2 Neural Cross-Lingual Summarization

Considering the excellent text generation performance of Transformer encoder-decoder network (Vaswani et al., 2017), we implement our NCLS models entirely based on this framework in this work. As shown in Figure 2, given a set of CLS data $D = (X^{(i)}, Y^{(i)})$ where both X and Y are a sequence of tokens, the encoder maps the input document $X = (x_1, x_2, \cdots, x_n)$ into a sequence of continuous representations $z = (z_1, z_2, \cdots, z_n)$ whose size varies with respect to the source sequence length. The decoder generates a summary $Y = (y_1, y_2, \cdots, y_m)$, which is in a different language, from the continuous representations. The encoder and decoder are trained jointly to maximize the conditional probability of target sequence given a source sequence:

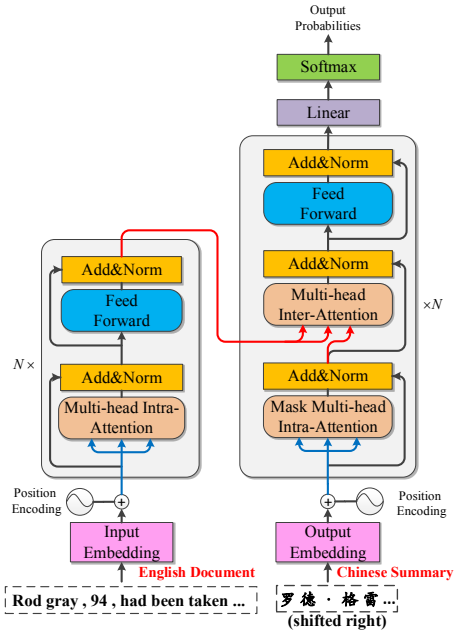

Figure 2: Transformer-based NCLS models (TNCLS).

$$L_\theta = \sum_{t=1}^{N} \log \mathrm{P}(\mathrm{y_t} | \mathrm{y_{<t}}, \mathrm{x}; \theta) \quad (1)$$

Transformer is composed of stacked encoder and decoder layers. Consisting of two blocks, the

encoder layer is a self-attention block followed by a position-wise feed-forward block. Despite the same architecture as the encoder layer, the decoder layer has an extra encoder-decoder attention block. Residual connection and layer normalization are used around each block. In addition, the self-attention block in the decoder is modified with masking to prevent present positions from attending to future positions during training.

For self-attention and encoder-decoder attention, a multi-head attention block is used to obtain information from different representation subspaces at different positions. Each head corresponds to a scaled dot-product attention, which operates on the query Q, key K, and value V:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \quad (2)$$

where d_k is the dimension of the key.

Finally, the output values are concatenated and projected by a feed-forward layer to get final values:

$$\begin{aligned}\text{MultiHead}(Q, K, V) &= \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O \\ \text{where head}_i &= \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)\end{aligned} \quad (3)$$

where W^O, QW_i^Q, KW_i^K, and VW_i^V are learnable matrices, h is the number of heads.

3.3 Improving NCLS with MS and MT

Considering there is a strong relationship between CLS task and MS task, as well as between CLS task and MT task: (1) CLS shares the same goal with MS, i.e., to grasp the core idea of the original document, but the final results are presented in different languages. (2) From the perspective of information compression, machine translation can be regarded as a special kind of cross-lingual summarization with a compression ratio of 1:1. Therefore, we consider using MS and MT datasets to further improve the performance of CLS task under multi-task learning.

Inspired by Luong et al. (2016), we employ the one-to-many scheme to incorporate the training process of MS and MT into that of CLS. As shown in Figure 3, this scheme involves one encoder and multiple decoders for tasks in which the encoder can be shared. We study two different task combinations here: CLS+MS and CLS+MT.

CLS+MS. Note that the reference in each of CLS datasets has a bilingual version. For instance, En2ZhSum dataset contains a total of 370,687 documents with corresponding summaries in both

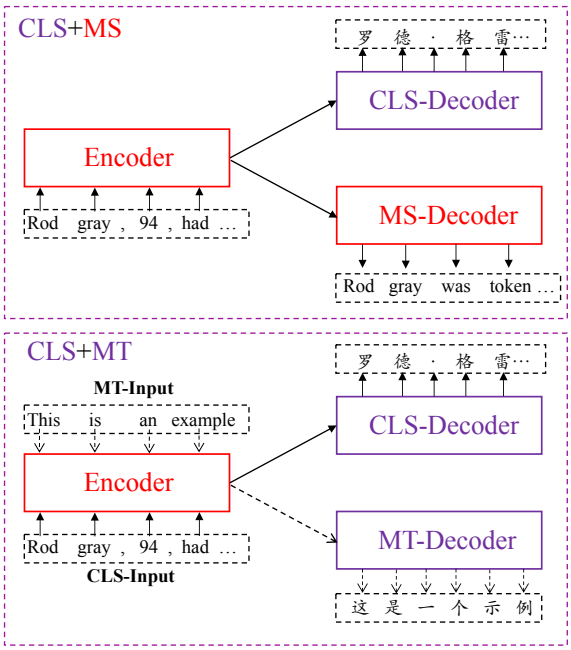

Figure 3: Overview of multi-task NCLS. The lower half is CLS+MT using alternating training strategy. Different colors represent different languages.

Chinese and English. Thus, we consider jointly training CLS and MS as follows. Given a source document, the encoder encodes it into continuous representations, and then the two decoders simultaneously generate the output of their respective tasks. The loss can be calculated as follows:

$$L_\theta = \sum_{t=1}^{N^{(1)}} \log P(y_t^{(1)} | y_{<t}^{(1)}, x; \theta) + \sum_{t=1}^{N^{(2)}} \log P(y_t^{(2)} | y_{<t}^{(2)}, x; \theta) \quad (4)$$

where $y^{(1)}$ and $y^{(2)}$ are the outputs of two tasks.

CLS+MT. Since CLS input-output pairs are different from MT input-output pairs, we consider adopting the alternating training strategy (Dong et al., 2015), which optimizes each task for a fixed number of mini-batches before switching to the next task, to jointly train CLS and MT. For MT task, we employ 2.08M[6] sentence pairs from LDC corpora with CLS dataset to train CLS+MT.

4 Experiments

4.1 Experimental Settings

For English, we apply two different granularities of segmentation, i.e., words and subwords (Sennrich et al., 2016). We lowercase all English characters. We truncate the input to 200 words and the output to 120 words (150 characters for Chinese output) . For Chinese, we employ three different

[6]LDC2000T50, LDC2002L27, LDC2002T01, LDC2002E18, LDC2003E07, LDC2003E14, LDC2003T17, LDC2004T07

Model	ROUGE-1	ROUGE-2	ROUGE-L
Gu et al. (2016)	35.00	22.30	32.00
Li et al. (2017)	36.99	24.15	34.21
Transformer	39.71	27.45	37.13

Table 2: Performance of our implemented transformer-based monolingual summarization model on LCSTS.

granularities of segmentation: characters, words, and subwords. It is worth noting that we only apply subword-based segmentation in Zh2En model since subword-based segmentation will make the English article much longer in En2Zh (especially at the Chinese target-side output), which makes the Transformer performs extremely poor. For our baseline pipeline models, the vocabulary size of Chinese characters is 10,000, and that of Chinese words, Chinese subwords, and English words are all 100,000. In our En2Zh NCLS models, the vocabulary size of source-side English words is 100,000, and that of target-side Chinese characters and words are 18,000, and 50,000 respectively. In our Zh2En models, the vocabulary size of source-side Chinese characters, words, and subwords are 10,000, 100,000, and 100,000 respectively, and that of target-side English words and subwords are all 40,000. We initialize all the parameters via Xavier initialization methods (Glorot and Bengio, 2010). We train our models using configuration *transformer_base* (Vaswani et al., 2017), which contains a 6-layer encoder and a 6-layer decoder with 512-dimensional hidden representations.

During training, in En2Zh models, each mini-batch contains a set of document-summary pairs with roughly 2,048 source and 2,048 target tokens; in Zh2En models, each mini-batch contains a set of document-summary pairs with roughly 4,096 source and 4,096 target tokens. We use Adam optimizer (Kingma and Ba, 2015) with $\beta_1 = 0.9$, $\beta_2 = 0.998$, and $\epsilon = 10^{-9}$. We use a single NVIDIA TITAN X to train our models. Convergence is reached within 1,000,000 iterations in both TNCLS models and baseline models. We train each task for about 800,000 iterations in multi-task NCLS models (reaching convergence). At test time, our summaries are produced using beam search with beam size 4.

4.2 Baselines and Model Variants

We compare our NCLS models with the following two traditional methods:

TETran: We first translate the source document via a Transformer-based machine translation model trained on LDC corpora. Then we employ LexRank (Erkan and Radev, 2004), a strong and widely used unsupervised summarization method, to summarize the translated document. The reason why we choose to apply an unsupervised method is that we lack the version of MS dataset in the target language to train a supervised model to summarize the translated document.

Model	ROUGE-1	ROUGE-2	ROUGE-L
See et al. (2017)	39.53	17.28	36.38
Transformer	39.24	16.67	36.42

Table 3: Performance of our implemented transformer-based MS model on CNN/Dailymail.

TLTran: We first build a Transformer-based MS model which is trained on the original MS dataset. Then the MS model aims to summarize the source document into a summary. Finally, we translate the summary into target language by using the Transformer-based machine translation model trained on LDC corpora. The performance of our transformer-based MS models is given in Table 2 and Table 3.

To make our experiments more comprehensive, during the process of TETran and TLTran, we replace the Transformer-based machine translation model with Google Translator[7], which is one of the state-of-the-art machine translation systems. We refer to these two methods as **GETran** and **GLTran** respectively.

There are three variants of our NCLS models:

TNCLS: Transformer-based NCLS models where the input and output are different granularities combinations of units.

CLS+MS: It refers to the multi-task NCLS model which accepts an input text and simultaneously performs text generation for both CLS and MS tasks and calculates the total losses.

CLS+MT: It trains CLS and MT tasks via alternating training strategy. Specifically, we optimize the CLS task in a mini-batch, and we optimize the MT task in the next mini-batch.

4.3 Experimental Results and Analysis

Comparison between NCLS with baselines. We evaluate different models with the standard ROUGE metric (Lin, 2004), reporting the F1 scores for ROUGE-1, ROUGE-2, and ROUGE-L. The results are presented in Table 4.

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

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

Model	Unit	En2ZhSum	En2ZhSum*	Zh2EnSum	Zh2EnSum*
		RG1-RG2-RGL(↑)	RG1-RG2-RGL(↑)	RG1-RG2-RGL(↑)	RG1-RG2-RGL(↑)
TETran	–	**26.12-10.59-23.21**	**26.15-10.60-23.24**	**22.81- 7.17-18.55**	**23.09- 7.33-18.74**
GETran	–	**28.17-11.38-25.75**	**28.19-11.40-25.77**	**24.03- 8.91-19.92**	**24.34- 9.14-20.13**
TLTran	c-c	–	–	32.85-15.34-29.21	33.01-15.43-29.32
	w-w	**30.20-12.20-27.02**	**30.22-12.20-27.04**	31.11-13.23-27.55	31.38-13.42-27.69
	sw-sw	–	–	**33.64-15.58-29.74**	**33.92-15.81-29.86**
GLTran	c-c	–	–	34.44-15.71-30.13	34.58-16.01-30.25
	w-w	**32.15-13.84-29.42**	**32.17-13.85-29.43**	32.42-15.19-28.75	32.52-15.39-28.88
	sw-sw	–	–	**35.28-16.59-31.08**	**35.45-16.86-31.28**
TNCLS	c-w	–	–	36.36-19.74-32.66	35.82-19.04-32.06
	w-c	**36.83-18.76-33.22**	**36.82-18.72-33.20**	–	–
	w-w	33.09-14.85-29.82	33.10-14.83-29.82	38.54-22.34-35.05	37.70-21.15-34.05
	sw-sw	–	–	**39.80-23.15-36.11**	**38.85-21.93-35.05**

Table 4: ROUGE F1 scores (%) on En2ZhSum and Zh2EnSum test sets. En2ZhSum* and Zh2EnSum* are the corresponding human-corrected test sets. *Unit* denotes the granularity combination of text units, where *c* means character, *w* means word, and *sw* means subword. RG refers to ROUGE for short. ↑ indicates that the larger values, the better the results are. Our NCLS models perform significantly better than baseline models by the 95% confidence interval measured by the official ROUGE script[8].

We can find that GLTran outperforms TLTran and GETran outperforms TETran, which indicates that pipeline-based methods perform better when using a stronger machine translation system. Compared with GLTran or GETran, our TNCLS models both achieve significant improvements, which can verify our motivation and demonstrate the efficacy of our constructed corpora.

In En2Zh CLS task, the results of each model on En2ZhSum are similar to those on En2ZhSum*. This is because the original ENSUM dataset comes from the news reports. Existing MT for news reports has excellent performance. Besides, we have pre-filtered samples with low translation quality during dataset construction. Therefore, the quality of the automatic test set is high. TNCLS (*w-c*) performs significantly better than TNCLS (*w-w*). This is because the character-based segmentation can greatly reduce the vocabulary size at the Chinese target-side, which leads to generating nearly no UNK token during the decoding process.

In Zh2En CLS task, the subword-based models outperform others since subword-based segmentation can greatly reduce the vocabulary size and the generation of UNK. Compared with baselines, TNCLS can achieve maximum improvement up to **+4.52 ROUGE-1**, **+6.56 ROUGE-2**, **+5.03 ROUGE-L** on Zh2EnSum and **+3.40 ROUGE-1**, **+5.07 ROUGE-2**, **+3.77 ROUGE-L** on Zh2EnSum*. The results of TNCLS drops obviously on the human-corrected test set, showing that the quality of the translated reference is not as perfect as expected. The reason is straightforward that the original LCSTS dataset comes from social media so that the proportion of abbreviations and omitting punctuation in its text is much higher than in news, resulting in lower translation quality.

In conclusion, TNCLS models significantly outperform the traditional pipeline methods on both En2Zh and Zh2En CLS tasks.

Why Back Translation? To show the influence of filtering the corpus by back translation during the RTT process, we use three kinds of datasets to train our TNCLS models and compare their performance. They are: (a) the CLS dataset obtained by simply employing forward translation on MS dataset (*Non-Filter*); (b) the CLS dataset obtained by a complete RTT process (*Filter*); (c) the dataset obtained by sampled from *Non-Filter* dataset to keep the same size as the *Filter* dataset (*Pseudo-Filter*). The results are given in Table 5. The models trained on *Filter* dataset significantly outperform the models trained on *Pseudo-Filter* dataset on both En2Zh and Zh2En tasks, which indicates that the back translation can effectively filter dirty samples and improve the overall quality of corpora, thus boosting the performance of NCLS.

DataVersion	BT?	En2ZhSum	En2ZhSum*	Zh2EnSum	Zh2EnSum*
		RG1-RG2-RGL(↑)	RG1-RG2-RGL(↑)	RG1-RG2-RGL(↑)	RG1-RG2-RGL(↑)
Filter	YES	36.83-18.76-33.22	36.82-18.72-33.20	39.80-23.15-36.11	38.85-21.93-35.05
Pseudo-Filter	NO	36.04-17.80-32.49	36.03-17.78-32.48	35.58-17.93-31.71	35.00-17.37-31.10
Non-Filter	NO	37.62-19.88-33.99	37.62-19.85-33.99	36.51-19.23-32.77	36.03-18.63-32.19

Table 5: Experimental results on different versions of datasets. *Filter* refers to the version of dataset for which we employ RTT strategy to filter. *Non-Filter* denotes the version of the dataset obtained by simply forward translation without filtering process including back translation. *Pseudo-Filter* is the dataset randomly sampled from *Non-Filter* version and is of the same size as *Filter* version. BT refers to back translation in RTT. For En2Zh task, we train the TNCLS (*w-c*). For Zh2En task, we train the TNCLS (*sw-sw*).

Model	En2ZhSum	En2ZhSum*	Zh2EnSum	Zh2EnSum*
	RG1-RG2-RGL(↑)	RG1-RG2-RGL(↑)	RG1-RG2-RGL(↑)	RG1-RG2-RGL(↑)
TNCLS	**36.83-18.76-33.22**	**36.82-18.72-33.20**	**39.80-23.15-36.11**	**38.85-21.93-35.05**
CLS+MS	38.23-20.21-34.76	38.25-20.20-34.76	41.08-23.67-37.19	**40.34-22.65-36.39**
CLS+MT	**40.24-22.36-36.61**	**40.23-22.32-36.59**	**41.09-23.70-37.17**	40.25-22.58-36.21

Table 6: Results of multi-task NCLS. The granularity combination of input and output in En2Zh task is "word to character" (*w-c*), and that in Zh2En task is "subword to subword" (*sw-sw*).

In En2Zh task, the model trained on *Non-Filter* dataset performs best. The reasons are two-fold: (1) the quality of machine translation for English news is reliable; (2) the scale of *Non-Filter* dataset is almost twice that of the two others so that after the amount of data reaches a certain level, it can make up for the noises caused by the translation error in the corpus. In Zh2En task, the performance of the model trained on *Non-Filter* dataset is not as good as that on *Filter*. It can be attributed to the fact that current MT is not very ideal in the translation of texts on social media so that the dataset constructed by only using forward translation contains too many noises. Therefore, when the quality of machine translation is not that ideal, backward translation is especially important during the process of constructing corpus.

Results of Multi-task NCLS. To explore whether MS and MT can further improve NCLS, we compare the multi-task NCLS with NCLS using one same granularity combination of units. The results are given in Table 6. As shown in Table 6, both CLS+MS and CLS+MT can improve the performance of NCLS, which can be attributed to that the encoder is enhanced by incorporating MS and MT data into the training process. CLS+MT significantly outperforms CLS+MS in En2Zh task while CLS+MS performs comparably with CLS+MT in Zh2En task. The reasons are two-fold: (1) In En2Zh task, MT dataset is much larger than both MS and CLS datasets, which makes it more necessary for enhancing the robustness of encoder. (2) We use the LDC MT dataset, which belongs to the news domain similar to our En2ZhSum, during the training of CLS+MT. However, Zh2EnSum belongs to social media domain, thus resulting in the greater improvement of CLS+MT in En2Zh than in Zh2En. In general, NCLS with multi-task learning achieves more significant improvement in En2Zh task than in Zh2En task, which illustrates that extra dataset in other related tasks is essentially important for boosting the performance when CLS dataset is not very large.

Human Evaluation. We conduct the human evaluation on 25 random samples from each of the En2ZhSum and Zh2EnSum test set. We compare the summaries generated by our methods (including TNCLS, CLS+MS, and CLS+MT) with the summaries generated by GLTran. Three graduate students are asked to compare the generated summaries with human-corrected references, and assess each summary from three independent perspectives: (1) How informative the summary is? (2) How concise the summary is? (3) How fluent, grammatical the summary is? Each property is assessed with a score from 1 (worst) to 5 (best). The average results are presented in Table 7.

As shown in Table 7, TNCLS can generate more informative summaries compared with GLTran, which shows the advantage of end-to-end models. The conciseness score and fluency score of TNCLS are comparable to those of GLTran. This is because both GLTtrans and TNCLS employ a single encoder-decoder model, which eas-

Input (Chinese): 在成本压力加大的情况下，流通企业不仅没有缩减IT投资反而继续增加。2011年，中国流通行业的IT投资规模由2010年的96.6亿元增加至2011年的109.2亿元。预计2012年流通行业的IT投资增速将达14.1%，规模超120亿元。 Under the circumstance of increasing cost pressure, circulation enterprises not only did not reduce IT investment but also continued to increase. In 2011, the scale of IT investment in China 's circulation industry increased from 9.66 billion yuan in 2010 to 10.92 billion yuan in 2011. It is estimated that the IT investment in the circulation industry will grow by 14.1% in 2012, with a scale exceeding 12 billion yuan.
Gold Summary: in 2012 , the scale of it investment in china 's circulation industry will exceed 12 billion yuan .
GETran: in the case of increased cost pressures, distribution companies have not only reduced it investment but continued to increase.
GLTran: it investment in china 's circulation industry will increase by 14.1 % in 2011
TNCLS: it investment in circulation industry continues to increase
CLS+MS: it investment in china 's circulation industry will exceed 12 billion yuan in 2012 .
CLS+MT: china 's circulation industry is expected to increase it investment by 14.1 % in 2012 .

Figure 4: Examples of generated summaries.

Model	En2Zh			Zh2En		
	IF	CC	FL	IF	CC	FL
GLTran	3.06	3.37	3.13	3.53	4.21	4.25
TNCLS	3.25	3.33	3.17	3.67	4.25	4.24
CLS+MS	3.53	3.58	3.53	3.72	4.31	4.28
CLS+MT	**3.58**	**3.76**	**3.63**	**3.78**	**4.43**	**4.35**

Table 7: Human evaluation results. IF, CC and FL denote informative, concise, and fluent respectively.

ily leads to under-generation and repetition. Our CLS+MS and CLS+MT can significantly improve the conciseness and fluency of generated summaries, which shows that these methods can generate shorter summaries and reduce grammatical errors. In conclusion, TNCLS can generate more informative summaries, but it is difficult to improve the conciseness and fluency. However, with the help of MT and MS tasks, conciseness and fluency scores can be significantly improved.

4.4 Case Study

We show the case study of a sample from the Zh2EnSum human-corrected test set in Figure 4. As shown in Figure 4, the summary generated by **GETran** obviously suffers from errors of machine translation ("distribution companies" should be corrected as "circulation enterprises"). Since **GETran** first translates all the source text, it is easier to bring the errors from machine translation. The **GLTran**-generated summary contracts the fact that the year in it should be 2012 instead of 2011. The translation quality of the sentence is relatively reliable, thus the errors are probably produced during the summarization step. Compared with the first two generated summaries, although the summary produced by **TNCLS** does not emphasize the time and place of occurrence, there is no mistake in the logic of its expression. The summaries generated by **CLS+MS** and **CLS+MT** are generally consistent with the facts, but their emphases are different. The **CLS+MS** summary matches the gold summary better. The flaws of both of them are that they do not reflect the "scale" in the original text. In conclusion, our methods can produce more accurate summaries than baselines.

5 Related Work

Cross-lingual summarization has been proposed to present the most salient information of a source document in a different language, which is very important in the field of multilingual information processing. Most of the existing methods handle the task of CLS via simply applying two typical translation schemes, i.e., early translation (Leuski et al., 2003; Ouyang et al., 2019) and late translation (Orasan and Chiorean, 2008; Wan et al., 2010). The early translation scheme first translates the original document into target language and then generates the summary of the translated document. The late translation scheme first summarizes the original document into a summary in the source language and then translates it into target language.

Leuski et al. (2003) translate the Hindi document to English and then generate the English headline for it. Ouyang et al. (2019) present a robust abstractive summarization system for low resource languages where no summarization corpora are currently available. They train a neural abstractive summarization model on noisy English documents and clean English reference summaries. Then the model can learn to produce fluent summaries from disfluent inputs, which allows generating summaries for translated documents. Orasan and Chiorean (2008) summarize the Romanian news with the maximal marginal relevance method (Goldstein et al., 2000) and produce the English summaries for English speakers. Wan et al. (2010) adopt the late translation scheme for the task of English-to-Chinese CLS. They extract English sentences considering both the informativeness and translation quality of sentences and automatically translate the English summary into the final Chinese summary. The above researches only make use of the information from only one language side.

Some methods have been proposed to improve CLS with bilingual information. Wan (2011) proposes two graph-based summarization methods to leverage both the English-side and Chinese-side information in the task of English-to-Chinese CLS. Inspired by the phrase-based translation models, Yao et al. (2015) introduce a compressive CLS, which simultaneously performs sentence selection and compression. They calculate the sentence scores based on the aligned bilingual phrases obtained by MT service and perform compression via deleting redundant or poorly translated phrases. Zhang et al. (2016) propose an abstractive CLS which constructs a pool of bilingual concepts represented by the bilingual elements of the source-side predicate-argument structures (PAS) and the target-side counterparts. The final summary is generated by maximizing both the salience and translation quality of the PAS elements.

However, all these researches belong to the pipeline paradigm which not only relies heavily on hand-crafted features but also causes error propagation. End-to-end deep learning has proven to be able to alleviate these two problems, while it has been absent due to the lack of large-scale training data. Recently, Ayana et al. (2018) present zero-shot cross-lingual headline generation based on existing parallel corpora of translation and monolingual headline generation. Similarly, Duan et al. (2019) propose to use monolingual abstractive sentence summarization system to teach zero-shot cross-lingual abstractive sentence summarization on both summary word generation and attention. Although great efforts have been made in cross-lingual summarization, how to automatically build a high-quality large-scale cross-lingual summarization dataset remains unexplored.

In this paper, we focus on English-to-Chinese and Chinese-to-English CLS and try to automatically construct two large-scale corpora respectively. In addition, based on the two corpora, we perform several end-to-end training methods noted as Neural Cross-Lingual Summarization.

6 Conclusion and Future Work

In this paper, we present neural cross-lingual summarization for the first time. To achieve that goal, we propose to acquire large-scale supervised data from existing monolingual summarization datasets via round-trip translation strategy. Then we apply end-to-end methods on our constructed datasets and find our NCLS models significantly outperform the traditional pipeline paradigm. Furthermore, we consider utilizing machine translation and monolingual summarization to further improve NCLS. Experimental results have shown that both machine translation and monolingual summarization can significantly help NCLS generate better summaries.

In our future work, we will adopt our RTT strategy to obtain CLS datasets of other language pairs, such as English-to-Japanese, English-to-German, Chinese-to-Japanese, and Chinese-to-German, etc.

7 Acknowledgments

The research work described in this paper has been supported by the National Key Research and Development Program of China under Grant No. 2016QY02D0303. We thank the three anonymous reviewers for their careful reading of our paper and their many insightful comments and suggestions. We would like to thank He Bai, Yuchen Liu, Haitao Lin, Yang Zhao, Cong Ma, Lu Xiang, Weikang Wang, Zhen Wang, and Jiaqi Liang for their invaluable contributions in shaping the early stage of this work. We thank Xina Fu, Jinliang Lu, and Sikai Liu for conducting human evaluation.

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.

Rich Caruana. 1997. Multitask learning. *Machine Learning*, 28(1):41–75.

Daxiang Dong, Hua Wu, Wei He, Dianhai Yu, and Haifeng Wang. 2015. Multi-task learning for multiple language translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 1723–1732.

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 the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 3162–3172.

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 the International Conference on Artificial Intelligence and Statistics (AISTATS10). Society for Artificial Intelligence and Statistics*.

Jade Goldstein, Vibhu Mittal, Jaime Carbonell, and Mark Kantrowitz. 2000. Multi-document summarization by sentence extraction. In *Proceedings of the NAACL-ANLP Workshop on Automatic summarization*, pages 40–48.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1631–1640.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1693–1701.

Baotian Hu, Qingcai Chen, and Fangze Zhu. 2015. LCSTS: A large scale Chinese short text summarization dataset. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1967–1972.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Unsupervised machine translation using monolingual corpora only. In *International Conference on Learning Representations (ICLR)*.

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 (TALIP)*, 2(3):245–269.

Piji Li, Wai Lam, Lidong Bing, and Zihao Wang. 2017. Deep recurrent generative decoder for abstractive text summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2091–2100.

Jung-Min Lim, In-Su Kang, and Jong-Hyeok Lee. 2004. Multi-document summarization using cross-language texts. In *Proceedings of the Fourth NTCIR Workshop on Research in Information Access Technologies Information Retrieval, Question Answering and Summarization (NTCIR)*.

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

Thang Luong, Quoc V. Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2016. Multi-task sequence to sequence learning. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Constantin Orasan and Oana Andreea Chiorean. 2008. Evaluation of a cross-lingual Romanian-English multi-document summariser. In *Proceedings of Language Resources and Evaluation Conference (LREC)*.

Jessica Ouyang, Boya Song, and Kathy McKeown. 2019. A robust abstractive system for cross-lingual summarization. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 2025–2031.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1073–1083.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1715–1725.

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 *Advances in Neural Information Processing Systems (NIPS)*, pages 5998–6008.

Xiaojun Wan. 2011. Using bilingual information for cross-language document summarization. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1546–1555.

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

Jin-ge Yao, Xiaojun Wan, and Jianguo Xiao. 2015. Phrase-based compressive cross-language summarization. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, pages 118–127.

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

Junnan Zhu, Haoran Li, Tianshang Liu, Yu Zhou, Jiajun Zhang, and Chengqing Zong. 2018. MSMO: Multimodal summarization with multimodal output. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4154–4164.