Composing Elementary Discourse Units in Abstractive Summarization

Zhenwen Li and **Wenhao Wu** and **Sujian Li**
Key Laboratory of Computational Linguistics(MOE)
Department of Computer Science, Peking University
{lizhenwen, waynewu, lisujian}@pku.edu.cn

Abstract

In this paper, we argue that *elementary discourse unit* (EDU) is a more appropriate textual unit of content selection than the sentence unit in abstractive summarization. To well handle the problem of composing EDUs into an informative and fluent summary, we propose a novel summarization method that first designs an EDU selection model to extract and group informative EDUs and then an EDU fusion model to fuse the EDUs in each group into one sentence. We also design the reinforcement learning mechanism to use EDU fusion results to reward the EDU selection action, boosting the final summarization performance. Experiments on *CNN/Daily Mail* have demonstrated the effectiveness of our model.

1 Introduction

Abstractive summarization focuses on generating fluent and concise text from the original input document and has achieved considerable performance improvement with the rapid development of deep learning technology (See et al., 2017; Paulus et al., 2017; Celikyilmaz et al., 2018; Gehrmann et al., 2018). In abstractive summarization, the recently popular and practical paradigm usually generates summary sentences by independently compressing or rewriting each pre-extracted sentence, which is from the source documents (Chen and Bansal, 2018; Lebanoff et al., 2019).

However, a single document sentence usually cannot provide enough information that a summary sentence expresses, which is supported by the recent study of Lebanoff et al. (2019). They show that a high percentage of summary sentences include information from more than one document sentences, and composing a summary through only compressing sentences can cause performance degradation. Simultaneously, in contrast to the brevity requirements of a summary, each document sentence usually offers trivial details and expresses a relatively independent meaning, posing difficulty of combining multiple sentences into one summary sentence. So we hope to seek a new summary composition unit which is more information-intensive and elementary than sentence.

In this paper, we choose to use Elementary Discourse Unit (EDU) as the summarization unit, which is first proposed from Rhetorical Structure Theory (Mann and Thompson, 1988) and defined as a clause. The finer granularity makes EDU more suitable than sentence to be the basic summary composition unit (Li et al., 2016). At the same time, benefited from the development of EDU segmentation technology, which can achieve a high accuracy of 94% (Wang et al., 2018), it is feasible to automatically obtain EDUs from the text. Next, the problems are: (1) which EDUs should be selected to compose a good summary? Moreover, (2) how to well assemble the selected EDUs into a fluent summary?

To solve the problems above, we need to extract the information-intensive EDUs from the source documents and effectively fuse the related EDUs into fluent summary sentences. With such an idea, inspired by Chen and Bansal (2018)'s work, we design an abstractive summarization method which is composed of two parts: EDU selection and EDU fusion. EDU selection aims to extract informative EDUs and group them while EDU fusion takes the grouped EDUs as input to generate a sentence. As the EDU selection process lacks labeling training data, we apply the EDU fusion results as the feedback to tune the EDU selection model which in turn influences the EDU fusion process. Here, the actor-critic reinforcement learning algorithm is employed to train our EDU-based summarization method. To the best of our knowledge, we are the first to propose a practical solution to compose EDUs in summarization. Experiments show that compared to previous models, our EDU based model achieves a significant improvement on the

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 6191–6196

CNN/Daily Mail dataset.

2 Model

Our model is mainly composed of two modules: *EDU Selection* and *EDU Fusion*. *EDU Selection* aims to extract salient EDUs from the source document and group the closely related EDUs. Here, we adopt a smart unified end-to-end method to implement both the extraction and grouping. Next, *EDU Fusion* takes the EDUs in a group to generate a fluent and informative sentence. To train our method, we adopt reinforcement learning to leverage both the two modules. Figure 1 shows the whole architecture of our method.

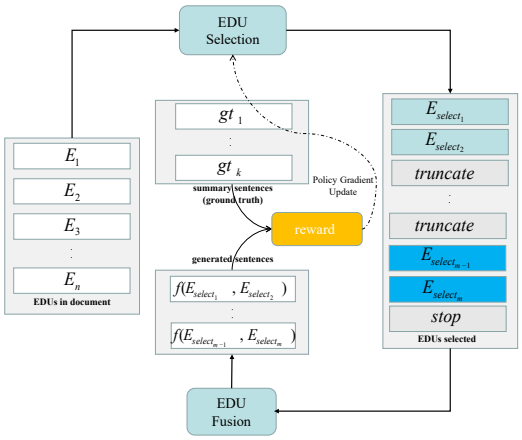

Figure 1: Overall Architecture of Our Model

2.1 EDU Selection

The EDU selection model is mainly based on a sequence-to-sequence pointer network. In the encoding stage, we use a hierarchical encoder to get the contextual representation of each EDU, which consists of a word-level temporal convolutional neural network (Kim, 2014) and an EDU-level Bidirectional Long Short-Term Memory Network(Bi-LSTM) (Hochreiter and Schmidhuber, 1997).

In the decoding stage, we design an LSTM decoder to identify the informative EDUs with their group information. To group the related EDUs, we design a particular label *truncate* whose representation is a trainable parameter $h_{truncate}$. We also add another special label *stop* with its representation h_{stop} to determine the end of the selection process. $h_{truncate}$ and h_{stop} are first randomly initialized and then learned in the training process. In each decoding step, the decoder computes a selection probability distribution on EDUs, $truncate$ and $stop$. Assuming at time step t, the indices of the EDUs which have been extracted are included in the set Sel_t, the decoder first uses the Luong attention (Luong et al., 2015) to get the context c_t and then computes a score s_i^t for each EDU or label by:

$$s_i^t = \begin{cases} v_p^{\mathrm{T}} tanh(W_p[c_t; h_i]) & i \ not \ in \ Sel_t \\ -\infty & \text{otherwise} \end{cases} \quad (1)$$

where i represents the index of an EDU, $truncate$ or $stop$, and h_i denotes the corresponding representation. v_p and W_p are the trainable parameters. In order to avoid repeated selection of the same EDUs, we assign the score of $-\infty$ to the EDUs that have been extracted. It is noted that the label $truncate$ can be generated multiple times since it is not included in Sel_t. Finally, we get the selection probability at time step t by applying softmax to regularize the scores.

Once the decoder selects the $stop$ label, it stops the selection process and gets a sequence which is composed of EDUs, $truncate$ labels and one $stop$ label. Next, the EDUs separated by $truncate$ are grouped for fusion.

2.2 EDU Fusion

The EDU fusion module uses the standard pointer generator (See et al., 2017) to generate one sentence for each group of EDUs. This design allows the model to directly copy words from the inputted EDUs to the generated sentence, which is beneficial to keeping the cross-sentence information in the source documents. At the same time, benefited from the conditional language model training objective, the coherence of the generated sentences is highly improved to remedy the poor readability of EDUs.

To leverage EDU selection and fusion for generating a good summary, reinforcement learning mechanism is designed to use EDU fusion results to tune the selection process, which in turn affects the fusion performance. We introduce the learning process detailedly in Section 3.

3 Learning

We firstly pre-train the EDU selection and EDU fusion module separately and then use the pre-trained model as initialization for reinforcement learning(RL).

3.1 Model Pretraining

Because the summarization datasets do not label the salient EDUs, we propose a greedy method to provide the labeled data for pre-training. For each pair of the document and summary, we select several groups of EDUs from the document as the oracle EDU labels, with each group corresponding to a summary sentence. For each summary sentence, we construct a group of EDUs iteratively. We start from an empty group and repeatedly select the EDU from the document that can maximize the ROUGE-L$_{recall}$ score between the ground-truth summary sentence and the group of EDUs after the EDU is added into the group until no EDU can increase the score. We use ROUGE-L$_{recall}$ so that the EDU selection module can select as much information as possible for EDU fusion. With such a dataset, we pre-train the EDU selection module. To pre-train the EDU fusion module, the input and output are the concatenation of oracle EDUs and summary sentences. We pre-train the two modules separately by optimizing maximum likelihood (ML).

3.2 Reinforcement Learning

We use the *Advantage Actor-Critic* (A2C) algorithm to train our model end-to-end. Following Chen and Bansal (2018)'s work, we fix the parameters of the EDU fusion module during RL training. Here, we regard the EDU selection module as the agent whose decoding stage is formulated as a Markov Decision Process (MDP). In each decoding step, the agent executes one selection action, which is selecting an EDU or a label ($truncate$ or $stop$) according to the selection probability. Then the agent gets a reward according to the EDU fusion results. As for reward computation, given the group i of the selected EDUs, we use the EDU fusion module to generate a sentence s_i and compute its score r_i to measure the overlap between s_i and the sentence gt_i in the ground truth summary.

$$r_i = \begin{cases} ROUGE\text{-}L_F(s_i, gt_i) & i \leq n \\ 0 & i > n \end{cases} \quad (2)$$

where n is the number of sentences in the ground truth summary. For each selection action to compose the group, we set its reward as $\frac{r_i}{l_i}$, where l_i is the action number of selecting an EDU or $truncate$. Similar to (Chen and Bansal, 2018), we compute the $ROUGE\text{-}1_F$ score between the ground-truth summary and the whole fused sentences as the reward for the final action that selects the $stop$ label.

Model	R-1	R-2	R-L
Lead-3	40.34	17.70	36.57
NN(2016)	35.5	14.7	32.2
REFRESH	40.0	18.2	36.6
Pointer Generator	39.53	17.28	36.38
Fan et al. (2017)	39.75	17.29	36.54
Fast-Abs	40.88	17.80	38.54
EDUSum$_{sel+RL}$	40.89	**18.30**	37.79
EDUSum	**41.40**	18.03	**38.79**

Table 1: Model Comparison

4 Experiments

4.1 Experiment Setup

We conduct experiments on the non-anonymized version of the *CNN/Daily Mail* dataset (Hermann et al., 2015; See et al., 2017). Using the same processing method as See et al. (2017), the dataset contains 287,226 training pairs, 13,368 validation pairs and 11,490 test pairs. To segment the documents into EDUs, we use Wang et al. (2018)'s model which achieves a 94% F-score in EDU segmentation. To evaluate summarization performance, we use the ROUGE metrics (R-1, R-2 and R-L) (Lin, 2004). For our model, the dimensions of hidden states and word embeddings are set 256 and 128 respectively. The batch size of training is 32, and the discount factor for reward in RL training is set to 0.95. The optimizer is Adam (Kingma and Ba, 2015) with a 0.001 learning rate for pre-training and 0.0001 learning rate for RL training. [1]

4.2 Results

To evaluate model performance, we compare our model (named *EDUSum*) with the state-of-the-art extractive and abstractive summarization methods. Three extractive methods are a strong *Lead*-3 baseline, *NN* (Cheng and Lapata, 2016) which applies neural networks with attention to extract sentences directly, and REFRESH (Narayan et al., 2018) which uses reinforcement learning to rank sentences. Three abstractive methods for comparison include: Pointer Generator (See et al., 2017), a controllable text generation method (Fan et al., 2017), and *Fast-Abs* (Chen and Bansal, 2018) which uses

[1] The source code is available at https://github.com/PKU-TANGENT/EDUSum

Model	R-1	R-2	R-L
EDUSum$_{SameSent}$	41.17	17.84	38.62
EDUSum$_{group-1}$	40.02	17.21	37.76
EDUSum$_{group-2}$	41.09	17.59	38.54
EDUSum$_{group-3}$	40.20	17.06	37.53
EDUSum	**41.40**	**18.03**	**38.79**

Table 2: Ablation Study on EDU Selection Module

Model	Read.	Non-redund.
Fast-Abs	1.86	2.1
EDUSum$_{sel+RL}$	2.22	1.94
EDUSum	1.92	1.96

Table 3: Human Evaluation. The smaller value of the metric of the average rank, the better the performance.

reinforcement learning to extract and rewrite sentences. As we can see in Table 1, EDUSum outperforms all the baselines. Compared to *Fast-Abs* which is similar to *EDUSum* in model architecture, *EDUSum* achieves better performance with respect to the three metrics, showing EDU is more informative than sentence and appropriate to be the basic selection unit in summarization. From the table, we can also see that all the summarization methods with RL achieve comparable performance, meaning the RL mechanism can effectively supervise a system to acquire valuable information. We also design a model *EDUSum*$_{sel+RL}$ which is similar to EDUSum except that it does not include the EDU fusion module and directly concatenates the selected EDUs as a summary. *EDUSum*$_{sel+RL}$ performs worse with respect to R-1 and R-L when the EDU fusion module is removed, because the direct concatenation of EDUs may bring redundancy into the summary and EDU fusion can make the summary sentence more informative. We also note that EDUSum$_{sel+RL}$ performs better than $EDUSum$ with respect to R-2, perhaps because EDU fusion may generate some fake information and need further improvement which will be our future work.

Further, we conduct a thorough analysis of the EDU selection module which is the main component of our method. Compared to previous work, the EDU selection module can automatically determine which EDUs and how many EDUs can be grouped. Such a design is convenient for capturing cross-sentence information effectively. To evaluate whether it is necessary to capture cross-sentence information in summarization, we add a constraint to our model: the EDU selection module can only select those EDUs that belong to the same sentence into the same group. We name this model *EDUSum*$_{SameSent}$. From Table 2, we can see that *EDUSum*$_{SameSent}$ behaves a little worse than *EDUSum*. This makes sense because the content of each summary sentence mostly derive from one source sentence and is supplemented by some information from other sentences. We also evaluate the grouping effects of our model and remove the automatic grouping mechanism by grouping every *K* adjacent selected EDUs into a group. We set *K* as 1, 2, and 3 respectively where the value of 1 means no group at all. Table 2 shows *EDUSum*$_{group-2}$ performs the best among all the size settings, but performs worse than *EDUSum* and *EDUSum*$_{Samesent}$. This means that a summary sentence is usually composed of two EDUs but a hard grouping can degrade the performance.

We also give a summary sentence generated by our method as an example to illustrate the advantage of our model, as in Figure 2. We can see that our model can well select and group the EDUs (the underlined EDUs in Sent. 1 and Sent. 2) which have similar meanings, and fuse the grouped EDUs coherently by grabbing the key entity information (i.e., person and team information in Sent. 1) and combining them into the final summary sentence.

4.3 Human Evaluation

To evaluate the abstractive ability of our method, we conduct a human evaluation on the two aspects of readability and non-redundancy. Readability measures how easy a text is to read, and depends on the elements of grammaticality and coherence. Non-redundancy mainly denotes the degree of linguistic brevity of a text in conveying the main idea. To save labor, we only choose two baselines *Fast-Abs* and *EDUSum*$_{sel+RL}$, which perform well with ROUGE metrics, for comparison. Comparing to scoring, ranking is relatively easy for an annotator to implement and we follow the evaluation method of (Wu and Hu, 2018). We randomly sample 50 test documents and generate their summaries using our model and the two baselines. Three annotators are asked to rank each set of three summaries with respect to readability and non-redundancy. The best is ranked the first while the worst is the third, and the ranks are allowed to be tied. Then we compute the average ranks of the three models, as shown in Table 3. We see that EDUSum can

Original sentences segmented into EDUs	System-generated sentence	Ground truth
Sent 1: [Juan Mata has collected his player of the month award for March from Manchester United] [and was quick to thank his supporters after receiving the gong .] **Sent 2:** [Mata scored both goals as united overturned Liverpool with a 2-1 win at Anfiled.] [while also producing an impressive display in the 3-0 home victory over Tottenham]	Juan Mata scored both goals as Manchester United overturned Liverpool's 2-1 win at Anfield.	Juan Mata scored both times as Manchester United beat Liverpool 2-1.

Figure 2: An example of a generated summary sentence that is fused by cross-sentence EDUs.

well leverage readability and non-redundancy compared to the two baselines. Both *EDUSum* and *EDUSum*$_{sel+RL}$ achieve a significant improvement in non-redundancy, because the fine-grained EDUs can contain more informative cross-sentence information and make the summaries briefer. We can also see *EDUSum*$_{sel+RL}$ suffers from bad readability because it simply concatenates EDUs into a sentence, which is the main problem that EDU based models are faced with. As for *EDUSum*, benefited from EDU fusion, this model can achieve nearly the same readability as the sentence based model *Fast-Abs*.

5 Conclusions

In this paper, we choose EDU as the basic summary unit and propose a novel EDU based summarization model *EDUSum*. In our model, the module of EDU selection is designed to extract and group salient EDUs and the module of EDU fusion to convert groups of EDUs into summary sentences. We also apply reinforcement learning to leverage EDU selection and EDU fusion for improving summarization performance. With such a design, *EDUSum* can fuse cross-sentence information and remedy the poor readability problem brought by EDUs. Compared to previous work, this work has provided a feasible and effective method which makes full use of EDUs in summarization.

Acknowledgements

We thank the anonymous reviewers for their helpful comments on this paper. This work was partially supported by National Key Research and Development Project (2019YFB1704002) and National Natural Science Foundation of China (61876009 and 61572049). The corresponding author of this paper is Sujian Li.

References

Asli Celikyilmaz, Antoine Bosselut, Xiaodong He, and Yejin Choi. 2018. Deep communicating agents for abstractive summarization. *arXiv preprint arXiv:1803.10357*.

Yen-Chun Chen and Mohit Bansal. 2018. Fast abstractive summarization with reinforce-selected sentence rewriting. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–686, Melbourne, Australia. Association for Computational Linguistics.

Jianpeng Cheng and Mirella Lapata. 2016. Neural summarization by extracting sentences and words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 484–494, Berlin, Germany. Association for Computational Linguistics.

Angela Fan, David Grangier, and Michael Auli. 2017. Controllable abstractive summarization. *arXiv preprint arXiv:1711.05217*.

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

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 1693–1701. Curran Associates, Inc.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Logan Lebanoff, Kaiqiang Song, Franck Dernoncourt, Doo Soon Kim, Seokhwan Kim, Walter Chang, and Fei Liu. 2019. Scoring sentence singletons and pairs for abstractive summarization. *arXiv preprint arXiv:1906.00077*.

Junyi Jessy Li, Kapil Thadani, and Amanda Stent. 2016. The role of discourse units in near-extractive summarization. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 137–147, Los Angeles. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Proc. ACL workshop on Text Summarization Branches Out*, page 10.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text - Interdisciplinary Journal for the Study of Discourse, 8(3)*, 8(3):243–281.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Ranking sentences for extractive summarization with reinforcement learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1747–1759, New Orleans, Louisiana. Association for Computational Linguistics.

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

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Yizhong Wang, Sujian Li, and Jingfeng Yang. 2018. Toward fast and accurate neural discourse segmentation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 962–967, Brussels, Belgium. Association for Computational Linguistics.

Yuxiang Wu and Baotian Hu. 2018. Learning to extract coherent summary via deep reinforcement learning. In *Thirty-Second AAAI Conference on Artificial Intelligence*.