Incorporating Linguistic Knowledge for Abstractive Multi-document Summarization

Congbo Ma, Wei Emma Zhang, Hu Wang, Shubham Gupta and Mingyu Guo
The University of Adelaide, Adelaide, Australia
{congbo.ma,wei.e.zhang,hu.wang,a1787223,mingyu.guo}@adelaide.edu.au

Abstract

Linguistic knowledge plays an important role in assisting models to learn informative representations that could help guide better natural language generation. In this paper, we develop a Transformer-based abstractive multi-document summarization method with linguistic-guided attention (LGA) mechanism for better representation learning. The proposed linguistic-guided attention mechanism can be seamlessly incorporated into multiple mainstream Transformer-based summarization models to improve the quality of the generated summaries. We develop the proposed method based on Flat Transformer (FT) and Hierarchical Transformer (HT), named ParsingSum-FT and ParsingSum-HT respectively. Empirical studies on both models demonstrate this simple but effective mechanism can help the models outperform existing Transformer-based methods on the benchmark datasets by a large margin. Extensive analyses examine different settings and configurations of the proposed model, providing a good reference to the text summarization community.

1 Introduction

Multi-document summarization (MDS) is a critical task in natural language processing aiming at generating a informative summary from a set of content-related documents. There are two types of summary generation: extractive summarization by selecting salient sentences from original texts directly and abstractive summarization to generate summaries by models from the understanding of the input contents (Ma et al., 2022). Under comparison, abstractive summarization is more challenging because it requires models to truly understand the input documents and generate corresponding summaries. With the development of deep learning techniques, neural network-based models that can help to capture high-quality latent features are widely applied in MDS (Dhakras et al., 2018; Alexander et al., 2019; Liu et al., 2019; Li et al., 2020; Han et al., 2020; Wen et al., 2021; Beltrachini et al., 2021).

Recently, Transformer (Ashish et al., 2017) shows outstanding performances in various natural language processing tasks, and it is also introduced into MDS (Alexander et al., 2019). Transformer has natural advantages for parallelization and could retain long-range relations between pairs of tokens among documents. Liu et al. (Liu et al., 2018) adopted a Transformer model to generate Wikipedia articles. The model selects top-K tokens and feeds them into the decoder-only sequence transduction. Built upon this work, Liu et al. (Liu et al., 2019) proposed a Hierarchical Transformer (HT) containing token-level and paragraph-level Transformer layers for cross-document relations capturing. Wen et al. (Wen et al., 2021) proposed a pre-train language model PRIMERA, using encoder-decoder transformers to simplify the processing of concatenated input documents, leverages the Longformer (Beltagy et al., 2020) to pre-train with a novel entity-based sentence masking objective. However, computing token-wise self-attention in the Transformer takes pairs of token relations into account but lacks syntactic support that may cause content irrelevance and deviation for summary generation (Jin et al., 2020).

Table 1: Generated summaries via different MDS models. Different colors mean different thought groups.

Source Documents	a girl reported missing more than two years ago when she was 15 told police she escaped a home in illinois ... ... they recovered the child and arrested a 24-year-old man ... ... she was 15 when she disappeared. she escaped from the home in washington park earlier this week and went to police ...
HT	... she was also taken into custody. ...
FT	... the girl , who was 15 when she escaped from a home in washington park earlier this week. ...
ParsingSum-HT (Ours)	... a 24-year-old man were arrested and taken into custody. ...
ParsingSum-FT (Ours)	... she was 15 when she disappeared from the home. ...

Many research works seek to incorporate linguistic knowledge to further improve the quality of summaries. Daniel et al. (Daniel et al., 2007) suggested that linguistic knowledge help improve the informativeness of summaries. Sho et al. (Sho et al., 2016) proposed an attention-based encoder-decoder model that adopts abstract meaning representation parser to capture structural syntactic and semantic information. The authors also pointed out that for natural language generation tasks in general, semantic information obtained from external parsers could help improve the performance of encoder-decoder based neural network model. Patrick et al. (Patrick et al., 2019) adopted named entities and entity coreferences for summarization problem. Jin et al. (Jin et al., 2020) enriched a graph encoder with semantic dependency graph to produce semantic-rich sentence representations. Song et al. (Song et al., 2020) presented a LSTM-based model to generate sentences and the parse trees simultaneously by combining a sequential and a tree-based decoder for abstractive summarization generation.

Dependency parsing, an important linguistic knowledge that retains the intra-sentence syntactic relations between words, has been adopted and shown promising results in a variety natural language processing task (Hiroyuki et al., 2019; Sun et al., 2019; Kai et al., 2020; Cao et al., 2021; Wu et al., 2017). The parsing information is usually formed as a tree structure that offers discriminate syntactic paths on arbitrary sentences for information propagation (Sun et al., 2019). The grammatical structure between the pair of words can be extracted from the dependency parser helping the model retain the syntactic structure. Therefore, in this work, we introduce a generic and flexible framework linguistic guided attention to incorporate dependency information into the Transformer based summarization models. We develop the proposed framework based on Flat Transformer (FT) and Hierarchical Transformer (HT), named ParsingSum-FT and ParsingSum-HT. Our proposed models can also be applied for both single and multiple document summarization.

Table 1 is an example to illustrate why dependency information helps improve the quality of summaries. The data source is from Multi-News dataset (Alexander et al., 2019). The HT model can not distinguish who was arrested: it should be *"a 24-year-old man"* rather than *"she"*. In contrast, ParsingSum-HT (our model) shows consistent content with source documents. The potential reason is that the dependency parsing captures the relation between *"arrested"* and *"man"*, which keeps the token relations for summaries generation. We also find the FT model mingles two events within two sentences. However, the source documents show two events: (1) the disappearance of the girl in Illinois was at her age of 15; (2) she escaped from her Washington Park home two years later. Comparatively, ParsingSum-FT (our model) retains correct information. This is due to, from the linguistic perspective, a sentence is a linguistic unit that has complete meaning (Halliday et al., 2014). Furthermore, dependency parsing focuses on intra-sentence relations that help summaries retain correct syntactic structure. The main contributions of this paper include:

- We propose a simple yet effective linguistic-guided attention mechanism to incorporate dependency relations with multi-head attention. The proposed linguistic-guided attention can be seamlessly incorporated into multiple mainstream Transformer-based summarization models to improve their performances.

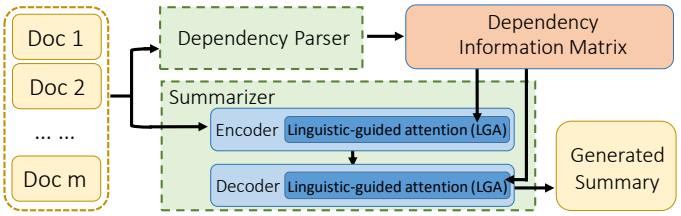


Figure 1: The framework of ParsingSum. The set of documents are first fed into the encoder to generate the representations. In the meantime, these documents are input to a dependency parser to produce their sentence dependency information. The dependency information matrix will be further processed into a linguistic-guided attention mechanism and then fused with Transformer's multi-head attention to guide the downstream summary generation.

- We evaluate and compare the proposed model with several strong techniques. The results of automatic and human evaluation demonstrate that the models equipped with the linguistic-guided attention receive better performances over the compared models.

- We provide an extensive analysis of various settings and configurations of the proposed model. These results can help researchers understand the intuition of ParsingSum and serve as an informative reference for the summarization community.

2 Methodology

Figure 1 presents the framework of the proposed model ParsingSum. The proposed linguistic-guided attention mechanism is generic and flexible to be applied in different Transformer structures. Inside the model, the encoder is a representations learner to learn distinctive feature representations from the source documents and decoder is able to decipher representations into language domain for summary generation. More concretely, the document sets are first fed into a Transformer-based encoder for representation learning. Meanwhile, the source documents are passed into an external dependency parser to fetch the dependency relations. These relations and the Transformer's multi-head attention then be input into the linguistic-guided attention mechanism to construct the linguistic attention map. With the assistance of linguistic information, the model can grasp intra-sentence linguistic relations for summaries generation.

2.1 Dependency Information Matrix

Dependency grammar is a family of grammar formalisms that plays an important role in natural language processing. The dependency parser constructs several dependency trees that represent grammatical structure and the relations between *head* words and corresponding *dependent* words. To utilize these dependency information, we first adopt an external dependency parser (Dozat et al., 2017), which can handle sentences of any length, to generate a set of dependency trees from multiple documents. The trees contain dependencies between any pair of dependent words in one sentence. Let P denotes the dependency information matrix for one sentence. $p_{ij} \in P$ is a dependency weight between token t_i and token t_j. We simplify the definition of the weight as shown in Eq.(1):

$$p_{ij} = \begin{cases} 1 & t_i \ominus t_j \\ 0 & t_i \oslash t_j \end{cases} \quad (1)$$

where $t_i \ominus t_j$ indicates that t_i and t_j have a dependency relation, while $t_i \oslash t_j$ represents there is no dependency between the two tokens. To simplify the model, we consider the relations are undirected by ignoring the direction of *head* word and *dependent* word. For any pair of tokens, as long as there is a dependency between them, the dependency information matrix is assigned a value of 1, otherwise it will be set to 0. We hope to keep all dependency relations between the pair words in a simple yet effective manner.

2.2 Linguistic-Guided Attention Mechanism

In order to process source documents effectively and preserve salient source relations in the summaries, in

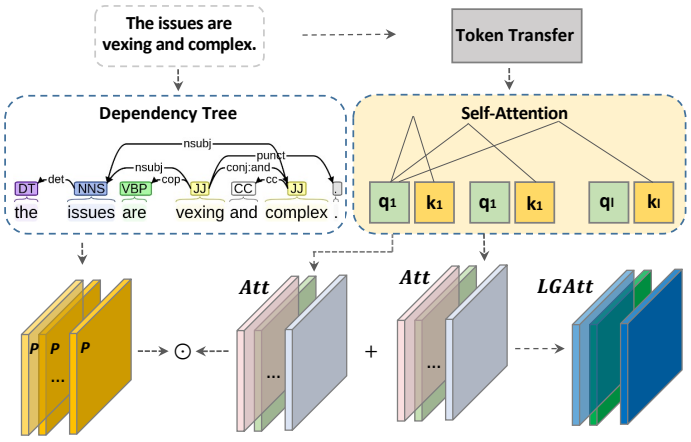


Figure 2: The linguistic-guided attention mechanism. The given exemplary sentence *The issues are vexing and complex.* is from Multi-News dataset (Alexander et al., 2019). Different properties of vocabularies and relations between words are included in the parsing information. The linguistic-guided attention mechanism incorporates the dependency information matrix P constructed from dependency trees of the input content and the Transformer's multi-head attention of this input content.

ParsingSum, we propose a novel linguistic-guided attention mechanism to extend the Transformer architecture (Ashish et al., 2017; Liu et al., 2019). Figure 2 depicts this mechanism on an exemplary sentence from Multi-News dataset (Alexander et al., 2019). linguistic-guided attention joins the dependency information matrix with the multi-head attention from source documents to generate syntactic-rich features. The linguistic-guided attention mechanism can be viewed as learning graph representations for the input sentences. Let $x_i^l \in \mathbb{R}^{d_{model}*1}$ denotes the output vector of the last encoding layer of Transformer for token t_i. For the attention head $head_z \in Head(j = 1, 2, ..., h)$, h represents the number of head. We have:

$$\begin{aligned} q_{i,head_z} &= W^{q,head_z} x_i^l \\ k_{i,head_z} &= W^{k,head_z} x_i^l \\ v_{i,head_z} &= W^{v,head_z} x_i^l \end{aligned} \tag{2}$$

where $W^{q,head_z}$, $W^{k,head_z}$, $W^{v,head_z} \in \mathbb{R}^{d_k * d_{model}}$ are weight matrices. d_k is the dimension of the key, query and value. $q_{i,head_z}$, $k_{i,head_z}$, $v_{i,head_z} \in \mathbb{R}^{d_k*1}$ are sub-query, sub-key and sub-values in different heads and we concatenate them respectively:

$$\begin{aligned} Q_i &= concat(q_{i,head_1}, q_{i,head_2}, ..., q_{i,head_h}) \\ K_i &= concat(k_{i,head_1}, q_{i,head_2}, ..., q_{i,head_h}) \\ V_i &= concat(v_{i,head_1}, q_{i,head_2}, ..., q_{i,head_h}) \end{aligned} \tag{3}$$

where $Q_i, K_i, V_i \in \mathbb{R}^{h*d_k*1}$ are corresponding key, query and value for attention calculation. In ParsingSum, the linguistic-guided attention merges dependency information with multi-head attention in the following manner:

$$LGAtt_{ij} = \alpha M_{ij} \odot Att_{ij} + Att_{ij} \tag{4}$$

where

$$Att_{ij} = softmax\left(\frac{Q_i{}^T K_j}{\sqrt{d_k}}\right) \tag{5}$$

$$M_{ij} = Stack\_h(p_{ij}) \tag{6}$$

where α is a trade-off hyper-parameter to balance the linguistic-guided information M_{ij} and multi-head attention Att_{ij}. In order to fuse dependency weight p_{ij}, we build a function $stack\_h(\cdot)$ to repeat p_{ij} on the dimension of head to have the same size with $Att_{ij} \in \mathbb{R}^{h*1*1}$. $\odot$ denotes the element-wise Hadamard product. Then, we have:

$$Context_i = \sum_j LGAtt_{ij} \cdot V_j \tag{7}$$

where $Context_i$ represents the context vectors generated by linguistic-guide attention. Later on, two layer-normalization operations are applied to $Context_i$ to get the output vector of current encoder layer for token t_i:

$$x_i^{l+1} = LayerNorm\left(k_i + FFN\left(k_i\right)\right) \tag{8}$$

$$k_i = LayerNorm\left(x_i^l + Context_i\right) \tag{9}$$

where FFN is a two-layer feed-forward network with ReLU as activation function. Then, the learned feature representations are passed into multiple decoder layers that are fairly similar to the Flat Transformer structure (Sebastian et al., 2018).

3 Experiments

In this section, we report the effectiveness of the proposed linguistic-guided attention. Extensive analyses have been done on how to select suitable fusion weights in linguistic-guided attention, as well as the influence of batch size for model training. Later on, discussions on different fusion methods and their visualization are conducted.

3.1 Models for Comparison

We compare ParsingSum with the following models: *LexRank* computes the importance of a sentence-based on the concept of eigenvector centrality in a sentence graph (Gunes et al., 2004). *TextRank* is a graph-based ranking model (Rada et al., 2004). *Maximal Marginal Relevance (MMR)* (Jaime et al., 1998) considers the importance and redundancy of a sentence in a complementary way to decide whether to select the sentence for the summary. *SummPip* (Zhao et al., 2020) considers both linguistic knowledge and deep neural representations for summary generation. *BRNN*[1] is an bidirectional RNN-based model. *Flat Transformer (FT)* (Alexander et al., 2019) is a Transformer-based model on a flat token sequence. *Hi-MAP* (Alexander et al., 2019) incorporates MMR into a pointer-generator network. *Hierarchical Transformer (HT)* (Liu et al., 2019) is an abstractive summarizer that can capture cross-document relationships via hierarchical Transformer encoder and flat Transformer decoder.

3.2 Experimental Settings

We equip the proposed linguistic-guided attention on both Hierarchical Transformer (HT) and Flat Transformer (FT) architectures. Two models are thus derived: ParsingSum-HT and ParsingSum-FT. For ParsingSum-HT, we follow the implementation of the HT model by using six local Transformer layers and two global Transformer layers with eight heads[2]. For ParsingSum-FT, we follow FT model settings and adopt four encoder layers and four decoder layers[3]. For training, we use *Adam* optimizer (β1=0.9 and β2=0.998). The dropout rates of both encoder and decoder are set to 0.1. The initial learning rate is set to 1×10^{-3}. The first 8000 steps are trained for warming up and the models are trained with a multi-step learning rate reduction strategy. We evaluate the proposed model and compare its performances with multiple baseline models using ROUGE scores (Lin et al., 2004), the most commonly used evaluation metrics, and human evaluation. The experiments are conducted on two datasets : Multi-News dataset (Alexander et al., 2019) and WCEP-100 dataset (Demian et al., 2020). Multi-News a large-scale English MDS benchmark dataset

Table 2: Models comparison on Multi-News test set. We rerun all the compared models under the same environment. The best results for each column are in bold.

Models	ROUGE-F		
	1	2	L
LexRank	37.92	13.10	16.86
TextRank	39.02	14.54	18.33
MMR	42.12	13.19	18.41
SummPip	42.29	13.29	18.54
BRNN	38.36	13.55	19.33
FT	42.98	14.48	20.06
Hi-MAP	42.98	14.85	20.36
HT	36.09	12.64	20.10
ParsingSum-HT (Ours)	37.34	13.00	20.42
ParsingSum-FT (Ours)	**44.32**	**15.35**	**20.72**

Table 3: Models comparison on WCEP-100 test set. The best results for each column are in bold.

Models	ROUGE-F		
	1	2	L
HT	23.20	5.78	17.45
FT	23.41	6.64	17.93
ParsingSum-HT (Ours)	24.03	6.42	18.31
ParsingSum-FT (Ours)	**26.45**	**7.06**	**18.98**

[1]We implement the BRNN model based on https://github.com/Alex-Fabbri/Multi-News/tree/master/code/OpenNMT-py-baselines

[2]We train the HT model on one GPU for 100,000 steps with batch-size 13,000.

[3]We implement the FT model based on https://github.com/Alex-Fabbri/Multi-News/tree/master/code/OpenNMT-py-baselines. We train the FT model for 20,000 steps with batch-size 4096 on one GPU.

Table 4: The analysis of fusion weights of linguistic-guided attention on Multi-News validation set. The best results for each column are in bold.

Models	ROUGE-F		
	1	2	L
HT	36.02	12.57	20.05
ParsingSum-HT (α=1)	36.71	12.79	20.27
ParsingSum-HT (α=2)	35.64	12.18	19.80
ParsingSum-HT (α=3)	**36.74**	**12.86**	**20.29**
FT	42.81	14.25	19.81
ParsingSum-FT (α=1)	43.69	14.67	19.95
ParsingSum-FT (α=2)	**43.84**	**15.01**	**20.50**
ParsingSum-FT (α=3)	43.61	14.92	20.13

extracted from news articles. It includes 56,216 article-summary pairs and it is further scattered with the ratio 8:1:1 for training, validation and testing respectively. Each document set contains 2 to 10 articles with a total length of 2103.49 words. The average length of golden summaries is 263.66 words. WCEP-100 consists of 10,200 document sets (8158 for training, 1020 for validation and 1022 for testing) with one corresponding human-written summary. The average length of the summaries are 32 words. Deep Biaffine dependency parsing (Dozat et al., 2017) are used to generate dependency information for these source documents.

3.3 Overall Performance

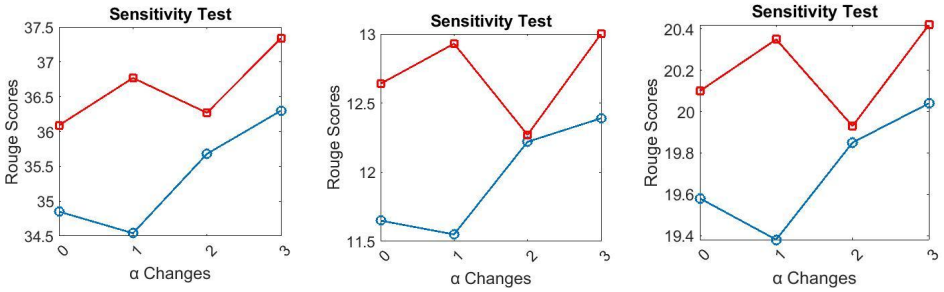
We evaluate the proposed ParsingSum-HT, ParsingSum-FT and compare them with multiple mainstream models on both Multi-News and WCEP-100 datasets. For fair comparisons, we rerun all the compared models under the same environment. For Multi-News dataset, as shown in Table 2, the ParsingSum-HT model receives higher ROUGE scores (across all ROUGE-1, ROUGE-2 and ROUGE-L) steadily compared to the original HT model. The linguistic-guided attention helps the model raise 1.25 on ROUGE-1 score, 0.36 on ROUGE-2 score, and 0.32 on ROUGE-L respectively. It indicates the outstanding capability of ParsingSum models to retain the intention of original documents when generating summaries. A similar phenomenon shows on the ParsingSum-FT model. More specifically, ParsingSum-FT surpasses FT model 1.34 on ROUGE-1 score, 0.87 on ROUGE-2 score, and 0.66 on ROUGE-L score, which shows the effectiveness of linguistic-guided attention on the Transformer-based models. It is worth noting that the proposed ParsingSum-FT is able to outperform its baseline (i.e., FT model) by a large margin and also receives the highest ROUGE scores across all the compared methods. The effect of linguistic-guided attention can be verified on the WCEP-100 dataset. The ROUGE results can be improved on both two version of Transformer based summarization models. These results indicate the outstanding capability of linguistic-guided attention to retain the intention of original documents when generating summaries.

Table 5: Human evaluation results on the Multi-News dataset. The best results for each column are in bold.

Models	Fluency	Informativeness	Consistency
Hi-MAP	2.53	2.80	2.33
FT	2.47	2.67	2.60
HT	2.20	2.13	2.40
ParsingSum-HT	2.73	**2.93**	**2.87**
ParsingSum-FT	**2.87**	2.87	2.73

3.4 Human evaluation

Although ROUGE are the standard evaluation metrics for summarization tasks, they focus on lexical matching instead of semantic matching. Therefore, in addition to the automatic evaluation, we access model performance by human evaluation in a semantic way. We invite three annotators who research natural language processing to evaluate the performance of five models (Hi-MAP, FT, HT, ParsingSum-FT, ParsingSum-HT) independently. For each model, 30 summaries are randomly selected from the Multi-News dataset. Three criteria are taken into account to evaluate the quality of generated summaries: (1) Informativeness: how much important information does the generated summary contain from the input document? (2) Fluency: how coherent are the generated summaries? (3) Consistency: how closely the information in the generated summaries are consistent with the input documents? Annotators are asked to give scores from 1 (worst) to 5 (best). Table 5 summarizes the comparison results of five summarization models. For each model, the score of each criterion is computed by averaging the score of all summary samples. The re-



(ROUGE-1) (ROUGE-2) (ROUGE-L)

Figure 3: The performance of ParsingSum-HT on small (in blue) and large batch-size setting (in red).

Table 6: Performance of ParsingSum-HT via different fusion methods on Multi-New validation set. The best results for each column are in bold.

Models	ROUGE-F		
	1	2	L
ParsingSum-HT (P0.25)	19.50	3.40	12.59
ParsingSum-HT (G0.25)	16.84	1.92	11.36
ParsingSum-HT (G8)	20.18	3.55	13.00
ParsingSum-HT (α=3)	**36.74**	**12.86**	**20.29**

sults demonstrate that the Transformer based models equipped with linguistic-guided attention are able able to generate higher quality summaries than the baseline models in terms of informativeness, fluency, and consistency. These human evaluation results further validate the effectiveness of our proposed linguistic-guided attention mechanism.

3.5 Analysis

We further analyze the effects of the trade-off parameter α and batch-size in ParsingSum. We also examine and discuss different manners to incorporate parsing information into the proposed model.

The Analysis of the Fusion Weights. The trade-off factor α controls the intensity of attention from a linguistic perspective to be fused with multi-head attention. To analyze its importance, we conduct experiments by setting α to 0, 1, 2, and 3 ($\alpha = 0$ denotes the naive Transformer model without linguistic-guided attention) on the two proposed models on the validation set. The results are shown in Table 4. Generally, there is an increasing trend with the increment of α. This rising trend further proves assigning a relatively larger α in a suitable range can improve the performance of summarization models.

The Analysis of Batch-size. Batch-size is considered to have a great effect on the mini-batch stochastic gradient descent process of model training (Smith et al., 2018) and it will thus further affect the model performance. To validate it empirically, we train the model with small/large batch-size (the small batch-size is 4,500 and the large one is 13,000) of the ParsingSum-HT model. The experiments are conducted with different α. The results in Figure 3 show that smaller batch-size reduces the performance on all the evaluation metrics. Interestingly, the ROUGE scores of the small batch-size setting are steadily increasing with α changes from 1 to 3; when the model is trained with large batch-size, the increasing trend is retained but the ROUGE scores are jittering when α equals two. It indicates different batch-sizes have different sensitivities towards the change of α.

The Analysis of the Fusion Methods. How to integrate the parsing information into the Transformer-based model is important in our work. In addition to the fusion method introduced in Section 2.2, we attempt several other fusion methods under a small batch-size setting of the ParsingSum-HT model: (1) Direct fusion. Weight the dependency parsing matrix and add it directly to the multi-head attention. It denotes as ParsingSum-HT (P0.25):

$$LGAtt_{ij} = 0.25M_{ij} + Att_{ij} \tag{10}$$

(2) Gaussian-based fusion. We adopt the idea from (Li et al., 2020) and apply Gaussian weights to the product of the dependency information and the multi-head attention. The Gaussian weights are set to 0.25 (ParsingSum-HT (G0.25)) and 8

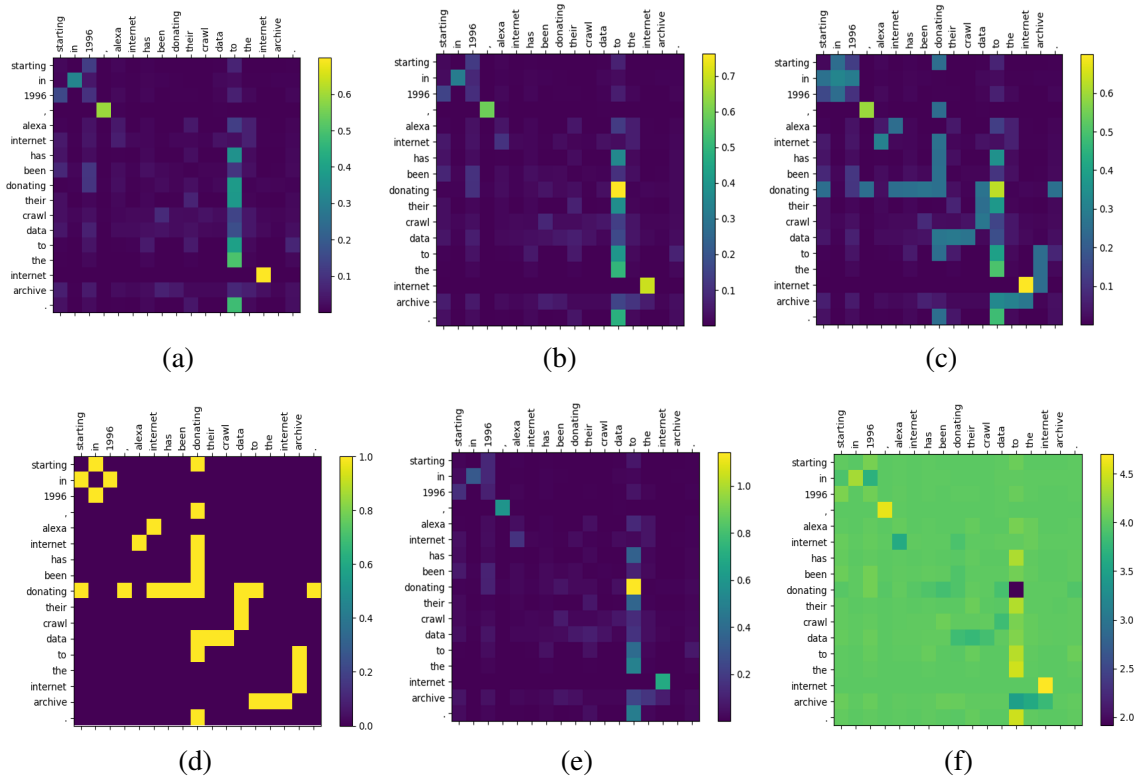


Figure 4: Visualization of different fusion methods. (a) HT model; (b) ParsingSum-HT (α=1); (c) ParsingSum-HT (P0.25); (d) dependency parsing matrix; (e) ParsingSum-HT (α=3); (f) ParsingSum-HT (G0.25).

(ParsingSum-HT (G8)):

$$LGAtt_{ij} = \frac{(1 - M_{ij}Att_{ij})^2}{0.25} + Att_{ij} \quad (11)$$

$$LGAtt_{ij} = \frac{(1 - M_{ij}Att_{ij})^2}{8} + Att_{ij} \quad (12)$$

Figure 4(a) and 4(d) represent the heatmap of the HT model and dependency parsing matrix. Figure 4(b), 4(c), 4(e), and 4(f) illustrate the attention maps of different fusion methods. Table 6 presents the performance of the mentioned fusion methods on Multi-New validation set. ParsingSum-HT with α=3 receives the best results for all ROUGE scores. The potential reason is that through direct fusion and Gaussian fusion, the scale of the original multi-head attention has been overwhelmed, leading to posing the dependency information in a dominant position. In this case, the normal gradient backpropagation process has been disturbed. The experiment results indicate that a direct summation of the weighted dependency parsing matrix and multi-head attention may damage the original attention. On the other hand, a "soft" fusion (when α is adopted) of these two attentions can achieve promising results.

4 Conclusion

This paper presents a generic framework to leverage linguistic knowledge to improve the performance of abstractive Transformer-based summarization models. The proposed linguistic guided attention mechanism can be seamlessly incorporated into multiple mainstream Transformer-based summarization models and can be outperform existing Transformer-based methods by a large margin. We develop two models based on Flat Transformer (FT) and Hierarchical Transformer (HT). The proposed ParsingSum-HT and ParsingSum-FT incorporate dependency relations with Transformer's multi-head attention for summaries generation. The experiments confirm that utilizing dependency information from the source documents is beneficial to guide the summaries generation process.

References

Perez-Beltrachini, L. & Lapata, M. Multi-Document Summarization with Determinantal Point Process Attention. *Journal Of Artificial Intelligence Research*. **71**, pp. 371-399 (2021)

Dhakras, P. & Shrivastava, M. BoWLer. A Neural Approach to Extractive Text Summarization. *Proceedings Of The 32nd Pacific Asia Conference On Language, Information And Computation* (2018)

Xiao, W., Beltagy, I., Carenini, G. & Cohan, A. PRIMERA: Pyramid-based Masked Sentence Pretraining for Multi-document Summarization. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics* (2022)

Beltagy, I., Peters, M. & Cohan, A. Longformer: The long-document Transformer. *ArXiv Preprint ArXiv:2004.05150* (2020)

Deguchi, H., Tamura, A. & Ninomiya, T. Dependency-Based Self-Attention for Transformer NMT. *Proceedings Of The International Conference On Recent Advances In Natural Language Processing*. pp. 239-246 (2019)

Ghalandari, D., Hokamp, C., Pham, N., Glover, J. & Ifrim, G. A Large-Scale Multi-Document Summarization Dataset from the Wikipedia Current Events Portal. *Proceedings Of The 58th Annual Meeting Of The Association For Computational Linguistics*. pp. 1302-1308 (2020)

Smith, S., Kindermans, P., Ying, C. & Le, Q. Don't Decay the Learning Rate, Increase the Batch Size. *Proceedings Of The 6th International Conference On Learning Representations* (2018)

Zhao, J., Liu, M., Gao, L., Jin, Y., Du, L., Zhao, H., Zhang, H. & Haffari, G. SummPip: Unsupervised Multi-Document Summarization with Sentence Graph Compression. *Proceedings Of The 43rd International ACM SIGIR Conference On Research And Development In Information Retrieval*. pp. 1949-1952 (2020)

Halliday, M., Matthiessen, C., Halliday, M. & Matthiessen, C. An Introduction to Functional Grammar. *Routledge* (2014)

Gehrmann, S., Deng, Y. & Rush, A. Bottom-Up Abstractive Summarization. *Proceedings Of The 2018 Conference On Empirical Methods In Natural Language Processing*. pp. 4098-4109 (2018)

Wang, K., Shen, W., Yang, Y., Quan, X. & Wang, R. Relational Graph Attention Network for Aspect-based Sentiment Analysis. *Proceedings Of The 58th Annual Meeting Of The Association For Computational Linguistics*. pp. 3229-3238 (2020)

Liu, P., Saleh, M., Pot, E., Goodrich, B., Sepassi, R., Kaiser, L. & Shazeer, N. Generating Wikipedia by Summarizing Long Sequences. *Proceedings Of 6th International Conference On Learning Representations* (2018)

Cao, Q., Liang, X., Li, B. & Lin, L. Interpretable Visual Question Answering by Reasoning on Dependency Trees. *IEEE Transactions On Pattern Analysis And Machine Intelligence*. **43**, 887-901 (2021)

Wu, S., Zhang, D., Yang, N., Li, M. & Zhou, M. Sequence-to-Dependency Neural Machine Translation. *Proceedings Of The 55th Annual Meeting Of The Association For Computational Linguistics*. pp. 698-707 (2017)

Sun, K., Zhang, R., Mensah, S., Mao, Y. & Liu, X. Aspect-Level Sentiment Analysis Via Convolution over Dependency Tree. *Proceedings Of The 2019 Conference On Empirical Methods In Natural Language Processing And The 9th International Joint Conference On Natural Language Processing*. pp. 5678-5687 (2019)

Carbonell, J. & Goldstein, J. The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries. *Proceedings Of The 21st Annual International ACM SIGIR Conference On Research And Development In Information Retrieval*. pp. 335-336 (1998)

Jin, H., Wang, T. & Wan, X. Multi-Granularity Interaction Network for Extractive and Abstractive Multi-Document Summarization. *Proceedings Of The 58th Annual Meeting Of The Association For Computational Linguistics*. pp. 6244-6254 (2020)

Fabbri, A., Li, I., She, T., Li, S. & Radev, D. Multi-News: A Large-Scale Multi-Document Summarization Dataset and Abstractive Hierarchical Model. *Proceedings Of The 57th Conference Of The Association For Computational Linguistics*. pp. 1074-1084 (2019)

Li, W., Xiao, X., Liu, J., Wu, H., Wang, H. & Du, J. Leveraging Graph to Improve Abstractive Multi-Document Summarization. *Proceedings Of The 58th Annual Meeting Of The Association For Computational Linguistics*. pp. 6232-6243 (2020)

Fernandes, P., Allamanis, M. & Brockschmidt, M. Structured Neural Summarization. *Proceedings Of The 7th International Conference On Learning Representations* (2019)

Takase, S., Suzuki, J., Okazaki, N., Hirao, T. & Nagata, M. Neural Headline Generation on Abstract Meaning Representation. *Proceedings Of The 2016 Conference On Empirical Methods In Natural Language Processing*. pp. 1054-1059 (2016)

Song, K., Lebanoff, L., Guo, Q., Qiu, X., Xue, X., Li, C., Yu, D. & Liu, F. Joint Parsing and Generation for Abstractive Summarization. *Proceedings Of The Thirty-Fourth AAAI Conference On Artificial Intelligence*. pp. 8894-8901 (2020)

Lin, C. ROUGE: A Package for Automatic Evaluation of Summaries. *Proceedings Of The Workshop Of Text Summarization Branches Out*. pp. 74-81 (2004)

Leite, D., Rino, L., Pardo, T. & Nunes, M. Extractive Automatic Summarization: Does more Linguistic Knowledge Make a Difference? *Proceedings Of The 2rd Workshop On TextGraphs: Graph-based Algorithms For Natural Language Processing*. pp. 17-24 (2007)

Liu, Y. & Lapata, M. Hierarchical Transformers for Multi-Document Summarization. *Proceedings Of The 57th Conference Of The Association For Computational Linguistics*. pp. 5070-5081 (2019)

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, L. & Polosukhin, I. Attention is All you Need. *Proceedings Of The Annual Conference On Neural Information Processing Systems*. pp. 5998-6008 (2017)

Mihalcea, R. & Tarau, P. TextRank: Bringing Order into Text. *Proceedings Of The 2004 Conference On Empirical Methods In Natural Language Processing*. pp. 404-411 (2004)

Erkan, G. & Radev, D. LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. *Journal Of Artificial Intelligence Research*. **22** pp. 457-479 (2004)

Ma, C., Zhang, W., Guo, M., Wang, H. & Sheng, Q. Multi-document Summarization via Deep Learning Techniques: A Survey. *ACM Computing Surveys* (2022)

Dozat, T. & Manning, C. Deep Biaffine Attention for Neural Dependency Parsing. *Proceedings Of The 5th International Conference On Learning Representations* (2017)

Jin, H., Wang, T. & Wan, X. SemSUM: Semantic Dependency Guided Neural Abstractive Summarization. *Proceedings Of The Thirty-Fourth AAAI Conference On Artificial Intelligence*. pp. 8026-8033 (2020)