Computer Speech & Language 66 (2021) 101159

Contents lists available at ScienceDirect

Computer Speech & Language

journal homepage: www.elsevier.com/locate/csl

Gated dynamic convolutions with deep layer fusion for abstractive document summarization

Hongseok Kwon[a,*], Byung-Hyun Go[a], Juhong Park[a], Wonkee Lee[a], Yewon Jeong[a], Jong-Hyeok Lee[b]

[a] Department of Computer Science and Engineering, Pohang University of Science and Technology, 77 Cheongam-ro, Nam-gu, Pohang 37673 Republic of Korea
[b] Pohang University of Science and Technology Graduate School of Artificial Intelligence

ARTICLE INFO

Article History:
Received 25 November 2019
Revised 1 August 2020
Accepted 13 September 2020
Available online 25 September 2020

Keywords:
Document summarization
Gated dynamic convolutions
Deep layer fusion
Convolutional encoder-decoder
Text generation

ABSTRACT

We present a novel abstractive document summarization based on the recently proposed dynamic convolutional encoder-decoder architectures. We address several aspects of summarization that are not well modeled by the basic architecture, by integrating multiple layers of the encoder, controlling information flow in the hierarchy, and exploiting external knowledge. First, we propose a simple and efficient deep layer fusion to extract salient information from the encoder layers. Second, we propose a gating mechanism to control and maintain important contextual information through the encoder-decoder layers into dynamic convolutions. Lastly, we put part-of-speech information into the model as external knowledge to better predict filters for dynamic convolutions. We evaluate our model using ROUGE metrics on three different datasets: CNN-DM, NEWSROOM-ABS, and XSUM. Experimental results show that the proposed model outperforms the state-of-the-art abstractive models on NEWSROOM-ABS and XSUM and shows comparable scores on CNN-DM.

1. Introduction

The text summarization task aims to automatically generate a condensed text that captures the most salient information from a given source text. Text summarization systems can be divided into several types according to their usage and approach to generating summaries. First, according to usage, summarization systems may be classified into two categories: informative and indicative. Informative summarization attempts to include all salient information in the document into a summary. Therefore, an informative summary could, in theory, substitute the entire document. On the other hand, indicative summarization conveys the main topic of the document briefly without disclosing its details. Indicative summaries enable users to decide whether to read a document or not in a matter of a few seconds. Some examples of indicative summaries are news headlines and library card catalog entries. Second, there are largely two approaches to text summarization: extractive and abstractive. The extractive approach writes a summary by extracting important linguistic constituents from the document and assembling them to make grammatical sentences. In contrast, the abstractive approach constructs summaries using words that may or may not exist in the document using sophisticated techniques such as meaning representation, content organization, and surface realization Yao et al. (2017). Abstractive systems generally can generate shorter summaries than extractive systems due to being able to paraphrase, resulting

*Corresponding author.
E-mail address: hkwon@postech.ac.kr (H. Kwon).

https://doi.org/10.1016/j.csl.2020.101159
0885-2308/

in summaries containing less unnecessary or redundant information and with more coherent relations between sentences. For the scope of this study, we aim to model informative summarization using the abstractive approach.

The last few years of research on abstractive summarization have seen significant progress, of which a considerable amount of studies have been performed on sequence-to-sequence (seq2seq) models with the attention mechanism. Most of the recent work is based on recurrent neural networks (RNNs) (See et al., 2017; Celikyilmaz et al., 2018), convolutional neural networks (CNNs) (Fan et al., 2018; Narayan et al., 2018; Kim et al., 2019) and self-attention networks (Liu et al., 2018). The dominant approaches are based on the LSTM-RNN encoder-decoder architecture that encodes an input sequence into a fixed-size summary representation, which is then used to generate a summary token by token. Other approaches use CNNs and self-attention networks: CNN-based models summarize an input sequence with a predefined context window through stacked layers, while self-attention-based models summarize all context. They then proceed to generate a summary in a similar fashion to LSTM-RNN models but use CNNs or self-attention networks in the decoder, respectively. Most of the above approaches are enhanced using the pointer mechanism (Vinyals et al., 2015), which allows the decoder to copy words from the document. While the pointer mechanism alleviates the out-of-vocabulary (OOV) problem, it potentially leads to excessive copying of long sequences from the document directly into the summary, consequently diminishing the model's abstraction capabilities (Kryściński et al., 2018). For this reason, the pointer-generator networks perform poorly on highly abstractive datasets (Kim et al., 2019).

Recently, (Wu et al., 2018) proposed a convolutional seq2seq model utilizing dynamic convolutions, which, unlike standard convolutions, use a different kernel at every position. The kernel is predicted by a simple linear function of the current word only. Dynamic convolutions have shown promising results for machine translation, language modeling, and abstractive summarization while being simpler and more efficient than self-attention networks. The proposed model outperformed the previous state-of-the-art self-attention models in the WMT'14 en-de translation task.

In this paper, we adopt recently proposed dynamic convolutions to text summarization task and propose a novel solution tailored to English abstractive summarization with informative purposes. Despite the similarities between summarization and MT in terms of natural language generation task and use similar sequence-to-sequence architectures, abstractive summarization is a very different task from MT (Nallapati et al., 2016). Unlike in MT, the source text and target (summary) have radically different lengths, with the target length typically being very short and not depending on the length of the source. Moreover, one of the goals of text summarization is to generate condensed text in a lossy manner such that only the key information in the source document is preserved, whereas in MT, the translation is expected to contain the entire meaning of the source sentence in the target sentence.

The typical key factors of abstractive document summarization are informativeness, coherence, redundancy, and novelty of the generated summary. Among these traits, we focus on how to improve neural summarization models' capacity to capture informative contents from long input sequences and generate informative summaries. Generating informative summaries requires the ability to extract salient linguistic constituents based on an understanding of the whole article, and to separate the key constituents from the details so that they may be compiled into a concise summary. For these reasons, we present gated dynamic convolutions augmented with POS information and deep layer fusion. First, we utilize POS tags to capture morphological and syntactic features from a given sentence. These lexical categories are useful in recognizing salient content and predicting a better kernel in dynamic convolutions. Second, we exploit multiple layers of the encoder to ease the burden of abstracting large n-gram features to the top of the encoder layer caused by the long length of the input in document summarization. Moreover, we fuse the multiple layers to produce a multi-perspective representation of the document that spans from local key n-gram features to full document understanding. Lastly, inspired by the success of gating mechanisms (Bahdanau et al., 2014; Gehring et al., 2017) used in many NLP tasks, we propose a gating mechanism with memory cells and apply it to dynamic convolutions.

We conduct experiments on a variety of datasets: XSUM, NEWSROOM-ABS (NR-ABS) and CNN-DM with different characteristics to verify the effectiveness of our model. Empirical results on ROUGE metrics demonstrate that our proposed model outperforms existing state-of-the-art abstractive systems on the XSUM and NR-ABS datasets and produce results comparable to those of recent abstractive models on CNN-DM.

Our contribution can be summarized as follows:

- We propose a novel abstractive document summarization system based on newly proposed dynamic convolutions.
- We present a simple and efficient deep layer fusion method to capture different types of syntactic and semantic features scattered in the encoder layers.
- We propose a gating mechanism with memory cells to control and maintain salient information in the hierarchy.
- We explicitly supply POS information to the model to assess to what the extent the addition of external knowledge can improve abstractive summarization capabilities.

2. Related work

Our work is related to the following three topics.

2.1. POS information

POS information has been employed in several NLP tasks based on neural networks such as machine translation (Sennrich and Haddow, 2016), language models (Niehues et al., 2016) and text summarization (Nallapati et al., 2016). Incorporating external

knowledge into neural models have been shown to produce better results and is now a common technique in NLP. Using POS tags has known to be useful for Word Sense Disambiguation (WSD) and modeling for word agreement. Nallapati et al. (2016)'s utilization of POS tags to the purpose of WSD is the only other instance of syntactic information being integrated to neural abstractive summarization, to the best of our knowledge. They showed the impact of employing linguistic features by incorporating POS tags, named-entity tags, term frequency, and inverse document frequency statistics of words at once. In this work, we utilize POS tags as an external feature to exploit useful grammatical and morphological information to enhance dynamic convolutions. To our knowledge, our work is the first attempt to investigate the usefulness of using POS tags as auxiliary inputs in neural abstractive summarization.

2.2. Layer aggregation

Neural seq2seq models based on self-attention and CNNs contain two parts: an encoder and decoder each consisting of a stack of identical layers. A text summarization model should be capable of constructing a rich representation that not only serves as an understanding of the document, but also allows the identification of salient information. However, most existing methods only use the top layer of the encoder to generate output sequences, discarding any useful features located in the lower layers of the encoder.

To address these issues, some researchers (Yu et al., 2018; Dou et al., 2018; 2019) have proposed layer aggregation methods to better utilize the different levels of abstraction in the encoder for Neural Machine Translation (NMT), and their models significantly outperform their respective baselines.

However, all of these works require a huge amount of additional parameters to fuse the encoder layers, leading to slower training and decoding. To solve these problems, we propose a simple and effective layer fusion method that achieves better results with a much smaller increase in the number of parameters.

2.3. Gating mechanism

Gating mechanisms control information flow in networks and have been shown to be effective in many applications. LSTMs have a long-term memory that is handled by input and forget gates across time steps to capture long-term dependencies and alleviate the vanishing gradient problem. Conversely, CNNs are relatively robust to the vanishing gradient problem because they capture long-term dependencies through relatively short stacked layers, instead of long time steps. Therefore, CNNs do not use the more sophisticated gating mechanisms used in LSTMs, and instead only rely on output gates such as ReLU (Nair and Hinton, 2010), Gated Tanh Units (GTU) (Van den Oord et al., 2016) and Gated Linear Units (GLU) (Gehring et al., 2017).

However, preserving important features throughout layers is also crucial to improving performance. Chen et al. (2018) proposed a gated CNN (GCNN) equipped with the following three gates and memory cells for sentence matching. The output gate controls what information should be propagated to the next layer. The forget and update gates are designed similarly to Gated Recurrent Units (GRU) (Cho et al., 2014): to determine which information from the previous memory cells should be discarded and to determine how much past information should be stored in the current memory cells respectively. In our work, we adopt a modified version of these gates which contain explicit update gates to extract salient information from previous layers to dynamic convolutions.

3. Model

Previous research on abstractive summarization has seen significant improvement by using seq2seq models with the attention mechanism. Most of the previous approaches are based on LSTM-RNN encoder-decoder architectures while some studies use hierarchical structures such as 2-level LSTM-RNNs, CNNs, or self-attention networks to handle long sequences of words. Some researchers argue that exploiting more fine-grained levels of representation from the encoder (i.e. word, phrase, etc.) is useful to generate a good summary (Celikyilmaz et al., 2018; Kim et al., 2019) due to the following reason: summarizing a document requires not only an understanding of the document but also the ability to extract salient fragments, and using these various levels of representation yields a better summary. In this sense, CNNs are well-suited for utilizing various ranges of inputs from n-gram window contexts compared to the other networks.

Motivated by this observation, our model builds on a recently proposed convolutional seq2seq (ConvS2S) model utilizing dynamic convolutions, which, unlike standard convolutions, use a different convolution kernel at every position. The proposed model shows promising results for sequence modeling. We also augment dynamic convolutions with a gating mechanism, multiple layer aggregation, and POS information.

3.1. Model architecture

The overall architecture of our model is shown in Fig. 1. The encoder and decoder networks are almost identical to the model proposed by the original authors (DynamicConv) (Wu et al., 2018), which in turn are very similar to Transformer (Vaswani et al., 2017); we refer the reader to (Wu et al., 2018) for more details regarding the architecture. DynamicConv consists of N and M stacked layers not unlike those in Transformer; however, the self-attention modules are replaced with dynamic convolution modules in every layer. Our model is based on DynamicConv with some changes: we added a gating mechanism with memory

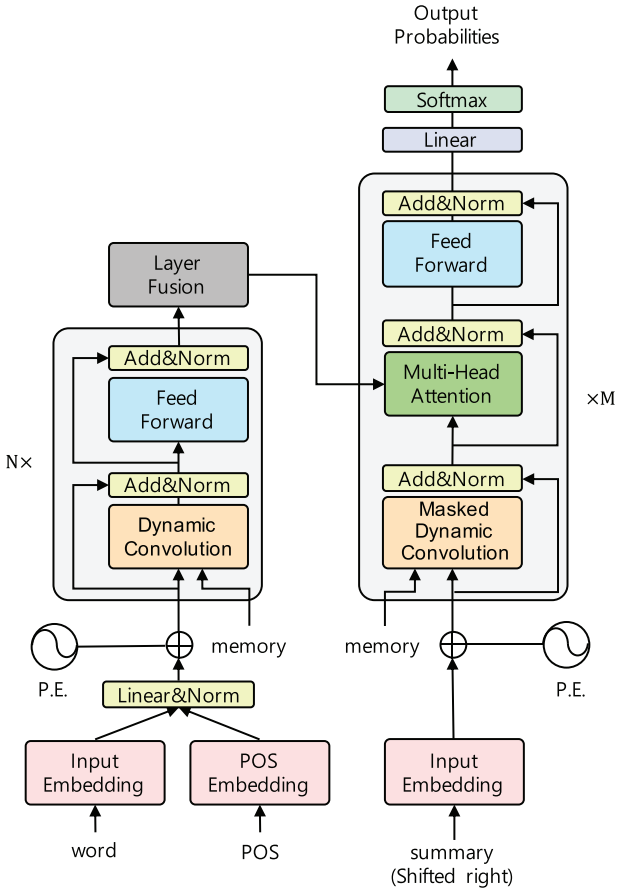

Fig. 1. Illustration of the proposed gated dynamic convolution with deep layer fusion.

cells into the encoder and decoder layers; a deep layer fusion module was stacked at the end of the encoder; and POS information was given as an additional input. More specifically, each encoder layer has two sub-layers. The first is a dynamic convolution module, and the second is a feed-forward (FFN) module. Sub-layers are surrounded by residual connections, followed by layer normalization (LayerNorm). The encoder is followed by the deep layer fusion module. Each decoder layer is identical to the encoder layers except that they have an additional layer called a multi-head attention module between the dynamic convolution module and the feed-forward module. The multi-head attention module is identical to that in Transformer except that it uses the deep layer fusion output for the key and value inputs.

The encoder input is the document word vectors augmented with POS information while the decoder input is the summary word vectors but without POS augmentation. Sinusoidal positional embeddings are also added to the encoder-decoder inputs to inject information about their absolute positions in the sequence. The encoder takes the input to generate contextual representations throughout the hierarchy with the gated dynamic convolutions. The deep layer fusion module fuses multiple layers of the encoder to produce multi-perspective representations for the document. The gating mechanism is controlled by three gates with memory cells and conveys crucial information about each intermediate representation from the first layer to the end of the stacked layer.

Similarly, the decoder takes the target sequence to obtain contextual representations over the words generated so far. In the decoder, multi-head source-target attention is applied to obtain salient source contents by attending to the multi-perspective representations. Finally, a softmax operation is performed on the output of the decoder to produce an output word distribution.

3.2. POS augmentation

The incorporation of external knowledge into neural models is a well-practiced technique in NLP, known to be useful in most tasks in the field. POS information is a type of such external knowledge that has been traditionally employed as a valuable resource for NLP tasks. On top of this, we further hypothesize that POS information is particularly beneficial in predicting better kernels in the dynamic convolution module. The word embeddings used as input to the dynamic convolution layers do not explicitly contain information about their context. This information is typically instead assumed to be encoded through applying multiple convolutions. We reason that explicit POS tags would allow the kernel function to recognize categories of words with

similar grammatical properties that would be informative in determining the desired focus in the given context window. Moreover, contents words, especially proper nouns, could potentially be a topic word or a clue to the position of salient information in the document.

Let $\mathbf{X}$ denote the input word embeddings in the document $(\mathbf{x}_i, \ldots, \mathbf{x}_n)$ where $\mathbf{x}_i \in \mathbb{R}^d$. We use positional encodings (Vaswani et al., 2017) to inject absolute word positions in the document $\mathbf{P} = (\mathbf{p}_1, \ldots, \mathbf{p}_n)$ where $\mathbf{p}_i \in \mathbb{R}^d$. We integrate POS information into the model via concatenation-based conditioning. First, POS tags are mapped onto POS embeddings $\mathbf{T} = (\mathbf{t}_1, \ldots, \mathbf{t}_n)$ where $\mathbf{t}_i \in \mathbb{R}^d$. Then, POS and word embeddings are concatenated into a single vector and fed to a fully connected layer to produce the input vector of the first layer. Accordingly, the input for the encoder is represented by $\mathbf{E} = (\mathbf{e}_1, \ldots, \mathbf{e}_n)$, where $\mathbf{e}_i$ is

$$\mathbf{e}_i = \text{LayerNorm}(\mathbf{W}_p[\mathbf{x}_i; \mathbf{t}_i]) + \mathbf{p}_i \in \mathbb{R}^d \tag{1}$$

On the decoder side, we do not use POS features because obtaining POS features from hypotheses at generation time is another challenging task and requires a large amount of computation. In our experiments, we use BPE (Byte-Pair Encoding) subword encodings as the encoder-decoder inputs. However, BPE segmentation, in which rare words are segmented into a sequence of subword units, blurs the word boundaries for some words. But such word boundary information may be useful to recognize individual words through the stacked layers. Therefore, we adopt the BIOE format proposed by Sennrich and Haddow (2016), where B stands for beginning, I for intermediate, E for ending, and O for full word. However, instead of supplying BIOE information separately, we directly attach the BIOE tags to POS tags (e.g. B-NNP).

3.3. Deep layer fusion

Unlike in MT, the average input length of document summarization is usually longer than 400 words. Given a multi-layer CNN based architecture such as dynamic convolution, each hidden state of the top and final layer of the encoder must be able to represent a long range of words. The intermediate layers, on the other hand, would ideally encode features from smaller n-grams. However, in conventional CNN architectures, these features are not directly accessible from the rest of the model. Therefore, to ease the burden of the top layer of the encoder and to capture key n-gram features, we collect hidden states from multiple layers of the encoder and fuse them to produce a document representation. Let $\mathbf{H}$ denote the input layers in the encoder $\mathbf{h}_l = [\mathbf{h}_1, \ldots, \mathbf{h}_l]$ where $\mathbf{h}_l \in \mathbb{R}^{d*n}$, n is the number of time steps, d the input/output dimension. Dou et al. (2018) proposed hierarchical layer aggregation methods in Fig. 2 (a) and showed the greatest score improvement among the layer aggregation methods in NMT. Formally, each aggregation node $\widehat{\mathbf{h}}_l$ is calculated as

$$\widehat{\mathbf{h}}_l = \begin{pmatrix} \text{AGG}(\mathbf{h}_{2l-1}, \mathbf{h}_{2l}), & l = 1 \\ \text{AGG}(\mathbf{h}_{l-1}, \mathbf{h}_{2l-1}, \mathbf{h}_{2l}), & l > 1 \end{pmatrix}$$

where $\text{AGG}(\mathbf{h}_{2l-1}, \mathbf{h}_{2l})$ and $\text{AGG}(\mathbf{h}_{l-1}, \mathbf{h}_{2l-1}, \mathbf{h}_{2l})$ are computed as

$$\text{AGG}(a, b) = \text{LayerNorm}(\text{FFN}([a; b]) + a + b) \tag{2}$$

$$\text{AGG}(a, b, c) = \text{LayerNorm}(\text{FFN}([a; b; c]) + a + b + c) \tag{3}$$

where FFN denotes feed-forward networks with sigmoid activation in between. The hierarchical layer aggregation combines the encoder layers through a tree structure to preserve and deeply merge features. However, this method requires a larger number

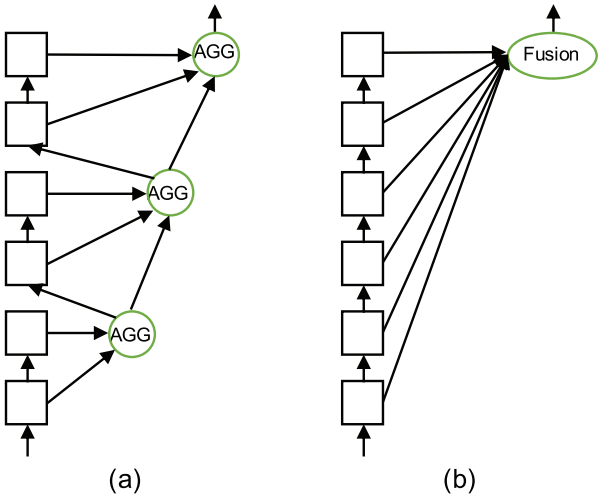

Fig. 2. Illustration of the hierarchical aggregation (a) and proposed deep layer fusion (b).

of additional parameters to aggregate layers as the number of the encoder layers increases, while also increasing the depth of the encoder.

To solve these problems, we introduce deep layer fusion, a simple and efficient layer aggregation method with fewer parameters in Fig. 2 (b). We sum all $\mathbf{h}_l$ into a distributional space $\widehat{\mathbf{h}} = \sum_{l=1}^{L} \mathbf{h}_l$ and feed them into the deep layer fusion module. Let Fusion denote the layer fusion function:

$$\text{Fusion}(\widehat{\mathbf{h}}) = \text{LayerNorm}\Big(\text{FFN}(\widehat{\mathbf{h}})\Big) \tag{4}$$

where FFN denotes feed-forward networks but with GeLU as the activation function. Note that we do not add residual connections. The proposed layer fusion method does not require additional parameters or increase the depth of the encoder network when adding further layers to the encoder in training and evaluation and yields better performance than the other options.

3.4. Gating mechanism

We proposed deep layer fusion, an architecture capable of utilizing all layer representations. In this section, we propose a gating mechanism that is capable of retaining and reusing representations from previous layers, allowing for the construction of rich intermediate layer representations.

As shown in Fig. 3 (b), the original dynamic convolutions model is composed of three modules: output gate o_i^L, dynamic convolution DC_i^L, and feed forward FFN, where i and L denote the position of the word and the number of the layer, respectively. The final output $\mathbf{h}_i^L$ is computed by applying a layer normalization to the FFN output.

$$\mathbf{o}_i^L = (\mathbf{h}_i^{L-1}\mathbf{W}_o^1 + \mathbf{b}_o^1) \otimes \sigma(\mathbf{h}_i^{L-1}\mathbf{W}_o^2 + \mathbf{b}_o^2) \tag{5}$$

$$\mathbf{DC}_i^L = \text{DynamicConv}(\mathbf{o}_i^L) \tag{6}$$

$$\mathbf{h}_i^L = \text{LayerNorm}\Big(\text{FFN}(\mathbf{DC}_i^L)\Big) \tag{7}$$

$\mathbf{W} \in \mathbb{R}^{m \times k \times n}$, $\mathbf{b} \in \mathbb{R}^n$ are learned parameters, where m, n denote the number of the input and output dimensions, respectively, and k denotes the kernel size.

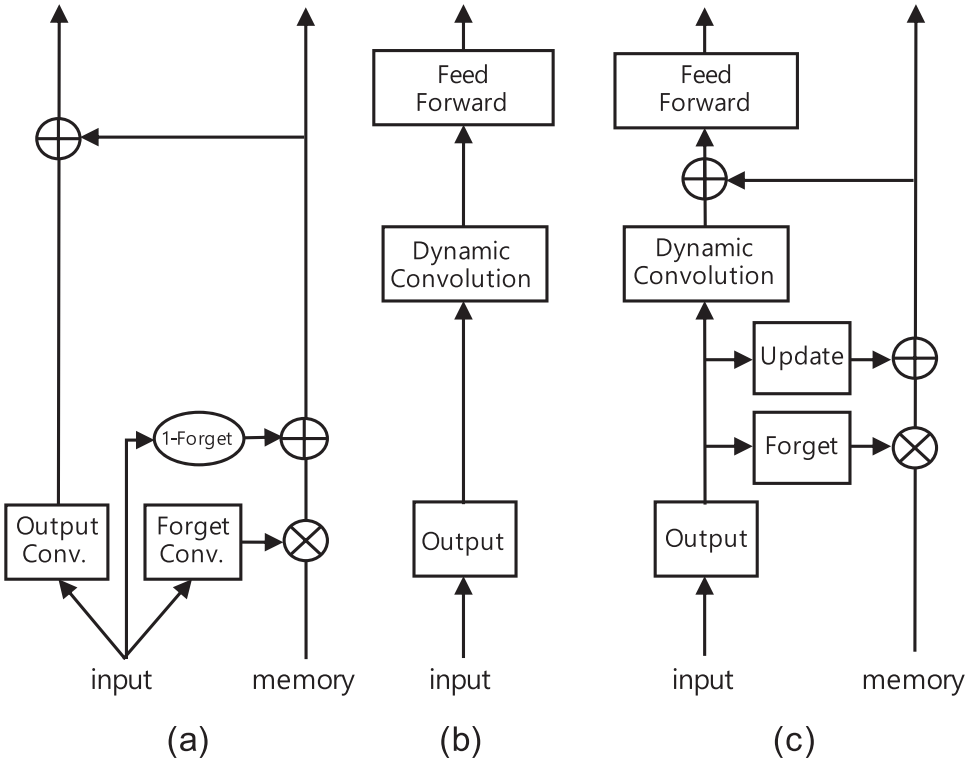

Fig. 3. Illustration of the (a) gated CNNs, (b) original dynamic convolutions, and (c) proposed gated dynamic convolutions.

As shown in Fig. 3 (c), the proposed gated dynamic convolution adds an new memory cell $\mathbf{m}_i^L$ to the original dynamic convolutions, computed by two additional gates: forget $\mathbf{f}_i^L$, and update $\mathbf{u}_i^L$.

$$\mathbf{f}_i^L = \sigma(\mathbf{o}_i^L \mathbf{W}_f + \mathbf{b}_f) \tag{8}$$

$$\mathbf{u}_i^L = \tanh(\mathbf{o}_i^L \mathbf{W}_u^1 + \mathbf{b}_u^1) \otimes \sigma(\mathbf{o}_i^L \mathbf{W}_u^2 + \mathbf{b}_u^2) \tag{9}$$

$$\mathbf{m}_i^L = \mathbf{f}_i^L \otimes \mathbf{m}_i^{L-1} + \mathbf{u}_i^L \tag{10}$$

$$\mathbf{h}_i^L = \text{LayerNorm}\Big(\text{FFN}(\mathbf{D}\mathbf{C}_i^L + \mathbf{m}_i^L)\Big) \tag{11}$$

DynamicConv is an original dynamic convolutions operation. σ is the sigmoid function and $\otimes$ is element-wise multiplication between matrices. Unlike LSTMs, all words have their own memory cells which are updated across layers, rather than time steps. The output gate $\mathbf{o}_i^L$ controls the information flow in the original DynamicConv network while doubling as an initial filter for the update and forget gates described below. The memory cells $\mathbf{m}_i^L$ carry salient contextual information extracted from the previous memory cells $\mathbf{m}_i^{L-1}$ and hidden output $\mathbf{h}_i^{L-1}$. The forget gate $\mathbf{f}_i^L$ controls the amount of information from the previous memory cells to be removed, while the update gate $\mathbf{u}_i^L$ decides what new salient features will be updated into the cell.

Unlike in the previous work, we use GLU as the output gate to help gradients flow through layers, and we separate the function of the update gate from the forget gate. Chen et al. (2018) implements a GRU-style update gate with $1-$*forget* which means it updates previous contextual features $\mathbf{h}_i^{L-1}$ to memory cells by only as much as the amount of information lost from the forget operation. We instead allow our model to update or retain information freely without restriction, even possibly retaining all previous contextual information while also strongly updating with new information. This allows the model to safely retain contextual information from lower to higher layers if necessary. Lastly, we add the information of the memory cells before the FFN in order to combine with the contextual features of the gated dynamic convolutions.

To summarize, we use dynamic convolutions as a base architecture to handle the long inputs of the document summarization task. We inject POS information into the lowest dynamic convolution layer so that the model may more easily extract features from salient keywords in the lower layers. We also propose a fusion layer that assembles useful features from the lower layers. Finally, we apply a gating mechanism that is capable of retaining and reusing representations from previous layers, allowing for the construction of rich intermediate layer representations.

4. Experimental setup

We present our experimental setup for evaluating the performance of our Gated Dynamic convolutions with deep Layer Fusion (GDLF).

4.1. Evaluation metrics

For evaluation metrics, we employ ROUGE scores (Lin, 2004) which are widely used to evaluate summarization quality by comparing overlapping lexical units between a given hypothesis and gold-summary. We evaluate on ROUGE-1 (unigram), ROUGE-2 (bigram), and ROUGE-L (longest common subsequence) using the official ROUGE script for our experiments.

4.2. Baselines

We report our results along with those from a variety of baseline systems. First, we compared GDLF against two extractive approaches: EXT-ORACLE selects summary sentences with the highest F1-ROUGE score between sentences and gold-summaries and LEAD takes the first N sentences of the document as a summary. We also compared GDLF against the LSTM-based abstractive approaches: S2S-attn is a vanilla seq2seq model with attention. PG (See et al., 2017) extends S2S-attn with the pointer mechanism and PG-cov (See et al., 2017) uses both the pointer and coverage mechanisms. DCA+RL (Celikyilmaz et al., 2018) uses two level attention with deep communicating agents, optimized with reinforcement learning. Finally, we compared our model against convolutional seq2seq models: ConvS2S is a vanilla seq2seq model with attention and T-ConvS2S (Narayan et al., 2018) is the topic-conditioned seq2seq model. MMN (Kim et al., 2019) applies attention over multi-level memory networks for improving multi-level abstraction. DC is the original dynamic convolution model. The models excluding PG, PG-cov and DCA+RL do not use the pointer mechanism.

4.3. Datasets

We evaluate our model on three datasets with different characteristics and the basic statistics are shown in Table 1. First, non-anonymized CNN-DM (Hermann et al., 2015; Nallapati et al., 2017) has a strong lead bias and shows weak abstractness which indicates that the dataset follows the typical news writing format in which the most important contents are placed at the front and summaries tend towards extractive rather than abstractive methods. Second, NR-ABS (Grusky et al., 2018) has a weak lead

Table 1
Statistics of the three summarization datasets. The words are assumed to be white-space-delimited.

Datasets	corpus size			avg. doc. length		avg. sum. length	
	train	valid	test	words	sents	words	sents
CNN-DM	287,113	13,368	11,490	752.5	39.8	52.0	4.7
XSUM	204,045	11,332	11,334	431.1	19.8	23.3	1.0
NR-ABS	333,349	36,480	36,595	746.4	35.1	20.8	1.9

bias and exhibits strong abstractness. Lastly, XSUM (Narayan et al., 2018) similarily has a weak lead bias and shows strong abstractness but summaries consist of only a single sentence.

4.4. Model parameters and optimization

We use a shared vocabulary of 30k BPE sub-word units for the encoder-decoder. Injecting POS information, we use the Stanford POS tagger (Toutanova et al., 2003). We truncate the document to 500 tokens and the summary to 150 tokens for all datasets. We use the same single set of training parameters and two different decoding parameters that are tuned on the validation set which is used for model evaluation during training and for early stopping. Our model uses almost the same settings (Wu et al., 2018) reported on CNN-DM. Both DC and GDLF set the dimensions to 256 for word and position embeddings, and GDLF additionally set the dimensions to POS and memory cells as well. We set the number of blocks to 7 for the encoder and 6 for the decoder. We set the encoder kernel sizes to 3, 7, 15, 31x4 and decoder kernel sizes to 3, 7, 15, 31x3. The filter size of FFN was set to 1024 for the encoder and decoder blocks, and 7680 for the layer fusion module. We set the number of attention heads to 8. We trained our models with Adam (Kingma and Ba, 2015) and a cosine learning rate schedule (Loshchilov and Hutter, 2017) with a warmup of 10k steps and a period of 20k updates. The label smoothing parameter was set to 0.1 for a uniform prior distribution of the decoder logits. The dropout rate was set to 0.3. For the decoding parameters, we set the minimum generation length to 15 for NR-ABS and XSUM, to 35 for CNN-DM. We set the length penalty which is lower than 1.0 favors shorter summary and greater than 1.0 favors longer summary to 2.1 for CNN-DM, to 1.3 for NR-ABS and XSUM. Furthermore, we follow a heuristic proposed by (Paulus et al., 2018) that restricts beam search decoding from selecting repeated trigrams for all models.

5. Results

We report our results in Table 2 using the ROUGE metrics on the three aforementioned datasets. LEAD and EXT-ORACLE may each serve as an indicator of how much of the summaries are located at the beginning of the document and how abstract they are, respectively. First, LEAD shows higher scores on CNN-DM compared to the other datasets. It can be deduced that the important contents of the documents in CNN-DM have a higher tendency of being positioned in the first three sentences than that in NR-ABS and XSUM. Second, EXT-ORACLE demonstrates that CNN-DM shows a lower degree of abstraction than the other datasets. It also suggests that the summaries of NR-ABS and XSUM were written using many novel words which did not originally exist in the document. Under these circumstances, we observed that PG performs relatively poorly on highly abstracted datasets (NR-ABS, XSUM). This observation supports the claim that the pointer mechanism leads to excessive copying of long sequences from the source text directly into the summary, and as a consequence, does not show much abstraction capabilities. DC performs

Table 2
Rouge F1 results on the three datasets: CNN-DM, NR-ABS, and XSUM. '*' indicates statistical significance of the corresponding model with respect to the baseline model (DC) on its dataset as given by the 95% confidence interval at most $\pm$ 0.28 as reported by the official Rouge script. '†' indicates the model without the coverage mechanism and better scores than PG-cov. Except DC and GDLF, all scores are referred to the original papers.

Model	CNN-DM			NR-ABS			XSUM		
	R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2	R-L
LEAD	39.6	17.7	36.2	13.7	2.4	11.2	16.3	1.6	12.0
EXT-ORACLE	54.7	30.4	50.8	29.7	10.5	27.2	29.8	8.8	22.7
S2S-attn	31.3	11.8	28.8	6.2	1.1	5.7	28.4	8.8	22.5
PG-cov	39.5	17.3	36.4	†14.7	†2.3	†11.4	†29.7	†9.2	†23.2
DCA+RL	**41.7**	**19.5**	37.9	-	-	-	-	-	-
ConvS2S	37.5	15.1	34.1	-	-	-	31.3	11.1	25.2
T-ConvS2S	-	-	-	-	-	-	31.9	11.5	25.8
MMN	-	-	-	17.5	4.7	14.2	32.0	12.1	26.0
DC	40.3	17.7	37.4	21.4	7.7	18.6	36.0	14.6	29.0
GDLF	*41.0	*18.3	***38.1**	***22.1**	***8.1**	***19.2**	***37.1**	***15.4**	***29.9**

Table 3
Human evaluation results on the three datasets.

Datasets	GDLF vs. DC		
	Win	Lose	Tie
CNN-DM	**25.7**	20.7	3.6
NR-ABS	**26.7**	15.7	7.6
XSUM	**24.3**	22.0	3.7

better than the basic LSTM based seq2seq and convS2S models. Also, DC and GDLF achieve competitive state-of-the-art results on CNN-DM and outperforms the previously state-of-the-art models on NR-ABS and XSUM with on ROUGE scores. GDLF improves the baseline model via the addition of a gating mechanism, layer fusion, and POS augmentation, and shows a statistically significant improvement over DC on all ROUGE metrics on the three datasets.

We conducted a human evaluation on the three datasets in which participants were asked to vote between two system generated summaries, without knowledge of which summary was generated by each system. More specifically, they were presented with a source document and its two system summaries (DC, GDLF) and asked to choose the more relevant summary between the two. The participants were given three options: two to indicate that one of the two summaries better represents the source text, and a *tie* option to indicate neither summary is significantly better than the other. The study was conducted on the same 50 documents and the summaries for each document was voted on by three different participants. Table 3 presents the result of this study. The participants consistently preferred the GDLF summaries to those of DC in all datasets. This indicates that the model learns to generate summaries with more pertinent details by capturing salient information from gated and fusion encoder with POS information.

6. Discussion

6.1. Ablation study

In this section, to validate the usefulness of each module, we first conduct contrastive experiments in which only one module is added to the baseline model as follows: DC denotes the baseline model, +POS denotes the addition of the POS module, +GATE the gating mechanism, and +LF the layer fusion module (first and second blocks). We also evaluate the combination of two modules: +POS+GATE, +POS+LF, and +LF+GATE (third block). Lastly, GDLF denotes our full model with POS, GATE, and LF modules (last block). The results are shown in Table 4. Unsurprisingly, the combination of all modules (GDLF) yields the largest improvement. The second block shows that each of the three modules shows minor improvements when applied to the baseline model consistently for all three datasets, with the largest improvement brought by +LF, followed by +GATE and +POS.

In contrast, the third block shows that the addition of two different modules sometimes fails to outperform the addition of single modules, especially in the case of +POS+GATE which causes seemingly negative effects. We speculate that this is due to the memory cells of the gating mechanism erroneously carrying explicit POS information to the higher layers, which are designed to incorporate paragraph or document level context instead of phrase or sentence level. Yet, as can be observed from the improvement brought about by our full model, the three modules yield the best results when working in tandem. The POS embeddings provide the model with the context of the surrounding tokens, allowing it to create better initial representations of the original text. The gating mechanism functions to capture and relay relevant information between different layers to construct better intermediate representations. The deep layer fusion module then collects these representations to create the final document representation, while also providing direct paths from intermediate encoder layers so that the gating mechanism is not troubled with the task of carrying information only required for lower layer representations.

Table 4
Ablation study on CNN-DM, NR-ABS, and XSUM dataset. '*' indicates statistical significance of the corresponding model with respect to the baseline model (DC) on its dataset as given by the 95% confidence interval at most $\pm$ 0.28 as reported by the official Rouge script.

Model	CNN-DM			NR-ABS			XSUM			Params
	R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2	R-L	
DC	40.3	17.7	37.4	21.4	7.7	18.6	36.0	14.6	29.0	19.2M
DC+POS	40.4	17.8	37.5	21.5	7.8	18.7	36.3	14.9	29.3	19.3M
DC+GATE	40.6	17.9	37.7	21.6	7.9	18.8	36.3	14.8	29.2	21.8M
DC+LF	40.7	18.0	*37.8	21.6	7.8	18.9	*36.6	15.1	29.5	23.1M
DC+POS+GATE	40.6	18.0	37.6	21.3	7.7	18.5	35.8	14.5	28.8	22.0M
DC+POS+LF	40.5	17.9	37.7	*21.8	7.9	18.9	*36.7	15.1	29.7	23.3M
DC+GATE+LF	*40.8	18.1	*38.0	*22.0	*8.1	*19.2	*36.9	*15.3	*29.8	25.8M
GDLF	*41.0	*18.3	*38.1	*22.1	*8.1	*19.2	*37.1	*15.4	*29.9	25.9M

Table 5
Comparisons between deep layer fusion and hierarchical aggregation on the CNN-DM, NR-ABS, and XSUM datasets. '*' indicates statistical significance of the corresponding model with respect to the baseline model (DC) on its dataset as given by a 95% confidence interval at most ± 0.28 as reported by the official Rouge script.

Model	CNN-DM			NR-ABS			XSUM			Params
	R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2	R-L	
DC	40.3	17.7	37.4	21.4	7.7	18.6	36.0	14.6	29.0	19.2M
DC+HAGG	40.5	17.9	37.7	21.5	7.8	18.7	36.4	15.0	29.3	28.6M
DC+LF	40.7	18.0	*37.8	21.6	7.8	18.9	*36.6	15.1	29.5	23.1M
GDHAGG	40.8	18.2	38.0	21.9	8.0	19.0	36.9	15.3	29.8	31.4M
GDLF	*41.0	*18.3	*38.1	*22.1	*8.1	*19.2	*37.1	*15.4	*29.9	25.9M

Next, in Table 5 we show the results of an experimental comparison of our deep layer fusion module with the hierarchical layer aggregation method proposed by Dou et al. (2018). DC denotes the baseline model, +LF the addition of our layer fusion module, and GDLF our full model. +HAGG denotes the addition of HAGG proposed by Dou et al. (2018), while GDHAGG denotes the replacement of LF in our full model with HAGG. Overall, our layer fusion module requires about 5M fewer parameters than HAGG while providing better performance. This is an indication that single addition and fusion operations serve as a better integration function than a nested structure, which has the risk of excessively weakening information from lower layers. By using a structure that does not discriminate between different layers, our LF module can directly utilize fine-grained phrase or sentence level information from the lower layers.

6.2. Impact of gating choices

In this experiment, we further compare the influence of well-known existing gating mechanisms, as shown in Table 6. We change the output gate in Fig. 3 (b) with the following gates (ReLU, GLU, GTU):

$$\mathbf{o}_i^L = \max(0, \mathbf{h}_i^{L-1}) \tag{12}$$

$$\mathbf{o}_i^L = (\mathbf{h}_i^{L-1}\mathbf{W}_o^1 + \mathbf{b}_o^1) \otimes \sigma(\mathbf{h}_i^{L-1}\mathbf{W}_o^2 + \mathbf{b}_o^2) \tag{13}$$

$$\mathbf{o}_i^L = \tanh(\mathbf{h}_i^{L-1}\mathbf{W}_o^1 + \mathbf{b}_o^1) \otimes \sigma(\mathbf{h}_i^{L-1}\mathbf{W}_o^2 + \mathbf{b}_o^2) \tag{14}$$

ReLU denotes the regular ReLU activation function. GLU is the chosen gate in the baseline model (DC) which is aimed at mitigating the vanishing gradient problem of GTU by having linear units coupled to the gates. GCNN is the adaptation of Chen et al. (2018)'s gates to DC as follows:

$$\mathbf{o}_i^L = \tanh(\mathbf{h}_i^{L-1}\mathbf{W}_o^1 + \mathbf{b}_o^1) \otimes \sigma(\mathbf{h}_i^{L-1}\mathbf{W}_o^2 + \mathbf{b}_o^2) \tag{15}$$

$$\mathbf{f}_i^L = \sigma(\mathbf{o}_i^{L-1}\mathbf{W}_f + \mathbf{b}_f) \tag{16}$$

$$\mathbf{u}_i^L = 1 - \mathbf{f}_i^L \otimes \mathbf{o}_i^{L-1} \tag{17}$$

$$\mathbf{m}_i^L = \mathbf{f}_i^L \otimes \mathbf{m}_i^{L-1} + \mathbf{u}_i^L \tag{18}$$

More specifically, the original output gate in DC was replaced with GTU and a forget gate was added, but with the previous layer output as its input. The update gate also takes the previous layer output as its input, but is calculated following the GRU-style $(1 - \textit{forget})$ as in Chen et al. (2018)'s. The rest is the same as our gating mechanism (GATE). GATE w/o update is identical to GATE except that the update gate is implemented in GRU-style in order to investigate the effectiveness of the update gate. Table 6

Table 6
Comparisons of different gating mechanisms on the XSUM test set.

Model	R-1	R-2	R-L	PPL
w/o any gating	33.9	13.0	27.3	13.8
ReLU	35.3	14.0	28.3	13.0
GTU	35.7	14.4	28.8	13.1
GLU	36.0	14.6	29.0	12.7
GCNN	35.9	14.6	29.0	12.8
GATE w/o update	36.2	14.8	29.2	12.5
GATE	**36.3**	**14.8**	**29.3**	**12.4**

shows that our GATE achieves the best ROUGE scores and converges to a lower perplexity than the other gates on XSUM. As expected, GLU, using linear units, showed better results than GTU. GCNN and our GATE showed that memory cells and additional gates (update and forget gate), as well as the output gate further yield improved results. Furthermore, GATE with the explicit update gate yields slightly higher ROUGE scores and lower perplexity. We use this data to support our claim that unrestricted update and forget gates allow salient contextual information to flow more smoothly through layers.

6.3. Named entity vs. POS information

In our work, we choose POS tags as external knowledge to be injected for two reasons. The foremost reason is that the contextual information included in POS tags can be useful in predicting a better convolutional kernel. Second, POS tags can help identify content words, especially proper nouns, which are often either topic words or words that hint at the location of topic words. In text summarization, such named entities are often valuable keywords and Named Entity Recognition (NER), which identifies such named entities, provides more detailed information on them by assigning predetermined categories. For example, the sentence "US exits from WHO" contains two named entities: "US" and "WHO". Through NER, "US" and "WHO" could be each labeled as a PLACE and ORGANIZATION, while a POS tagger would label both of them with the same proper noun tag.

In this study, we replace the POS inputs with NER inputs to investigate the efficacy of categorizing named entities into more fine-grained categories. We also experiment with POS tags augmented with NER information (POS+NER), in which the tags corresponding to named entities are replaced with their specific NER class, with all other tags left as is. For this experiment, to maintain the same tokenization across different taggers, we used a different library, spaCy, to tokenize, POS tag, and NER the texts.

Table 7 compares the results of applying POS, NER, and NER+POS to the baseline model DC. The ROUGE score difference with our other experiments is caused by using a different tokenization. The results show that, albeit without a strong statistical significance, using either NER or POS tags (or both) yield improved performance than not, and that using only POS tags yields the best results. We interpret these results as an indication that explicit contextual information helps the model understand the document, and that using only NER information is weaker than injecting contextual information for all categories of words. We partially attribute the poorer results of NER+POS compared to POS to the relatively lower accuracy of the named entity recognizer.

6.4. Abstractiveness of the generated summaries

We compared the extent of our model's abstractiveness of writing summaries with that of baseline models and the gold-standard summary. Table 8 shows the ratio of unique n-grams that are not present in the document for LEAD, PG, ConvS2S, T-ConvS2S, DC, GDLF, and GOLD (gold-standard summary). As can be seen, the ConvS2S models show the highest proportion of unique n-grams among the neural models, but is still worse than GOLD. Overall, we observed that all models generate a similar proportion of novel n-grams and our GDLF slightly improves the proportion of novel n-grams over DC, which we assume is due to being able to construct better representations in the encoder with gating and layer fusion.

Table 7
Comparisons of different combinations of named entity and part-of-speech information on the XSUM test set.

Model	R-1	R-2	R-L
DC	35.6	14.4	28.7
DC+POS	**36.1**	**14.7**	**29.1**
DC+NER	35.7	14.6	28.9
DC+NER+POS	35.9	14.6	28.9

Table 8
Proportion of unique n-grams in summaries generated by the different models on the XSUM test set.

Model	unigrams	bigrams	trigrams	4-grams
LEAD	0.0	0.0	0.0	0.0
PG	26.2	73.0	90.4	96.1
ConvS2S	**30.0**	**79.1**	**94.2**	**98.1**
T-ConvS2S	29.5	78.9	94.1	98.1
DC	25.3	73.7	91.3	96.7
GDLF	26.9	75.2	91.9	96.9
GOLD	35.0	83.8	95.6	98.5

Table 9
Machine translation accuracy in terms of BLEU on WMT'16 En-DE and WMT'16 En-Ro.

Model	WMT'16 En-De	WMT'16 En-Ro
DC	21.56	25.18
GDLF	22.34	25.71

6.5. Machine translation

In this study, we have explored ways to identify key information from and construct better understandings of long inputs typical in document summarization. In this section, we apply our model to MT in order to test its efficacy in tasks with shorter inputs. We experiment on two language pairs, one for which available training data is abundant and one, relatively scarce. For WMT English to German (En-De), we replicate the setup of Vaswani et al. (2017): we train on WMT'16 training data with 4.5M sentence pairs, validate and test on newstest2013 and newstest2014 with 3k pairs, respectively. For WMT English to Romanian (En-Ro), we train on WMT'16 training data with 612K sentence pairs, validate and test on newsdev2016 and newstest2016 with 2k pairs, respectively. We use byte pair encoding to construct a joint source and target vocabulary with size 40K. We train with Adam using a cosine learning rate schedule with a warmup of 10K steps and a period of 30K updates. Sentence pairs were batched to contain approximately 286K source tokens and the same number of target tokens for En-De, and 43K for En-Ro.

Table 9 shows the results. We obtained improved results over DC for both language pairs; however, the differences were not statistically significant. These results show that our proposed methods effectively alleviates the problems in document summarization caused by long inputs, but are less effective for machine translation, in which the length of the input size is not excessively long.

6.6. Examples of generated summaries

We present three summaries (Gold, DC, GDLF) for three datasets in Tables 10, 11, and 12. Document denotes the input article in the test set, Gold the reference, DC the baseline system summary, and GDLF the proposed system summary.

For the CNN-DM sample document in Table 10, both DC and GDLF demonstrate extractive behaviors. However, several of the selected sentences are extracted from different parts of the document. While DC extracts the second and the third sentences in the lead paragraph, the GDLF successfully selects salient sentences from sentences that are located in the middle of the document, specifically sentence 7 (colored in red). In addition, it successfully captured an important named entity, 'Gary Bowyer' and did not generate unimportant content (colored in blue) such as "Read: liverpool need... ″ or "click here for the latest... ″. This is a good demonstration of the capability of GDLF to successfully encode and extract salient keywords distributed in the document.

For the NR-ABS sample document in Table 11, DC fails to grasp the meaning of the text, erroneously generating a summary (colored in red) saying that the five people killed and the five people shot were separate entities. GDLF, however, accurately conveys that five people were shot and killed, while also getting other details such as the location (colored in blue) of the incident correct. This shows that GDLF competently constructs an understanding of both the fine details and main ideas of the document.

Given the sample document from XSUM in Table 12, the GOLD summary cannot be considered a proper summary because it only mentions a single important point along with some unnecessary details. In contrast, DC and GDLF both generate summaries containing both of the key ideas. Furthermore, GDLF inserts additional details about the subject 'flying scotsman' before the verb

Table 10
Example of generated summaries from the CNN-DM dataset.

Document: steven gerrard 's dream of featuring in the fa cup final at wembley on his 35th birthday remains a reality after liverpool saw off blackburn rovers on wednesday night. gerrard was suspended for the quarter-final replay following his red card for a stamp on manchester united 's ander herrera at anfield on march 22,. yet philippe coutinho came to the rescue for liverpool by scoring the game 's only goal at ewood park to book their place in the semi-final with aston villa. steven gerrard 's dream of featuring in the fa cup final at wembley on his 35th birthday remains a reality. gerrard was suspended for the quarter-final replay following his red card against manchester united. gerrard could feature for liverpool in the fa cup final with liverpool at wembley on may 30,. even blackburn manager gary bowyer feels it would be fitting for liverpool 's captain to lift the trophy before he leaves for major league soccer with la galaxy. ' i wish liverpool all the very best, ' bowyer said. ' it would be fitting if they could win it for gerrard, it would be a great way for him to finish his liverpool career. ' he 's been one of the best midfielders for a long, long time in this country. whilst i was n't wanting it to happen, now we 're out of the cup i certainly hope it does. '

Gold: liverpool beat blackburn rovers 1-0 at ewood park in the fa cup. steven gerrard turns 35 on the day of the fa cup final at wembley. Blackburn 's gary bowyer feels it would be fitting for gerrard to lift fa cup. read : liverpool need philippe coutinho to shine to get into the top four. click here for the latest liverpool news after wednesday 's fa cup win.
DC: steven gerrard was suspended for the quarter-final replay following his red card for a stamp on manchester united 's ander herrera on march 22,. liverpool face aston villa in the fa cup final on may 30,. click here for all the latest liverpool news.
GDLF: steven gerrard was suspended for the quarter-final replay following his red card for a stamp on manchester united 's ander herrera at anfield on march 22,. philippe coutinho came to the rescue for liverpool by scoring the only goal at ewood park to book their place in the fa cup final. even blackburn manager gary bowyer feels it would be fitting for liverpool 's captain to lift the trophy before he leaves for major league soccer with la galaxy.

Table 11
Example of generated summaries from the NR-ABS dataset.

Document: zurich – five people died in a shooting spree late saturday night in northern switzerland in what appears to be a domestic dispute, local police said sunday. the suspected shooter, a 36-year-old man, apparently killed a married couple and the couple 's son in the town of wurenlingen, markus gisin, a police officer investigating the incident, said at a briefing. a neighbor, who appears to have stepped out after hearing the shots, was also killed, mr. gisin said. the shooter than turned the gun on himself, according to mr. gisin. police did n't release the names of the victims or the suspect. the motive for the shooting is unclear, but there appears to be a family connection between the gunman and the family. the aargau cantonal police force is investigating the shooting. cantonal police are similar to state police in the u.s. police were called to the scene shortly after 11 p.m. on saturday by neighbors who heard gunfire. the police discovered the bodies of a 58-year-old man, his 57-year-old wife and their 32-year-old son in their house. the neighbor, 46, was shot after he had come to investigate, mr. gisin said. wurenlingen is about 40 km west of zurich and close to the border with germany. write to neil maclucas at neil.maclucas@wsj.com
Gold: five people have been found dead after a shooting in a town in northern switzerland, including the suspected gunman, police said. **DC:** at least five people were killed and five others were shot in a shooting saturday night in zurich, switzerland, police said. **GDLF:** authorities say five people have been shot and killed in a shooting in wurenlingen, switzerland.

Table 12
Example of generated summaries from the XSUM dataset.

Document: the locomotive has also had its british rail number, 60,103, repainted on its cab. it has been restored for york 's national railway museum at a cost of ps4 0.2 m. test runs were held in cumbria and lancashire in january and february. it is due to make its first run between london and york on 25 february. the trip will be its first official outing bearing its nameplates and sporting its new colours. andrew mclean, head curator at the museum, said : " to finally get flying scotsman fully restored, looking exactly as she should, in steam and alive again will be a really special moment for many people. " he said the shade of green chosen reflected how the engine would have looked during the 1950s and early-60s. ian hewitt, from lancashire-based heritage painting, said it had been a mammoth task to strip off the black paint and repaint the locomotive by hand. " twenty litres of undercoat, 30 litres of gloss and around 20 litres of varnish and there 's about 85 litres of white spirit that we 've gone through, " he said. " it 's the pinnacle of our careers, and to now see it coming together on the world 's most famous locomotive ... it does n't get any bigger for us. " flying scotsman was originally built for the london and north eastern railway company -lrb- lner -rrb- in 1923 and ran daily between london and edinburgh. the locomotive went out of service in 1963 and spent 40 years in private ownership touring the world. it was bought for the nation by the nrm in 2004, using ps415,000 in public donations, a ps365,000 gift from sir richard branson and a ps1 0.8 m grant from the national heritage memorial fund. restoration work had initially been due to be completed in 2011 but was delayed after cracks were found in the chassis. riley and sons ltd, of bury, greater manchester, were appointed to complete the restoration work in 2013.
Gold: the end of flying scotsman 's 10-year restoration project has been marked with it being repainted in traditional british rail green livery. **DC:** one of the world 's most famous locomotives, flying scotsman, is to return to the public for the first time. **GDLF:** flying scotsman, one of the world 's most famous locomotives, is to return to the public for the first time in more than 50 years.

'is' in a human-like manner (colored in red). It also correctly generates the obscure but relevant detail 'in more than 50 years.' (colored in blue).

7. Conclusion

In this paper, we introduce two approaches for effectively aggregating multiple layer representations and applying gating mechanisms into dynamic convolutions. Deep layer fusion efficiently encodes important features throughout multiple hierarchical layers and helps generate more informative and concise summaries. The proposed gating mechanism enables our model to control the information passed on in the hierarchy. Furthermore, we found that supplying POS information explicitly to the encoder successfully improves the model. Our model significantly outperforms the state-of-the-art results on NR-ABS and XSUM while exhibiting comparable scores on CNN-DM.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ICT Creative Consilience program (IITP-2020-2011-1-00783) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation).

References

Bahdanau, D., Cho, K., Bengio, Y., 2014. Neural machine translation by jointly learning to align and translate. arXiv:1409.0473.
Celikyilmaz, A., Bosselut, A., He, X., Choi, Y., 2018. Deep communicating agents for abstractive summarization. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pp. 1662–1675.
Chen, P., Guo, W., Chen, Z., Sun, J., You, L., 2018. Gated convolutional neural network for sentence matching. pp. 2853–2857.
Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y., 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1724–1734.

Dou, Z.-Y., Tu, Z., Wang, X., Shi, S., Zhang, T., 2018. Exploiting deep representations for neural machine translation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 4253–4262.
Dou, Z.-Y., Tu, Z., Wang, X., Wang, L., Shi, S., Zhang, T., 2019. Dynamic layer aggregation for neural machine translation with routing-by-agreement. Thirty-Third AAAI Conference on Artificial Intelligence.
Fan, A., Grangier, D., Auli, M., 2018. Controllable abstractive summarization. In: Proceedings of the 2nd Workshop on Neural Machine Translation and Generation, pp. 45–54.
Gehring, J., Auli, M., Grangier, D., Yarats, D., Dauphin, Y.N., 2017. Convolutional sequence to sequence learning. In: Proceedings of the International Conference on Machine Learning, pp. 1243–1252.
Grusky, M., Naaman, M., Artzi, Y., 2018. Newsroom: a dataset of 1.3 million summaries with diverse extractive strategies. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pp. 708–719.
Hermann, K.M., Kocisky, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., Blunsom, P., 2015. Teaching machines to read and comprehend. Advances in Neural Information Processing Systems, pp. 1693–1701.
Kim, B., Kim, H., Kim, G., 2019. Abstractive summarization of reddit posts with multi-level memory networks. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 2519–2531.
Kingma, D.P., Ba, J., 2015. Adam: A method for stochastic optimization. In: Proceedings of the International Conference on Learning Representations.
Kryściński, W., Paulus, R., Xiong, C., Socher, R., 2018. Improving abstraction in text summarization. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1808–1817.
Lin, C.-Y., 2004. ROUGE: a package for automatic evaluation of summaries. Text Summarization Branches Out. Association for Computational Linguistics, Barcelona, Spain, pp. 74–81.
Liu, P.J., Saleh, M., Pot, E., Goodrich, B., Sepassi, R., Kaiser, L., Shazeer, N., 2018. Generating Wikipedia by summarizing long sequences. In: Proceedings of the International Conference on Learning Representations.
Loshchilov, I., Hutter, F., 2017. SGDR: Stochastic gradient descent with warm restarts. In: Proceedings of the International Conference on Learning Representations.
Nair, V., Hinton, G.E., 2010. Rectified linear units improve restricted Boltzmann machines. In: Proceedings of the 27th international conference on machine learning (ICML-10), pp. 807–814.
Nallapati, R., Zhai, F., Zhou, B., 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence.
Nallapati, R., Zhou, B., dos Santos, C., Gulcehre, C., Xiang, B., 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In: Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning, pp. 280–290.
Narayan, S., Cohen, S.B., Lapata, M., 2018. Dont give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 1797–1807.
Niehues, J., Ha, T.-L., Cho, E., Waibel, A., 2016. Using factored word representation in neural network language models. In: Proceedings of the First Conference on Machine Translation: Volume 1, Research Papers, pp. 74–82.
Van den Oord, A., Kalchbrenner, N., Espeholt, L., Vinyals, O., Graves, A., et al., 2016. Conditional image generation with pixelcnn decoders. Advances in Neural Information Processing Systems, pp. 4790–4798.
Paulus, R., Xiong, C., Socher, R., 2018. A deep reinforced model for abstractive summarization. In: Proceedings of the International Conference on Learning Representations.
See, A., Liu, P.J., Manning, C.D., 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), pp. 1073–1083.
Sennrich, R., Haddow, B., 2016. Linguistic input features improve neural machine translation. In: Proceedings of the First Conference on Machine Translation: Volume 1, Research Papers, pp. 83–91.
Toutanova, K., Klein, D., Manning, C.D., Singer, Y., 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-volume 1. Association for Computational Linguistics, pp. 173–180.
Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Advances in Neural Information Processing Systems, pp. 5998–6008.
Vinyals, O., Fortunato, M., Jaitly, N., 2015. Pointer networks. Advances in Neural Information Processing Systems, pp. 2692–2700.
Wu, F., Fan, A., Baevski, A., Dauphin, Y., Auli, M., 2018. Pay less attention with lightweight and dynamic convolutions. In: Proceedings of the International Conference on Learning Representations.
Yao, J.-g., Wan, X., Xiao, J., 2017. Recent advances in document summarization. Knowl. Inf. Syst. 53 (2), 297–336.
Yu, F., Wang, D., Shelhamer, E., Darrell, T., 2018. Deep layer aggregation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2403–2412.