End-to-End Segmentation-based News Summarization

Yang Liu, Chenguang Zhu and Michael Zeng
Microsoft Cognitive Services Research
{yaliu10, chezhu, nzeng}@microsoft.com

Abstract

In this paper, we bring a new way of digesting news content by introducing the task of segmenting a news article into multiple sections and generating the corresponding summary to each section. We make two contributions towards this new task. First, we create and make available a dataset, SEGNEWS, consisting of 27k news articles with sections and aligned heading-style section summaries. Second, we propose a novel segmentation-based language generation model adapted from pre-trained language models that can jointly segment a document and produce the summary for each section. Experimental results on SEGNEWS demonstrate that our model can outperform several state-of-the-art sequence-to-sequence generation models for this new task.

1 Introduction

In recent years, automatic summarization has received extensive attention in the natural language processing community, due to its potential for processing redundant information. The evolution of neural network models and availability of large-scale datasets have driven the rapid development of summarization systems.

Despite promising results, there are specific characteristics of the traditional summarization task that impedes it to provide more beneficial ways of digesting long news articles. For instance, current news summarization system only provides one genetic summary of the whole article, and when users want to read in more details, the generated summary is not capable of helping navigate the reading. For example, given a news report, current system will output several highlight summaries (Nallapati et al., 2017; Liu and Lapata, 2019; Zhang et al., 2020). Under this circumstance, if a user expect to read more details about one highlight, he will still need to browse the whole article to locate related paragraphs. Meanwhile, when processing a long news article, current systems usually truncate the text and only generate a summary based on the partial article (Cheng and Lapata, 2016a; Zhang et al., 2020). Although this is reasonable since most important content usually lies in the initial portion, it also makes it difficult for users to quickly access information beyond the truncated portion.

In this paper, we propose a new task of Segmentation-based News Summarization. Given a news article, we aim to identify its potential sections and at the same time, to generate the corresponding summary for each section. This new task provides a novel alternative to summarizing a news article. We argue that it can lead to a more organized way of understanding long articles and facilitates a more effective style of reading documents.

First, segmenting a news article can provide a structural organisation of the content, which is not only helpful to reading but also benefit many important NLP tasks. For example, Brown et al. (1983) states that this kind of multi-paragraph division is one of the most fundamental tasks in discourse. However, many expository texts, like news articles, instruction manuals, or textbooks consist of long sequences of paragraphs with very little structural demarcation (Hearst, 1994), and for these documents a subtopical segmentation can be useful. Second, generating concise text descriptions of each sections further reduces the cognitive burden of reading the article (Florax and Ploetzner, 2010). Previous studies (Paice, 1990; Hearst, 1997) present that subtopic segments with their headings is an effective alternative to traditional summarization tasks.

In this paper, we make two main contributions towards the development of Segmentation-based News Summarization systems.[1]

First, we create and publicize a large-scale

[1]Dataset and code will be released at https://github.com/nlpyang/segnews

Findings of the Association for Computational Linguistics: ACL 2022, pages 544 - 554
May 22-27, 2022

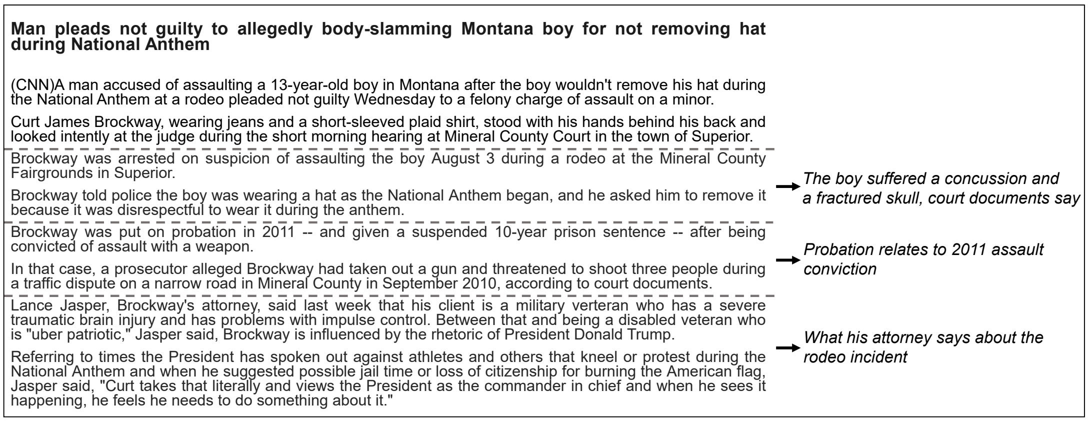


Figure 1: One example from the segmentation-based summarization task SEGNEWS. The news article is taken from a CNN news article and we truncate the article for display. CNN editors have divided this article into several sections and written a heading to section. The goal of this task is to automatically identify sub-topic segments of multiple paragraphs, and generate the heading-style summary for each segment. Dotted lines in the figure indicate segment boundaries. In this article, paragraphs 1,2 are annotated as the first segment, paragraphs 3,4 are annotated as the second segment, paragraphs 5,6 are annotated as the third segment, and paragraphs 7,8 are annotated as the forth segment. To the right of the article are the heading-style summaries for segments. Since the first segment is usually an overview of the news, we do not assign a summary to it.

benchmark, SEGNEWS, for Segmentation-based News Summarization task. Figure 4 shows one example article and its aligned segmentation and summaries from SEGNEWS.

Second, we propose a novel end-to-end approach for this task, which can jointly segment an article while generating the corresponding summaries. These two sub-tasks can learn from each other via a shared encoder. The model is equipped with a segmentation-aware attention mechanism, allowing it to capture segmentation information during summary generation. One important advantage of our framework is that it is a non-invasive adaptation of the Transformer (Vaswani et al., 2017) model, i.e. it does not alter the inner structure of Transformers. And our framework can integrate many pretrained language generation models, including BART (Lewis et al., 2020), GPT (Radford et al., 2019) and UNILM (Bao et al., 2020). This enables our framework to enjoy a high degree of flexibility and better performance.

We compare the proposed framework with several state-of-the-art methods on the SEGNEWS benchmark. Both automatic evaluation and human evaluation demonstrate the superiority of our model.

2 Related Work

2.1 Document Summarization

Document summarization is the task of automatically generating a shorter version text of one or multiple documents while retaining its most important information (Radev et al., 2002). The task has received much attention in the natural language processing community due to its potential for various information access applications. Most large-scale summarization datasets are built on news articles. Popular single-document summarization benchmarks include CNN/DM (Hermann et al., 2015; Nallapati et al., 2016; Cheng and Lapata, 2016a), NYT (Durrett et al., 2016) and XSum (Narayan et al., 2018).

Document summarization can be classified into different paradigms by different factors (Nenkova and McKeown, 2011). And among them, two have consistently attracted attention. *extractive* approaches form summaries by copying and concatenating the most important spans in a document; while in *abstractive* summarization, various text rewriting operations generate summaries using words or phrases that are not in the original text.

Recent approaches to extractive summarization frame the task as a sequence labeling problem by taking advantage of the success of neural network architectures (Bahdanau et al., 2015). The idea is to predict a label for each sentence specifying whether it should be included in the summary. Existing systems mostly rely on recurrent neural networks (Hochreiter and Schmidhuber, 1997) or Transformer model (Vaswani et al., 2017) to encode the document and obtain a vector representation for each sentence (Nallapati et al., 2017; Cheng and Lapata, 2016b; Liu et al., 2019).

In recent years, neural sequence-to-sequence approaches dominate abstractive summarization methods. Rush et al. (2015) and Nallapati et al. (2016) are among the first to apply the neural encoder-decoder architecture to text summarization. See et al. (2017) enhance this model with a pointer-generator network and a coverage mechanism. Pretrained language models have recently emerged as a key technology for improving abstractive summarization systems. These models first pretrain a language model with self-supervised objectives on large corpora and then fine-tune it on summarization datasets. Liu and Lapata (2019) combine a pretrained encoder based on BERT (Devlin et al.) with a randomly initialized decoder, demonstrating substantial gains on summarization performance. MASS (Song et al., 2019) is an encoder-decoder neural model pretrained with the objective of reconstructing a masked text and can be fine-tuned on summarization tasks. BART (Lewis et al., 2020) is an encoder-decoder Transformer (Vaswani et al., 2017) pretrained by reconstructing a text corrupted with several arbitrary noising functions. Bao et al. (2020) design UNILMv2, a Transformer-based neural network pretrained as a pseudo-masked language model.

2.2 Text Segmentation and Outline Generation

Text segmentation has been widely used in the fields of natural language processing and information extraction. Existing methods for text segmentation fall into two categories: unsupervised and supervised. TextTiling (Hearst, 1997) is one of the first unsupervised topic segmentation algorithms. It segments texts in linear time by calculating the similarity between two blocks of words based on the cosine similarity. Choi (2000) introduce a statistical model which can calculate the maximum-probability segmentation of a given text. The TopicTiling (Riedl and Biemann, 2012) algorithm is based on TextTiling, which uses the Latent Dirichlet Allocation to find topical changes within documents. LCSeg (Galley et al., 2003) computes lexical chains of documents and segments texts by a score which captures the sharpness of the change in lexical cohesion.

Supervised methods have also been proposed for text segmentation. Hsueh et al. (2006) integrate lexical and conversation-based features for topic and sub-topic segmentation. Hernault et al. (2010) use CRF to train a discourse segmenter with a set of lexical and syntactic features. Li et al. (2018) propose SEGBOT which uses a neural network model with a bidirectional recurrent neural network together with a pointer network to select text boundaries in the input sequence.

Recently, Zhang et al. (2019) propose Outline Generation task, aiming to identify potential sections of a multi-paragraph document and generate the corresponding section headings as outlines. This task is in form similar to segmentation-based summarization. However, there are two main differences. First, outline generation focused on academic or encyclopaedic documents, where the section headings are extremely short (on average less than two words) and cannot be considered as a summarization task. Second, since outlines care more about briefly describing their corresponding sections, headings in outlines are independently from each other. In segmentation-based summarization, despite describing the sections, heading-style summaries also devote to navigating the reading, and they are usually related and coherent in content.

3 The SEGNEWS Benchmark

3.1 Data Collection

In order to study and evaluate the Segmentation-based News Summarization task, we build a new benchmark dataset SEGNEWS. We take CNN website as our article source. As shown in Figure 1, there are a large part of CNN articles which are divided by editors into several sub-topic sections (see Appendix for details). And each section is assigned a heading-style summary also written by these editors. We collect articles published from 2017 to 2021, covering multiple CNN news channels, including US Politics, Business, Health, Entertainment, Travel and Sports. We filter articles with no sub-topic structures or editor written heading-style summaries. Since the first segment is usually an overview of the news, editors do not assign a summary to it. The resulting dataset contains 26,876 news articles. For each article, it has human annotated segmentation structures and each segment

# news articles	26,876
# paragraphs	40.31
# sections per article	3.17
# tokens per article	1362.24
# tokens per section summary	4.70

Table 1: Data statistics of the SEGNEWS dataset.

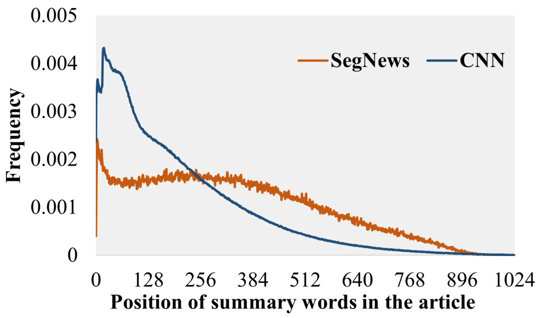


Figure 2: The frequency of the non-stop words in summary appearing at different positions of the source article. The positions range from [0, 1024].

has a human-written heading-style summary.

3.2 Data Statistics

Table 1 shows the overall statistics of our SEGNEWS benchmark dataset. We can see that the news articles in SEGNEWS contain rich structural information and are much longer (1,362 tokens per article) than traditional news summarization datasets: articles in CNN/DM (Cheng and Lapata, 2016b) dataset has an average length of 686.63 tokens and articles in NYT (Sandhaus, 2008) dataset has an average length of 800.04 tokens. This is in line with our motivation that segmentation-based summarization can help readers better understand longer articles.

It has been found that in many news articles, the most important information is often shown at the beginning (Kedzie et al., 2018). We compare SEGNEWS with CNN summarization dataset (Cheng and Lapata, 2016b) to investigate the difference of their positional bias. In Figure 2, we record the position of each non-stop word in the summary that also appears in the article. For both datasets, he beginning of article contains more summary words. However, different from conventional summarization dataset, SEGNEWS dataset has a much smoother position distribution and information in the middle of the article still contributes a lot to the summary.

4 Task Formulation

Given a multi-paragraph article, the segmentation-based summarization task aims to: i) identify sections of the article to unveil its inherent sub-topic structure, where each section consists of neighboring paragraphs with a coherent topic, and ii) generate the heading-style summary for each section to concisely summarize the section. Particularly, in one article, summaries of different sections should be coherent in content and consistent in style.

Formally, let d indicate a document consisting of paragraphs $[p_1, p_2, ..., p_M]$. The segmentation-based summarization task aims to recognize a sequence of section boundaries $[b_1, b_2, \cdots, b_{N-1}]$. These boundaries divide the document into N sections $s_1 = [p_1, ..., p_{b_1}], s_2 = [p_{b_1+1}, ..., p_{b_2}], \cdots, s_N = [p_{b_{N-1}+1}, ..., p_M]$. Meanwhile, summarization systems will generate the corresponding section summaries $[y_1, y_2, ..., y_N]$.

5 Systems for Segmentation-based News Summarization

In this section, we present two different frameworks to tackle the segmentation-based summarization task. In *Pipeline* approach, we first apply a segmentation model to identify potential sections, and then apply a generation model to produce the headings. In *Joint* approach, one neural model is able to jointly segment an article and produce the summaries. To achieve this, we design a novel segmentation-aware attention mechanism, which allows the model to capture segmentation information when generating summaries. This new attention mechanism can also be considered as a non-invasive adaption for conventional Transformer models. Thus, to take the most advantage of existing pre-trained models, we propose SEGUNILM and SEGBART which are respectively based on pre-trained UNILM model and BART model. They can be initialized completely from pre-trained models and achieve substantial improvement on segmentation-based summarization.

5.1 Pipeline Approach

Segmentation model We formulate the section identification process as a sequence labeling task. We insert a special symbol `[X_SEP]` at the boundary of paragraph p_i and p_{i+1} , and then concatenate all paragraphs into a single text input. A neural encoder is then applied to encode this input. Define u_i

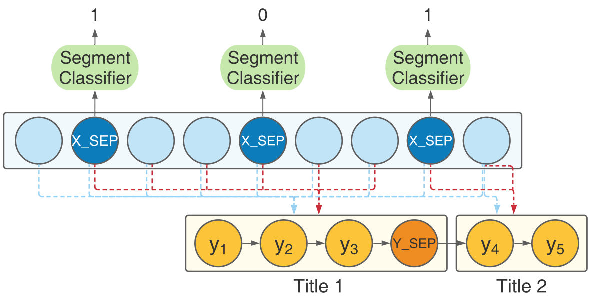


Figure 3: The overall framework of SEGTRANS model. The blue circles indicate input source text, where dark blue circles indicate paragraph boundaries. The yellow circles indicate output target text, where orange circles indicate heading boundaries. Dotted red lines indicate attention heads with segmentation-aware attention mechanism and dotted blue lines indicate attention heads with original full attention mechanism.

as the output vector of `[X_SEP]` after paragraph p_i. We then apply a binary classifier over u_i to obtain $y_i \in \{0, 1\}$. $y_i = 0$ indicates paragraph p_i and p_{i+1} are in one segmentation, and $y_i = 1$ indicates p_{i+1} should be the start of a new segment.

Generation model We then generate an aligned heading-style summary for each identified section s_j. The generation of each heading is independent. Here, we can choose existing extractive or abstractive summarization methods.

- TOPICRANK (Bougouin et al., 2013) is an extractive method for keyphrase extraction which represents a document as a complete graph depending on topical representations. We use the top ranked phrase as the summary for input section;
- SEQ2SEQ represents the sequence-to-sequence neural model, which is usually used in abstractive summarization. It first encodes the concatenated text of all paragraphs within this section, and the decodes the heading in an auto-regressive manner. In experiments, we try both non-pretrained Transformer model and pretrained UNILM and BART models as SEQ2SEQ models.

5.2 Joint Approach

Instead of relying on a pipeline framework, we can also tackle the segmentation-based summarization task with a single encoder-decoder neural model. This brings two main advantages. First, the encoders for segmentation and generation can be shared, benefiting both tasks as a multi-task learner. Second, we can decode all summaries in an auto-regressive manner. In this way, when the decoder generates the l-th heading, it will be exposed to the 1st to $(l-1)$-th generated headings. This is considerately helpful since in a news article, many headings are highly related and coherent in their content.

We use Transformer (Vaswani et al., 2017) as base model for the encoder and decoder. Formally, the encoder maps a sequence of tokens in the source document $\boldsymbol{x} = [x_1, ..., x_n]$ into a sequence of continuous representations $\boldsymbol{t} = [t_1, ..., t_n]$. Then a segment classifier is applied over output vectors of paragraph boundaries to identify correct segments $B = [b_1, b_2, \cdots, b_{N-1}]$ for the input article. The decoder then generates the tokens of target text $\boldsymbol{y} = (y_1, ..., y_m)$ auto-regressively based on the conditional probability: $p(y_1, ..., y_m | x_1, ..., x_n, B)$. As the decoder produces summaries for all sections in one pass, we add a special symbol `[Y_SEP]` between summaries from neighboring sections to indicate their boundaries. However, in this vanilla sequence-to-sequence model, during inference, the decoder is not aware of the segmentation results and can only implicitly use this information when decoding the summaries. Thus, to better jointly learn segmentation and generation tasks, we propose SEGTRANS model, which is equipped with Segmentation-aware Attention mechanism.

Segmentation-aware attention The multi-head decoder-to-encoder attention in a Transformer decoder defines that for a head $z \in \{1, \cdots, n_{head}\}$ at each layer, the model calculates attention probabilities a_{ij}^z against each source token x_j when generating the i-th token y_i.

$$q_i^z = W_q^z Y_i; k_j^z = W_k^z X_j, \tag{1}$$

$$a_{ij}^z = \frac{exp(q_i^{zT} k_j^z)}{\sum_{o=1}^{n} exp(q_i^{zT} k_o^z)}, \tag{2}$$

where $Y_i, X_j \in \mathbb{R}^d$ are the layer's input vectors corresponding to the token y_i and x_j, respectively. $W_q^z, W_k^z \in \mathbb{R}^{d_{head} * d}$ are learnable weights. n is the number of tokens in source input.

However, in segmentation-based summarization, when generating the heading for the i-th section, the decoder should focus more on the input tokens belonging to that section. Thus, we propose the segmentation-aware attention as follows.

We select a subset $\hat{z}$ of decoder heads to apply a segmentation mask to enforce that these heads only attend to the corresponding section. For a head in $\hat{z}$, Eq. 2 is modified to:

$$a_{ij}^{z} = \frac{exp(q_i^{zT} k_j^z) seg(y_i, x_j)}{\sum_{o=1}^{n} exp(q_i^{zT} k_o^z) seg(y_i, x_j)} \quad (3)$$

where $seg(y_i, x_j)$ is a indicator function. It equals 1 if and only if y_i and x_j both belong to the same section, and 0 otherwise. In this manner, parts of the heads in multi-head attention are able to dynamically capture segmentation information, while the other heads still model global features of the entire input article.

We illustrate a detailed example of our framework with segmentation-aware attention in Figure 3. We first encode the source text, and apply a segmentation classification layer over output vectors of paragraph boundaries. For this example input, the model classifies the first and the third paragraph boundaries to be segmentation points. Then the decoder will apply a segmentation-aware multi-head attention over the source outputs. It generates the summary for the first identified section with parts of the attention heads over only the first and the second paragraphs. After generating the first heading ending symbol [Y_SEP], the decoder changes the segmentation-aware attention to the third paragraph for generating the summary for the second section.

The final loss for training SEGTRANS is the summation of the segmentation loss (binary classification loss) $\mathcal{L}_{\text{seg}}$ and generation loss (negative likelihood loss) $\mathcal{L}_{\text{gen}}$.

One advantage of our framework is that it is a non-invasive adaptation of the Transformer model, i.e. it does not alter the inner structure of Transformers. This is important since this adaptation can be applied to many popular pretrained language generation models (e.g. MASS, BART and UNILM), offering our framework a high degree of flexibility and better performance. In this paper, we also augment pre-trained UNILM and BART with this mechanism and propose SEGUNILM and SEGBART to further boost their performance.

6 Experiments

In this section, we conduct experiments on SEGNEWS dataset by comparing our proposed model with several strong baselines.

6.1 Experimental Settings

In pre-processing, all the words in news articles and headings are transformed to lower case and tokenized with wordpiece tokenizer from BERT (Devlin et al.). In data splitting, we guarantee the headings of articles in the test set have low bigram overlap with articles in the training set. We obtain a splitting of 21,748 articles in training set, 2,688 in validation set and 2,444 in test set.

We experiment under both non-pretrained and pretrained settings. In non-pretrained setting, we use a 6-layer Transformer encoder-decoder model (SEGTRANS) with 512 hidden size and 2,048 feed-forward size. In pretrained setting, we propose SEGUNILM and SEGBART which adopts the base version of UNILMv2 (Bao et al., 2020) and the large version of BART (Lewis et al., 2020) as the pretrained model. UNILMv2 is a Transformer-based neural network with 12 Transformer layers and 12 attention heads, pretrained as a pseudo-masked language model. BART is a Transformer-based neural encode-decoder model with 12 layers and 16 attention heads, pretrained via a denoising auto-encoder loss. Label smoothing is used with smoothing factor 0.1. For segmentation-aware attention, we choose the best c (number of segmentation-aware heads) by experiments on the validation set, and $c = 9$ for SEGUNILM and $c = 13$ for SEGBART provide the best performance.

During all decoding we use beam search (size 5), and tune α for the length penalty (Wu et al., 2016) between 0.6 and 1 on the validation set. To guarantee the number of generated headings can match the number of predicted source segments, we take a trick of only generating the end-of-generation token (EOS) when these two numbers match.

We compare the proposed joint models with two sets of strong baselines. The first set of baselines are vanilla sequence-to-sequence models. These models take complete raw articles as input and output the concatenated headings. The second set are pipeline models. As described, these systems first use a segmentor to divide the article into several sections, and then apply a generator to produce summary for each section.

In segmentation-based summarization, summarization systems require segmentation results. We set two settings of segmentation. For the first setting, we provide golden segments to the models to evaluate their performance of generating the summaries when given the correct segments. For the second setting, we require the models to first segment the article and then generate summaries for the predicted segments.

Vanilla Seq2Seq		R1			R2		RL
TRANS		8.66			1.51		8.16
UNILM		19.22			7.18		16.99
Pipeline		With Gold Segments			With Predicted Segments		
Segmentor	Generator	R1	R2	RL	R1	R2	RL
Transformer	Transformer	8.69	1.83	9.09	–	–	–
Transformer	TopicRank	5.09	1.14	6.28	–	–	–
BART	BART	21.42	7.76	19.28	16.01	5.27	14.37
UNILM	UNILM	21.76	8.22	19.75	16.27	5.45	14.65
Joint		R1	R2	RL	R1	R2	RL
SEGTRANS		8.94	1.85	9.35	–	–	–
SEGBART		21.49	8.29	19.52	16.36	5.14	14.96
SEGUNILM		**22.17**	**8.86**	**20.17**	**17.59**	**6.20**	**15.90**

Table 2: ROUGE F1 results on SEGNEWS test set. R1 and R2 are shorthands for ROUGE scores of unigram and bigram overlap; RL is the ROUGE score of longest common subsequence. In pipeline approach, we try combinations of different segmentators and generators. Due to their failure on segmentation, non-pretraind models have very low ROUGE scores with predicted segments, and we do not compare them in the table.

Models	R1	R2	RL
SEGUNILM	22.17	8.86	20.17
(c=12)	22.14	8.81	20.09
(c=8)	22.13	8.84	20.10
(c=4)	21.39	7.99	19.23
(c=0)	19.85	7.74	17.62
(w/o seg loss)	22.06	8.66	20.02

Table 3: Ablation study results on SEGNEWS. We compare multiple variants of SEGUNILM. c indicates the number of decoder heads modified into segmentation-aware attention. Be default, SEGUNILM uses $c = 9$ to achieve the best performance. We also present a SEGUNILM model without (w/o) segmentation classification loss, and it is trained solely by generation loss.

6.2 Evaluation Metrics

Evaluation metrics for summarization performance are ROUGE (Lin, 2004) F1 scores of the generated headings against the gold headings. We report unigram and bigram overlap (ROUGE-1 and ROUGE-2) as a means of assessing informativeness and the longest common subsequence (ROUGE-L) as a means of assessing fluency.

We use standard metrics *Pk* (Beeferman et al., 1999) and *WinDiff* (Pevzner and Hearst, 2002) to evaluate segmentation results. Lower scores of these two metrics indicate that the predicted segmentation is closer to the ground truth. A EVEN baseline is included for comparison where it segments the whole article evenly.

6.3 Results

Table 2 describes our summarization results on the SEGNEWS dataset. The first vertical block includes the results of vanilla sequence-to-sequence models. TRANS is the non-pretrained Transformer encoder-decoder model. UNILM and BART are two pretrained baseline models. The second vertical block contains the results of pipeline models. We present the combinations of different segmentation models and generation models. For segmentor, we experiment non-pretrained Transformer model and pretrained BART and UNILM models. For generator, we also include TOPICRANK, which is a classical extractive summarization method.

The last vertical block includes the results of our joint models: SEGTRANS, SEGBART and SEGUNILM. They respectively rely on non-pretrained Transformer and pretrained BART and UNILM as backbone models. Segmentation-aware attention mechanism is used to augment these jointly trained systems.

We can see vanilla sequence-to-sequence models with no segmentation information input perform poorly on this task. End-to-end SEGUNILM model achieves the best performance among all systems. SEGUNILM outperforms the best pipeline system under both settings when gold segments or predicted segments are provided. This indicates SEGUNILM has better overall performance and will be more useful when applied as practical applications. It also shows higher summarization results than vanilla UNILM model, confirming the effectiveness of segmentation-aware attention mechanism. SEGBART and SEGTRANS also show similar superiority over their pipeline versions. Examples of system output are shown in Table 4.

Table 3 summarizes ablation studies aiming to assess the contribution of individual components of SEGUNILM. We first modify SEGUNILM by varying c, the number of heads of segmentation-aware attention. We can see the best results of ROUGE

Title: One JFK conspiracy theory that could be true	
GOLD	1. LBJ had it done; 2. The military industrial complex did it; 3. The mob did it; 4. Oswald acted alone as part of an unknown conspiracy; 5. The CIA did it
Pipeline UNILM	Those Kennedys will never embarrass me again; Did Kennedy want to withdraw us troops from Vietnam ?; 3. Different mobs; other conspirators ?; Would America be OK with that ?
SEGBART	1. They thought he was a crook; 2. He was going to pull American troops out of Vietnam; 3. The mob did this; 4. There were others, but who were they?; 5. The CIA ordered the killing
SEGUNILM	1. Those Kennedy's will never embarrass me again; 2. He said he'd pull troops out of Vietnam; 3. Mob members claim they were witnesses to the alleged shootings; 4. there were more people who knew where Oswald was; 5. The CIA didn't release any of the good stuff
Title: This man is tasked with finding out who failed Larry Nassar's victims	
GOLD	Seeking justice; A very youthful 68-year-old; A model independent prosecutor
Pipeline UNILM	Searching for truth; He couldn't stay retired; He didn't have an agenda
SEGBART	Searching for the truth; Working with juveniles; No stone unturned
SEGUNILM	Searching for the truth; He's has to do something; He doesn't have an agenda

Table 4: GOLD reference summaries and automatic summaries produced by pipeline UNILM, SEGBART and SEGUNILM on the SEGNEWS datasets. Semicolons indicate the boundaries of headings.

Model	WD	PK
EVEN	0.469	0.450
Transformer	0.563	0.462
BART	0.484	0.411
UNILM	0.479	0.391
SEGBART	0.471	0.405
SEGUNILM	0.462	0.380

Table 5: Experimental results on document segmentation task. WD indicates *WinDiff* metric.

Model	Quality	Fluency
Pipeline UNILM	1.93	2.62
SEGUNILM	2.17	2.59
Gold	2.44	2.79

Table 6: Human evaluation results based on summary quality and fluency.

are achieved when $c = 9$. With more or less heads modified as segmentation-aware attention heads, the summarization performance show a clear trend of decreasing. Also, as shown in the last column, when segmentation layer and segmentation loss are removed, we observe a sharp decrease on ROUGE scores. The results prove that both segmentation-aware attention and joint training provide improvement to the summarization results.

Table 5 describes the results on news segmentation task. SEGUNILM achieves the lowest WD and PK scores, revealing its ability to identify the structure of a news article. Compared with UNILM model without the segmentation-aware attention, SEGUNILM shows clear superiority on both metrics. The same trend is also observed in BART related models.

6.4 Human Evaluation

In addition to automatic evaluation, we also assess system performance by eliciting human judgments on 20 randomly selected test instances. The evaluation study assess the overall quality and fluency of the summaries by asking participants to rate them. We present the news article to evaluators along with system generated heading-style summaries, and we ask evaluators to read the complete article, and give scores based on summary quality and fluency respectively. Participants can have three scores (1-low quality/fluency, 2-median quality/fluency, 3-high quality/fluency).

Gold summaries, outputs from pipeline UNILM and SEGUNILM models are compared in evaluation. We invite three evaluators with linguist background to conduct the human evaluation. The averaged results are shown in Table 4. Overall, we observe pipeline UNILM and SEGUNILM perform similarly on fluency, but SEGUNILM shows its superiority on summary quality. Gold summaries are marginally better than automatic generated summaries.

7 Conclusion

In this work, we proposed a new task, segmentation-based news summarization. It aims to segment a news article into multiple sections and generate the corresponding summary to each section. This new task provides a novel alternative to digesting a news article. We built a new benchmark dataset SEGNEWS to study and evaluate the task. Furthermore, we designed a segmentation-aware attention mechanism, which allows neural decoder to capture segmentation information in the source texts. We jointly train the model for generating summaries and recognizing news segments. Experimental results on SEGNEWS demonstrate that our framework produces better segmentation-based summaries than competitive systems.

8 Ethical Statement

We honor and support the ACL Code of Ethics. We have used only the publicly available news articles from the CNN website and adhere to their only-for-research-purpose guideline. Meanwhile, to make sure the downstream usage of the data will not break the permission of CNN website, we only release the URLs of these articles along with a script to download and process them.

The content of the news and summaries only reflect the views of the media, and should be viewed with discretion.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *In Proceedings of the 3rd International Conference on Learning Representations*, San Diego, California.

Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Xiaodong Liu, Yu Wang, Songhao Piao, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2020. UniLMv2: Pseudo-masked language models for unified language model pre-training. *arXiv preprint arXiv:2002.12804*.

Doug Beeferman, Adam Berger, and John Lafferty. 1999. Statistical models for text segmentation. *Machine Learning*, 1(34):177–210.

Adrien Bougouin, Florian Boudin, and Béatrice Daille. 2013. Topicrank: Graph-based topic ranking for keyphrase extraction. In *International joint conference on natural language processing (IJCNLP)*, pages 543–551.

Gillian Brown, Gillian D Brown, Gillian R Brown, Brown Gillian, and George Yule. 1983. *Discourse analysis*. Cambridge university press.

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

Jianpeng Cheng and Mirella Lapata. 2016b. Neural summarization by extracting sentences and words. In *Proceedings of the ACL Conference*.

Freddy Y. Y. Choi. 2000. Advances in domain independent linear text segmentation. In *1st Meeting of the North American Chapter of the Association for Computational Linguistics*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the NAACL Conference*.

Greg Durrett, Taylor Berg-Kirkpatrick, and Dan Klein. 2016. Learning-based single-document summarization with compression and anaphoricity constraints. In *Proceedings of the ACL Conference*.

Mareike Florax and Rolf Ploetzner. 2010. What contributes to the split-attention effect? the role of text segmentation, picture labelling, and spatial proximity. *Learning and instruction*, 20(3):216–224.

Michel Galley, Kathleen McKeown, Eric Fosler-Lussier, and Hongyan Jing. 2003. Discourse segmentation of multi-party conversation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 562–569.

Marti A. Hearst. 1994. Multi-paragraph segmentation expository text. In *32nd Annual Meeting of the Association for Computational Linguistics*, pages 9–16, Las Cruces, New Mexico, USA. Association for Computational Linguistics.

Marti A Hearst. 1997. Text tiling: Segmenting text into multi-paragraph subtopic passages. *Computational linguistics*, 23(1):33–64.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *Advances in neural information processing systems*, 28.

Hugo Hernault, Danushka Bollegala, and Mitsuru Ishizuka. 2010. A sequential model for discourse segmentation. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 315–326. Springer.

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

Pei-Yun Hsueh, Johanna D Moore, and Steve Renals. 2006. Automatic segmentation of multiparty dialogue. In *11th Conference of the European Chapter of the Association for Computational Linguistics*.

Chris Kedzie, Kathleen McKeown, and Hal Daumé III. 2018. Content selection in deep learning models of summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1818–1828, Brussels, Belgium.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Jing Li, Aixin Sun, and Shafiq R Joty. 2018. Segbot: A generic neural text segmentation model with pointer network. In *IJCAI*, pages 4166–4172.

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

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China.

Yang Liu, Ivan Titov, and Mirella Lapata. 2019. Single document summarization as tree induction. In *Proceedings of the NAACL Conference*.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the AAAI Conference*.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Caglar Gulcehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't 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*, pages 1797–1807, Brussels, Belgium.

Ani Nenkova and Kathleen McKeown. 2011. Automatic summarization. *Foundations and Trends in Information Retrieval*, 5(2–3):103–233.

Chris D Paice. 1990. Constructing literature abstracts by computer: techniques and prospects. *Information Processing & Management*, 26(1):171–186.

Lev Pevzner and Marti A Hearst. 2002. A critique and improvement of an evaluation metric for text segmentation. *Computational Linguistics*, 28(1):19–36.

Dragomir R. Radev, Eduard Hovy, and Kathleen McKeown. 2002. Introduction to the special issue on summarization. *Computational Linguistics*, 28(4):399–408.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Martin Riedl and Chris Biemann. 2012. Topictiling: a text segmentation algorithm based on lda. In *Proceedings of ACL 2012 Student Research Workshop*, pages 37–42.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal.

Evan Sandhaus. 2008. The New York Times Annotated Corpus. *Linguistic Data Consortium, Philadelphia*, 6(12).

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

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. MASS: Masked sequence to sequence pre-training for language generation. In *Proceedings of International Conference on Machine Learning*, pages 5926–5936, Long Beach, California.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. In *arXiv preprint arXiv:1609.08144*.

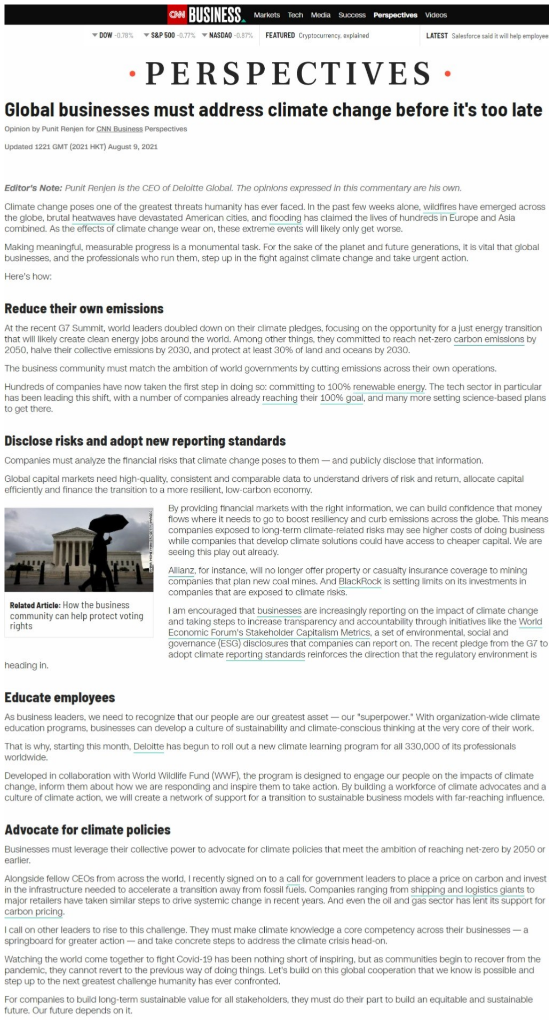
Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Ruqing Zhang, Jiafeng Guo, Yixing Fan, Yanyan Lan, and Xueqi Cheng. 2019. Outline generation: Understanding the inherent content structure of documents. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 745–754.

A Build SEGNEWS from CNN website

The SEGNEWS dataset is built from news articles on CNN website. For many news reports on CNN, news editors manually divide them into several sections and write a heading-style summary for each section. As illustrated in Figure 1, in a display of this news article[2], it has a general title "Global businesses must address climate change before it's too late". Below the title, there are several paragraphs of news content. This news article is divided into 5 sections. Despite the first section, the other 4 sections are assigned with their heading-style summaries: "Reduce their own emissions", "Disclose risks and adopt new reporting standards", "Educate employees" and "Advocate for climate policies".

We crawl news articles like this from CNN website. Articles without segmentation information or headings are filtered. The resulting SEGNEWS dataset contains 26,876 articles. Each instance in SEGNEWS consist of a news article, its segmentation structure and heading-style summaries for each segments.



CNN BUSINESS Markets Tech Media Success Perspectives Videos

DOW -0.78% S&P 500 -0.77% NASDAQ -0.87% FEATURED Cryptocurrency, explained LATEST Salesforce said it will help employee

PERSPECTIVES

Global businesses must address climate change before it's too late

Opinion by Punit Renjen for CNN Business Perspectives

Updated 1221 GMT (2021 HKT) August 9, 2021

Editor's Note: *Punit Renjen is the CEO of Deloitte Global. The opinions expressed in this commentary are his own.*

Climate change poses one of the greatest threats humanity has ever faced. In the past few weeks alone, wildfires have emerged across the globe, brutal heatwaves have devastated American cities, and flooding has claimed the lives of hundreds in Europe and Asia combined. As the effects of climate change wear on, these extreme events will likely only get worse.

Making meaningful, measurable progress is a monumental task. For the sake of the planet and future generations, it is vital that global businesses, and the professionals who run them, step up in the fight against climate change and take urgent action.

Here's how:

Reduce their own emissions

At the recent G7 Summit, world leaders doubled down on their climate pledges, focusing on the opportunity for a just energy transition that will likely create clean energy jobs around the world. Among other things, they committed to reach net-zero carbon emissions by 2050, halve their collective emissions by 2030, and protect at least 30% of land and oceans by 2030.

The business community must match the ambition of world governments by cutting emissions across their own operations.

Hundreds of companies have now taken the first step in doing so: committing to 100% renewable energy. The tech sector in particular has been leading this shift, with a number of companies already reaching their 100% goal, and many more setting science-based plans to get there.

Disclose risks and adopt new reporting standards

Companies must analyze the financial risks that climate change poses to them — and publicly disclose that information.

Global capital markets need high-quality, consistent and comparable data to understand drivers of risk and return, allocate capital efficiently and finance the transition to a more resilient, low-carbon economy.

Related Article: How the business community can help protect voting rights

By providing financial markets with the right information, we can build confidence that money flows where it needs to go to boost resiliency and curb emissions across the globe. This means companies exposed to long-term climate-related risks may see higher costs of doing business while companies that develop climate solutions could have access to cheaper capital. We are seeing this play out already.

Allianz, for instance, will no longer offer property or casualty insurance coverage to mining companies that plan new coal mines. And BlackRock is setting limits on its investments in companies that are exposed to climate risks.

I am encouraged that businesses are increasingly reporting on the impact of climate change and taking steps to increase transparency and accountability through initiatives like the World Economic Forum's Stakeholder Capitalism Metrics, a set of environmental, social and governance (ESG) disclosures that companies can report on. The recent pledge from the G7 to adopt climate reporting standards reinforces the direction that the regulatory environment is heading in.

Educate employees

As business leaders, we need to recognize that our people are our greatest asset — our "superpower." With organization-wide climate education programs, businesses can develop a culture of sustainability and climate-conscious thinking at the very core of their work.

That is why, starting this month, Deloitte has begun to roll out a new climate learning program for all 330,000 of its professionals worldwide.

Developed in collaboration with World Wildlife Fund (WWF), the program is designed to engage our people on the impacts of climate change, inform them about how we are responding and inspire them to take action. By building a workforce of climate advocates and a culture of climate action, we will create a network of support for a transition to sustainable business models with far-reaching influence.

Advocate for climate policies

Businesses must leverage their collective power to advocate for climate policies that meet the ambition of reaching net-zero by 2050 or earlier.

Alongside fellow CEOs from across the world, I recently signed on to a call for government leaders to place a price on carbon and invest in the infrastructure needed to accelerate a transition away from fossil fuels. Companies ranging from shipping and logistics giants to major retailers have taken similar steps to drive systemic change in recent years. And even the oil and gas sector has lent its support for carbon pricing.

I call on other leaders to rise to this challenge. They must make climate knowledge a core competency across their businesses — a springboard for greater action — and take concrete steps to address the climate crisis head-on.

Watching the world come together to fight Covid-19 has been nothing short of inspiring, but as communities begin to recover from the pandemic, they cannot revert to the previous way of doing things. Let's build on this global cooperation that we know is possible and step up to the next greatest challenge humanity has ever confronted.

For companies to build long-term sustainable value for all stakeholders, they must do their part to build an equitable and sustainable future. Our future depends on it.

Figure 4: One example news article on CNN website. It contains human-annotated segments and heading-style summaries.

[2]https://edition.cnn.com/2021/08/09/perspectives/climate-change-deloitte-global-ceo-punit-renjen/index.html