Efficient Attentions for Long Document Summarization

Luyang Huang [1] **Shuyang Cao**[1] **Nikolaus Parulian**[2] **Heng Ji**[2] **Lu Wang**[1]
[1]Computer Science and Engineering, University of Michigan, Ann Arbor, MI
[2]Department of Computer Science, University of Illinois at Urbana-Champaign, IL
[1]{lyhuang, caoshuy, wangluxy}@umich.edu
[2]{nnp2, hengji}@illinois.edu

Abstract

The quadratic computational and memory complexities of large Transformers have limited their scalability for long document summarization. In this paper, we propose **HEPOS**, a novel *efficient encoder-decoder attention with head-wise positional strides* to effectively pinpoint salient information from the source. We further conduct a systematic study of existing efficient self-attentions. Combined with HEPOS, we are able to process ten times more tokens than existing models that use full attentions. For evaluation, we present a new dataset, **GOVREPORT**, with significantly longer documents and summaries. Results show that our models produce significantly higher ROUGE scores than competitive comparisons, including new state-of-the-art results on PubMed. Human evaluation also shows that our models generate more informative summaries with fewer unfaithful errors.

1 Introduction

Long documents, such as scientific papers and government reports, often discuss substantial issues at length, and thus are time-consuming to read, let alone to comprehend. Generating abstractive summaries can help readers quickly grasp the main topics, yet prior work has mostly focused on short texts (containing hundreds of words), e.g., news articles (Gehrmann et al., 2018; Liu and Lapata, 2019; Zhang et al., 2019).

Model training efficiency and *summary quality* present a pair of challenges for long document summarization. State-of-the-art systems (Lewis et al., 2020; Zhang et al., 2019) are built upon Transformer (Vaswani et al., 2017), which uses attentions to compute pairwise relations between tokens. Such framework has quadratic time and memory complexities, and is too costly for long documents [1]. Solutions have been proposed to reduce the calculation of *encoder self-attentions* (Wang et al., 2020c; Zaheer et al., 2020) by selectively attending to neighboring tokens (Beltagy et al., 2020; Child et al., 2019) or relevant words (Kitaev et al., 2020; Tay et al., 2020a). Yet, these methods do not apply to *encoder-decoder attentions* in summarization models since they collaborate and dynamically pinpoint salient content in the source as the summary is decoded. Truncation is commonly used to circumvent the issue. However, training on curtailed content further aggravates "hallucination" in existing abstractive models (Maynez et al., 2020).

We argue that summarizing long documents (e.g., with thousands of words or more) requires efficient handling of both types of attentions. To this end, we propose an efficient encoder-decoder attention with **head-wise positional strides (HEPOS)**, where the attention heads follow a strided pattern and have varying starting positions. HEPOS reduces computational and memory costs while (1) maintaining the power of emphasizing important tokens, and (2) preserving the global context per head. HEPOS successfully doubles the processed input sequence size, when combined with any encoder. To the best of our knowledge, we are the first to study efficient encoder-decoder attentions and provide a systematic comparison of diverse encoder attentions for the task of summarization.[2]

For evaluation, we collect **a new large-scale dataset, GOVREPORT**, consisting of about 19.5k U.S. government reports with expert-written abstractive summaries.[3] GOVREPORT has two important features: (1) It contains significantly longer documents (9.4k words) and summaries (553 words) than existing datasets, such as PubMed and arXiv (Cohan et al., 2018) (see Table 2); (2) Salient

[1] For instance, to fine-tune BART on documents of 10K tokens with a batch size of 1, 70GB of memory is needed for encoder attentions, and 8GB for encoder-decoder attentions.

[2] Our code is released at https://github.com/luyang-huang96/LongDocSum.

[3] GOVREPORT can be downloaded from https://gov-report-data.github.io.

content is spread throughout the documents, as opposed to cases where summary-worthy words are more heavily concentrated in specific parts of the document. These properties make GovReport an important benchmark for producing long document summaries with multiple paragraphs.

We conduct experiments on GovReport and scientific papers in PubMed and arXiv. First, when summarizing documents of the same length, Hepos *attention yields significantly better ROUGE scores* than a non-trivial comparison that projects attentions into low-rank space (Wang et al., 2020c). Second, when trained on the same GPU, Hepos attention, combined with sparse encoder attentions, is able to read more than 10K words and obtains significantly higher ROUGE scores on GovReport and new state-of-the-art results on PubMed, compared with full encoder-decoder attention models which can process at most 5K input words. Human judges further rate the summaries generated by our models to be *more informative and faithful*.

We further propose **a new evaluation metric for faithfulness**, inspired by APES (Eyal et al., 2019), a fill-in-the-blank QA metric for summary evaluation. With questions generated from references, our metric, APES$_{src}$, compares QA answers by reading the source and the system summary. It is shown to be better correlated with human judgment than the original metric and an entailment-based scorer (Kryscinski et al., 2020).

The rest of the paper is organized as follows. We describe efficient encoder attentions in prior work in § 2, and formulate our proposed encoder-decoder attention in § 3. The GovReport data is presented in § 4. We then share details on evaluation metrics (§ 5) and experimental results (§ 6). Additional related work is listed in § 7, with conclusion in §8.

2 Prior Work on Efficient Encoder Attentions

Transformer models are built upon multi-head attentions in multiple layers. The attention is calculated as Attention$(\mathbf{Q}, \mathbf{K}, \mathbf{V}) =$ softmax$(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}})\mathbf{V}$, where $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ are query, key, and value matrices, each consisting of n vectors for a document with n tokens, thus the quadratic memory footprint.

Here, we present an overview of representative methods for efficient encoder self-attentions (henceforth "encoder attentions") that can be built upon large pre-trained seq2seq models, e.g., BART (Lewis et al., 2020). We follow the naming convention of Tay et al. (2020b), and summarize their *memory complexities* and numbers of *newly learned parameters* in Table 1.

Model	**Complexity**	**# New Para.**
Full	$\mathcal{O}(n^2)$	—
Encoder Self-attentions		
I. Fixed Patterns		
Sliding Window (2020)	$\mathcal{O}(nw)$	0
Adaptive Span (2019)	$\mathcal{O}(n\hat{w})$	$\mathcal{O}(1)$
Global Tokens (2020)	$\mathcal{O}(2ng)$	0
Stride (2019)	$\mathcal{O}(n^2/s)$	0
Random (2020)	$\mathcal{O}(nr)$	0
II. Low-rank		
Linformer (2020c)	$\mathcal{O}(nk)$	$\mathcal{O}(n)$
III. Learnable Patterns		
LSH (2020)	$\mathcal{O}(lnb_l)$	0
Sinkhorn (2020a)	$\mathcal{O}(2nb_s)$	0
Encoder-decoder Attentions		
Hepos (ours)	$\mathcal{O}(mn/s_h)$	0
Linformer	$\mathcal{O}(mk)$	$\mathcal{O}(n)$

Table 1: Summary of efficient Transformer attentions on *memory complexity* and *newly learned parameters* compared with full attentions at each layer. m and n are lengths of the input and the output. See § 2 and § 3 for model-specific hyperparameters.

2.1 Fixed Patterns

Fixed patterns are used to limit the scope of attentions. In our experiments, in addition to window-based attentions, we also combine them with global tokens, stride patterns, or random attentions.

Sliding window attentions (Beltagy et al., 2020) aim to capture the local context, which is critical for language understanding (Liu* et al., 2018; Child et al., 2019). Concretely, each query token attends to $w/2$ neighboring tokens on both left and right, yielding a memory complexity of $\mathcal{O}(nw)$.

Adaptive span is proposed by Sukhbaatar et al. (2019) to learn attention windows at different layers. This is implemented by learning a masking function for each head independently. In practice, the adaptive span attention has a complexity of $\mathcal{O}(n\hat{w})$, where $\hat{w}$ is the maximum values of predicted spans for all heads. Besides, it introduces $\mathcal{O}(1)$ new parameters for learning spans.

Global tokens (Beltagy et al., 2020) are often added to sliding windows to let pre-selected tokens attend to the full sequence, to build global representations. Importantly, global attention operations are symmetric, i.e., a global token is also attendable to all tokens in the sequence. We select the first g tokens as global tokens, as leading sentences are often important for summarization. Memory complexity is $\mathcal{O}(2ng)$ due to the symmetric attentions.

Stride patterns are proposed by Child et al. (2019) to capture long term interactions, where each query attends to every s-th token, with s as the stride size. It thus has a complexity of $\mathcal{O}(n^2/s)$.

Random attention is motivated by the fact that randomly constructed graphs with $\tilde{\Theta}(n)$ edges can approximate the complete graphs spectrally (Zaheer et al., 2020). Zaheer et al. (2020) propose to allow each query to attend to r random keys, resulting in a complexity of $\mathcal{O}(nr)$. For efficient implementations, input tokens are first segmented into blocks. Tokens in the same block attend to tokens in another randomly selected block.

2.2 Low-rank Methods

Wang et al. (2020c) show that self-attention matrices are low-rank. They propose **Linformer** that linearly projects key and value matrices into a low-dimensional space, e.g., from n to k, to achieve a $\mathcal{O}(nk)$ complexity. It also introduces $\mathcal{O}(n)$ new parameters for projection matrix learning.

2.3 Learnable Patterns

Recently, learnable sparse attentions are proposed to better capture both local and global contexts than attentions based on fixed patterns.

Locality-sensitive hashing (LSH) attentions use a random-projection hashing function to hash similar queries and keys into the same buckets in l rounds (Kitaev et al., 2020). Attentions are then computed among tokens within each bucket. For bucket size b_l, the complexity of LSH attention is $\mathcal{O}(lnb_l)$.

Sinkhorn attentions first segment a sequence into blocks, which are then arranged by a learned Sinkhorn sorting network (Tay et al., 2020a). Given the new permutation, each query attends to b_s tokens within the same block to maintain the local context and another b_s tokens in a neighboring block to capture global interactions. Its complexity is $\mathcal{O}(2nb_s)$.

2.4 Other Attentions

We also describe several notable methods that are not suitable for our experiments and excluded from this study: **Recurrence** over input segments are tailored for an autoregressive decoder only (Dai et al., 2019); **memory** methods use a separate memory module to attend to full sequences (Lee et al., 2019), which share a similar theoretical foundation as global tokens; and **kernel** methods over attentions require training models from scratch (Choromanski et al., 2020; Katharopoulos et al., 2020).

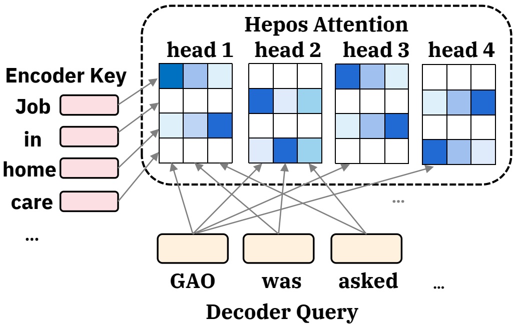

Figure 1: A toy example of our HEPOS attention, with a stride of 2 and four attention heads. Dark colors indicate that heads 1 and 3 attend to the first and third tokens (“Job" and “home") in the input, heads 2 and 4 look at the second and fourth words (“in" and “care").

3 Encoder-decoder Attention with Head-wise Positional Strides (Hepos)

The efficient design of encoder-decoder attentions with head-wise positional strides (HEPOS) allows models to consume longer sequences. Concretely, our design is motivated by two observations: (1) Attention heads are redundant (Voita et al., 2019). (2) Any individual head rarely attends to several tokens in a row (Clark et al., 2019). Therefore, as illustrated in Fig. 1, HEPOS uses separate encoder-decoder heads on the same layer to cover different subsets of source tokens at fixed intervals. Each head starts at a different position, and all heads collectively attend to the full sequence.

Given a stride size of s_h, for the h-th head, its attention value between decoder query $\mathbf{q}_j$ (at step j) and encoder key vector $\mathbf{k}_i$ (for the i-th input token) can be formulated as:

$$a_{ji}^h = \begin{cases} \text{softmax}(\mathbf{q}_j \mathbf{k}_i), & \text{if } (i-h) \bmod s_h = 0 \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

In HEPOS attention, each query token attends to n/s_h tokens per head, yielding a memory complexity of $\mathcal{O}(mn/s_h)$, where m is the output length.

For **comparison**, Linformer (§ 2.2) can be straightforwardly adapted for encoder-decoder attentions by using decoder queries for attention calculation instead. We do not adapt pattern-based attentions (§ 2.1 and § 2.3), since they rely on local token grouping which makes it difficult to pinpoint salient content.

4 GovReport Dataset

We introduce a new large-scale dataset, GovReport, containing $19,466$ long reports published by U.S. Government Accountability Office (GAO)[4] to fulfill requests by congressional members, and Congressional Research Service (CRS)[5], covering researches on a broad range of national policy issues. A human-written summary is provided along with each report. During data collection, we remove boilerplates from crawled files, and keep the section and paragraph structure of the documents and summaries. Additional data cleaning and processing details are in Appendix A.

We obtain $12,228$ GAO reports and $7,238$ CRS reports of high quality evidenced by human inspection of 200 parsed reports. Collected GAO reports and CRS reports have on average 6.9 and 4.6 sections, respectively. We split train, validation and test set by publication date on each dataset, and end up with 17519 training samples, 974 validation documents, and 973 test samples.

Notably, summaries of GAO reports are written by experts, and are often structured into three aspects in order: "`Why GAO did this study`"—motivation and problem(s) under discussion, "`What GAO found`"—findings of the report, and "`What GAO recommends`"—suggestions and solutions to the problem(s). All but three GAO summaries include "What GAO Found". The percentages of GAO summaries that contain "Why GAO did this study" and "What GAO recommends" are 94.8% and 29.0%. For comparison, structured summaries are also observed on PubMed (Cohan et al., 2018) samples. Though they do not contain explicit aspect labels, the summaries can often be broken down into "Introduction", "Methods", "Results", and "Conclusion" via keyword matching. Details about keyword choices for each aspect are provided in Table 11 in Appendix D.

Comparison with Existing Long Document Summarization Datasets. In Table 2, we compare GovReport with several existing long document summarization datasets, including PubMed and arXiv (Cohan et al., 2018) that consist of scientific publications; BillSum (Kornilova and Eidelman, 2019), a collection of congressional bills; and BigPatent (Sharma et al., 2019), a corpus of U.S. patent documents.

Dataset	# Doc	Summary # word	# sent	Doc # word	Comp.	Den.
PubMed	133,215	202.4	6.8	3049.0	16.2	5.8
arXiv	215,913	272.7	9.6	6029.9	39.8	3.8
BillSum	23,455	207.7	7.2	1813.0	13.6	4.1
BigPatent	1,341,362	116.5	3.7	3573.2	36.3	2.4
GovReport	19,466	**553.4**	**17.8**	**9409.4**	19.0	7.3

Table 2: Statistics of GovReport and existing long document summarization datasets. **Comp.**: compression ratio, **Den.**: extractive fragment density (Grusky et al., 2018). All values are mean over the whole dataset except for the "# Doc" column. Documents and summaries in GovReport are significantly longer.

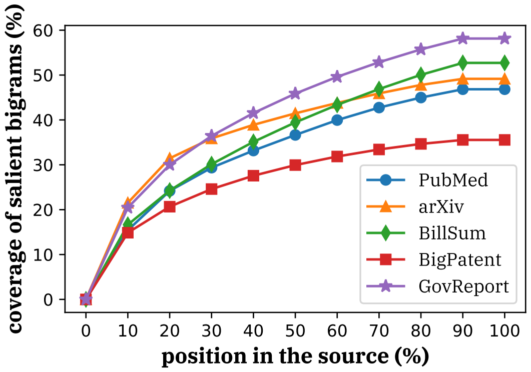

Figure 2: Percentage of unique salient bigrams accumulated from the start to X% of the source. Key information is spread over the documents in GovReport, highlighting the importance of understanding longer text.

First, *documents and summaries in GovReport are significantly longer than prior datasets*. Next, we inspect the distribution of summary-worthy bigrams in the source by dividing each document into ten equisized partitions. For each partition, we count the occurrence of unique bigrams that also appear in the reference, accumulated from the start of the document to the end of the partition. Fig. 2 shows that *key information is spread throughout documents in* GovReport, with new salient bigrams being steadily added as more content is consumed. For arXiv and BigPatent, only about 10% of new salient bigrams are accumulated in the second half of the documents, reflecting the heavy positional bias in these two datasets. In contrast, in GovReport and BillSum, more than 18% of new summary-worthy bigrams appear in the later half of the articles, showing a more even distribution. A similar trend is observed on unigrams. However, BillSum has the shortest documents among the five datasets.

[4] `www.gao.gov`

[5] `crsreports.congress.gov`

5 Summary Evaluation with Cloze QA

This work aims to evaluate whether processing more text improves both informativeness and faithfulness of abstractive summaries. In addition to ROUGE (Lin, 2004) and human evaluation, we extend existing QA-based metric (Eyal et al., 2019) and consider an entailment-based scorer.

QA-based Evaluation. We present a new faithfulness evaluation metric by extending the **APES** score (Eyal et al., 2019). We follow APES to construct a set of **cloze questions**, $\{q\}$, from each reference summary by masking entities. Events, dates, and numbers are also masked, as they are prevalent in our data. Each masked phrase becomes the gold-standard answer a_{ref} for a question q. We do not generate natural language questions (Durmus et al., 2020; Wang et al., 2020a), due to the lack of accurate question generation models for the domains of government reports and scientific papers.

QA models are trained by reading a question and a **context** to label the answer span in the context. We construct context by greedily selecting sentences that maximize the improvement of ROUGE-2 recall when compared with the reference summary. If the answer a_{ref} cannot be found in the context, the sample is excluded from training. We train all QA models by fine-tuning BERT (Devlin et al., 2019) to predict the answer span.

To evaluate the faithfulness of a system summary, **APES** uses the QA model to read the summary and a question q to label an answer a_{sys}. It calculates a unigram F1 score by *comparing* a_{sys} *and* a_{ref}. Different from APES, we further use the QA model to read the context (sentences selected from the source) and give an answer a_{cxt} to the question q. We compute a unigram F1 by *comparing* a_{sys} *and* a_{cxt}, denoted as **APES**$_{src}$. Given that existing summarization models rarely rewrite names or numbers correctly, our metric can better capture faithfulness by using a gold-standard answer constructed from the source article than from the human-written abstract.

To **extract entities and events**, we deploy a state-of-the-art IE framework, OneIE (Lin et al., 2020) on GovReport. On PubMed, we retrain OneIE on Genia 2011 (BioNLP, 2011) and 2013 (BioNLP, 2013), and PubMed (Wei et al., 2019) datasets to extract domain-specific entities and events, such as entities of *Gene* and *Disease*. We additionally include numbers and dates extracted by spaCy (Honnibal and Montani, 2017).

Entailment-based Evaluation. We further consider **FactCC** (Kryscinski et al., 2020), which evaluates factual consistency of a system summary by predicting an entailment score between the source and the summary. We reproduce their method on our datasets.

Additional details for implementing the evaluation models and the entity extraction models are given in Appendix B.

6 Experimental Results

In this section, we start with describing training details in § 6.1. We then compare attention variants on documents of the same length (§ 6.2) and study whether reading more text can generate more informative summaries (§ 6.3). We further report human evaluation on summary informativeness and faithfulness as well as automatic faithfulness scores (§ 6.4). Finally, we investigate whether automatic metrics correlate with human judgment (§ 6.5).

6.1 Training Details

We fine-tune BART (Lewis et al., 2020) for all experiments. We implement our models with PyTorch (Paszke et al., 2019) and Fairseq (Ott et al., 2019). Additional position embeddings are initialized randomly for models that handle longer inputs. The learning rate is set to 1×10^{-4} and learning rate warm-up is applied for the first 10,000 steps. Adafactor (Shazeer and Stern, 2018) optimizer with a gradient clipping of 0.1 is used. All models are trained on two Quadro RTX 6000 GPUs with 24GB memory or one Quadro RTX 8000 with 48GB memory. We set a batch size of 2 per step and accumulate gradient every 32 steps. During test, we adopt a beam size of 4 and a length penalty of 2 (Wu et al., 2016) on all datasets.

6.2 Comparing Attention Variants

Comparisons. We first experiment with articles that are all truncated at 1024 tokens. For encoder attentions, we consider the following variants: (1) sliding WINDOW; (2) adaptive span (ADASPAN); (3) GLOBAL tokens; (4) STRIDE; (5) RANDOM tokens; (6) Linformer (LIN.); (7) locality sensitive hashing (LSH); and (8) SINKHORN. We ensure models are comparable by setting hyperparameters to satisfy $w = \hat{w} = k = lb_l = 2b_s = 256$, so that models have similar memory complexity. For LSH attentions, we select $l = 4$ rounds of hashing. Following prior work (Zaheer et al.,

System	GovReport (new)			PubMed		
	R-1	R-2	R-L	R-1	R-2	R-L
FULL	52.83	20.50	50.14	45.36	18.74	40.26
Encoder variants w/ full enc-dec attn.						
I. Fixed Patterns						
Window	50.78	18.59	48.10	42.74	16.83	37.96
+ Global	51.24	19.01	48.58	43.44	17.07	38.55
+ Stride	51.53	19.14	48.68	43.73	17.25	38.82
+ Random	51.49	18.90	48.75	43.38	16.87	38.45
AdaSpan	50.76	18.69	48.13	43.42	17.16	38.60
+ Global	50.33	18.56	47.80	43.24	17.01	38.42
+ Stride	51.56	19.19	48.57	43.71	17.25	38.76
+ Random	51.39	18.89	48.74	43.28	16.87	38.45
II. Low-Rank Methods						
Lin.	50.70	18.48	47.85	43.65	17.12	38.71
III. Learnable Patterns						
LSH	51.95	19.36	48.85	44.74	18.07	39.76
Sinkhorn	53.00*	20.05*	50.25*	45.10	18.40*	40.11*
Enc-dec variants w/ full encoder attn.						
Lin.	47.79	14.93	45.15	45.16	17.66	40.25
Hepos (ours)	51.05*	19.44*	48.51*	45.80*	18.61*	40.69*
Enc-dec variants w/ Sinkhorn encoder attn.						
Lin.	42.90	12.86	40.32	44.84	17.65	39.98
Hepos (ours)	51.34*	19.09*	48.73*	44.85	18.19*	39.91

Table 3: Results on evaluating encoder and encoder-decoder attentions on input of the same length. Best ROUGE scores of *fixed patterns*, *learnable patterns*, and *enc-dec attentions* are in red, orange, and purple, respectively. $*$: significantly better than comparison(s) using the same encoder or enc-dec attention (approximation randomization test, $p < 0.0005$).

System (MaxLen)	GovReport			PubMed		
	R-1	R-2	R-L	R-1	R-2	R-L
Baselines						
PEGASUS (1024)	–	–	–	45.97	20.15	41.34
TLM (full)	–	–	–	42.13	16.27	39.21
SEAL (full)	–	–	–	46.50	20.10	42.20
DANCER (full)	–	–	–	46.34	19.97	42.42
BigBird (3072)	–	–	–	46.32	20.65	42.33
Encoder variants w/ full enc-dec attn.						
Full (1024)	52.83	20.50	50.14	45.36	18.74	40.26
Stride (4096)	54.29	20.80	51.35	46.95	19.98	41.67
Lin. (3072)	44.84	13.87	41.94	43.69	16.35	38.66
LSH (4096)	54.75	21.36	51.27	47.54	20.79	42.22
Sinkhorn (5120)	55.45	21.45	52.48	47.96	20.78	42.53
Encoder variants w/ Hepos enc-dec attn. (ours)						
LSH (7168)	55.00	21.13	51.67	**48.12**	**21.06**	**42.72**
Sinkhorn (10240)	**56.86**	**22.62**	**53.82**	47.93	20.74	42.58

Table 4: ROUGE scores for models trained on the same GPU. Sinkhorn with Hepos enc-dec attention and LSH with Hepos both read more text and obtain significantly better scores than other models on GovReport and PubMed ($p < 0.0005$).

System (MaxLen)	R-1	R-2	R-L
Baselines			
PEGASUS (1024)	44.21	16.95	38.83
TLM (full)	41.62	14.69	38.03
SEAL (full)	44.3	18.0	39.3
DANCER (full)	45.01	17.60	40.56
BigBird (3072)	46.63	19.02	41.77
Encoder variants w/ Hepos enc-dec attn. (ours)			
LSH (7168)	**48.24**	**20.26**	**41.78**
Sinkhorn (10240)	47.87	20.00	41.50

Table 5: Automatic evaluation on arXiv. Our best model yields better ROUGE scores than previous state-of-the-art models.

2020), we combine Global, Stride, and Random with Window and AdaSpan, where we set $g = n^2/s = r = 128$ for a fair comparison. We adapt Linformer to encoder-decoder attentions to compare with Hepos, where we use $s_h = n/k = 4$ for all experiments. Finally, we report results using FULL, i.e., the original, encoder and encoder-decoder attentions.

Results. *Among all **encoder** variants, learnable patterns perform the best, approaching the performance of full attentions* on both GovReport and PubMed, as shown in Table 3. Within learnable patterns, Sinkhorn attention consistently obtains better ROUGE scores. Moreover, combining techniques in fixed patterns is more effective than simply using window-based sparse attentions, though with an increased memory cost.

For ***encoder-decoder*** *attentions,* Hepos *consistently yields higher ROUGE scores than Linformer on both datasets*, using either full or Sinkhorn encoder. Notably, coupled with a Sinkhorn attention, our model's performance matches the variant using full encoder attention, implying the effectiveness of Hepos on both identifying the salient content and capturing the global context.

6.3 Reading More Input Boosts Informativeness

We investigate whether processing more words generates more informative summaries.

Comparisons include recent top-performing *abstractive* models: PEGASUS (Zhang et al., 2019), a large pre-trained summarization model with truncated inputs; TLM (Pilault et al., 2020), DANCER (Gidiotis and Tsoumakas, 2020), and SEAL (Zhao et al., 2020), all of them using hybrid extract-then-abstract methods; and BigBird (Zaheer et al., 2020), which combines sliding window, global and random token attentions in the encoder.

For encoder variants, we pick the best performing model from fixed patterns to be combined with full encoder-decoder attention, i.e., sliding window with stride (STRIDE), low-rank method (LIN.), and learnable patterns (LSH and SINKHORM). We then combine learnable patterns with HEPOS to support processing more text. All models consume as long an input as the memory allows.

Results. Overall, *models that read more text obtain higher ROUGE scores*, according to results on GovReport and PubMed in Table 4. First, different encoder variants with full encoder-decoder attentions attain better results than the full attentions baseline except Linformer. Second, *adding* HEPOS *encoder-decoder attention almost doubles the words that can be processed and further improves the performance.* This highlights the importance of handling both encoder attentions and encoder-decoder attentions efficiently. Notably, HEPOS with an LSH encoder *achieves new state-of-the-art results on PubMed*, outperforming BigBird which only uses sparse attentions on the encoder. We also report performances of our two best models with HEPOS on arXiv in Table 5, and they outperform all competitive abstractive models.

As can be seen from the sample summaries in Fig. 3, our model that reads in 10k tokens generates more informative summary than the full attention model that only processes 1k tokens. Fig. 4 further shows that ROUGE-2 scores can be consistently lifted when reading more input, with similar trends observed on ROUGE-1 and ROUGE-L. More sample outputs are presented in Appendix C.

6.4 Reading More Input Improves Faithfulness

Here we first show **human evaluation** results on informativeness and unfaithful errors in the generated summaries. We sample 100 documents from GovReport and PubMed (50 each) with structured references that are labeled with aspects as described in § 4 and Appendix D. Each sample is evaluated by two fluent English speakers, who have cumulatively annotated tens of thousands of sentences for the same tasks before this work. Annotators are asked to label each summary sentence with an aspect and then decide whether it contains any type of error. Three types of unfaithful errors are considered: (i) **hallucination**—fabricating content not present in the input, (ii) **deletion**—incorrectly deleting crucial entities, events, or clauses, and (iii) **false concatenation**—inappropriately concatenating components from different sentences. 1 is given if any judge determines that a certain type of error exists in the sentence, 0 otherwise.

Human-written Summary:
In fiscal year 2018, Medicaid covered approximately 75 million individuals at an estimated cost of $629 billion, $393 billion of which were federal funds. (...)
While CMS is generally required to disallow, or recoup, federal funds from states for eligibility-related improper payments if the state's eligibility error rate exceeds 3 percent, it has not done so for decades, because **the method it used for calculating eligibility error rates was found to be insufficient for that purpose.** To address this, in July 2017, CMS **issued revised procedures through which it can recoup funds for eligibility errors**, beginning in fiscal year 2022. (...)

Model w/ full attn.:
Medicaid is a federal-state program that provides health care coverage to low-income individuals and families. (...)
CMS officials stated that **they have provided states with guidance** on how to use data from SSA's automated system for eligibility determinations, (...)
CMS officials said that **they did not have guidance** on when states should use SSA data to evaluate eligibility based on nonfinancial or financial criteria. (...)

Model w/ HEPOS enc-dec attn. (ours):
The Patient Protection and Affordable Care Act (PPACA) expanded Medicaid coverage to millions of low-income adults and children with disabilities and their eligible dependents. (...)
The selected states also reported that **they did not have adequate processes to address these issues.** CMS has taken steps to **improve its oversight of the Medicaid program, including issuing guidance to states** on the use of MAGI-exempt bases for determining eligibility, but these efforts have not been fully implemented. (...)

Figure 3: Sample summaries for a government report. The model with truncated input generates **unfaithful content**. HEPOS attention with a Sinkhorn encoder covers **more salient information**.

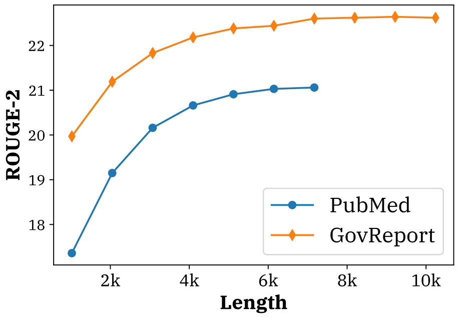

Figure 4: Summarizing articles truncated at different lengths by the best models: LSH (7168)+HEPOS on PubMed and SINKHORN (10240)+HEPOS on GovReport. Reading more consistently improves ROUGE-2.

After reading the full summaries, each judge also scores aspect-level **informativeness**—whether the

System (MaxLen)	Inf.↑	Hal.↓	Del.↓	Concat.↓
GovReport				
Encoder variants w/ full enc-dec attn.				
FULL (1024)	3.29	15.2%	3.5%	9.5%
SINKHORN (5120)	3.32	**11.0%**	**2.3%**	9.4%
Encoder variants w/ HEPOS enc-dec attn. (ours)				
SINKHORN (10240)	**3.53**	11.5%	3.4%	**8.8%**
PubMed				
Encoder variants w/ full enc-dec attn.				
FULL (1024)	3.27	20.1%	2.8%	14.3%
SINKHORN (5120)	3.94	4.8%	**1.6%**	9.6%
Encoder variants w/ HEPOS enc-dec attn. (ours)				
SINKHORN (10240)	**4.18**	**3.5%**	2.2%	**9.1%**

Table 6: Human evaluation on informativeness (Inf.) (1-to-5), and percentages of unfaithful errors due to hallucination (Hal.), deletion (Del.), and false concatenation (Concat.). Inter-rater agreement with Krippendorf's α for all columns: 0.59, 0.59, 0.53 and 0.60.

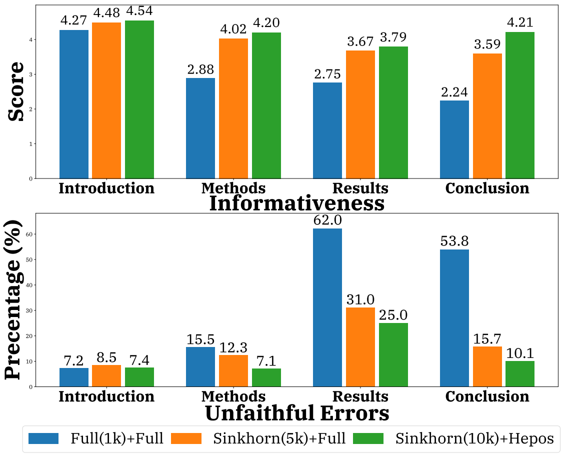

Figure 5: Aspect-level informativeness and percentages of sentences containing unfaithful errors as labeled by both human judges on PubMed. Models with efficient attentions reduce errors for later sections in the sources, e.g., "Results" and "Conclusion".

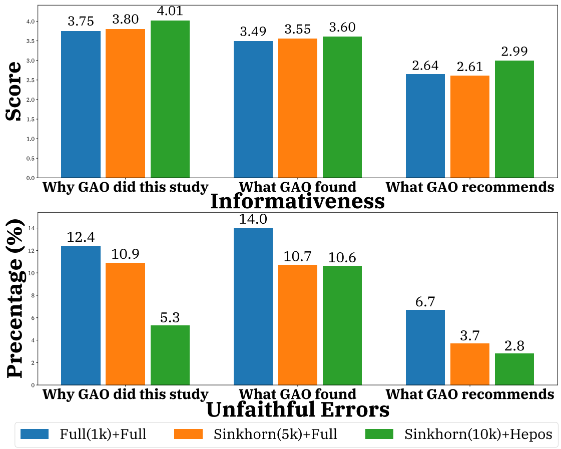

Figure 6: Aspect-level informativeness and percentages of sentences with unfaithful errors on GovReport.

summary covers important information of an aspect when compared with the reference. All system summaries and references are presented in a random order. Human evaluation guidelines and sample summaries for different aspects are included in Appendix D.

Results. Overall, *reading more text significantly improves informativeness as well as reduces fabricated content.* From Table 6, we observe that HEPOS attention, combined with a SINKHORN encoder, obtains better informativeness scores than comparisons that read in less text on both datasets. This echos results from automatic evaluation in the previous section. Moreover, both models that use efficient attentions reduce unfaithfulness, especially hallucination errors, when compared with the full attention model, which only reads 1024 tokens. As the models read more content, they learn to surface more factual and richer content in the summaries, as seen in Fig. 3.

Next, we explore if reading more helps correctly reflect the content in documents' later sections. We plot aspect-level human ratings of informativeness and unfaithful errors on PubMed and GovReport in Fig. 5 and Fig. 6. We report percentages of sentences with unfaithful errors by majority voting (i.e., at least one error is found by both annotators in the sentence). As can be seen, our models consistently improve informativeness and reduce errors across sections, especially for "Results" and "Conclusions" on PubMed and "What GAO recommends" on GovReport—these sections often appear in the later part of the source documents. Especially, we find that the full attention model tends to produce fabricated numbers in resultant summaries, whereas our models are able to correct them.

Lastly, we report the entailment-based FactCC and QA scores APES and APES$_{src}$ for top performing models in Table 7. The results again show that *consuming longer input leads to more faithful summaries*, though the differences are less pronounced.

6.5 Correlations between Human and Automatic Metrics

Finally, we study whether the faithfulness evaluation metrics correlate with human judgment. As shown in Table 8, on both government reports and scientific papers, *QA metrics are better correlated with human ratings, with our newly pro-*

	GovReport			PubMed		
System (MaxLen)	F.	APES	APES$_{src}$	F.	APES	APES$_{src}$
FULL (1024)	58.9	42.7	42.7	**74.6**	43.2	31.5
Encoder variants w/ full enc-dec attn.						
STRIDE (4096)	55.3	43.1	42.5	72.7	43.8	31.9
LIN. (3072)	48.4	35.7	36.3	67.7	39.3	29.5
LSH (4096)	55.7	**44.0**	43.6	73.2	46.7	35.1
SINKHORN (5120)	57.0	43.6	42.1	72.9	46.8	35.4
Encoder variants w/ HEPOS enc-dec attn. (ours)						
LSH (7168)	59.6	**44.0**	44.2	73.3	**47.5**	**35.6**
SINKHORN (10240)	**60.1**	**44.0**	**44.3**	71.9	46.2	34.8

Table 7: Evaluation with FactCC (F.), APES, and the new APES$_{src}$ metric, with higher numbers indicating more faithful summaries.

	GovReport		PubMed	
Metric	Inf.↑	Err.↓	Inf.↑	Err.↓
FactCC	0.07	-0.08	0.10	-0.14
APES	0.16	-0.15	0.25	-0.31
APES$_{src}$	**0.21**	**-0.23**∗	**0.32**∗	**-0.32**

Table 8: Pearson correlation between human ratings and metrics. We use aggregated unfaithful errors (Err.). ∗: significantly better than other metrics based on William's test (Williams, 1959) ($p < 0.05$).

posed APES$_{src}$ *being the stronger of the two.* After inspection, we find that human-written summaries contain paraphrases or acronyms that APES cannot capture via strict lexical matching. For instance, for the question "Diabetes may worsen ___ in patients", the reference answer is "death rate", whereas answers from the source and the system summary are both "mortality". APES$_{src}$ captures this, but not APES.

7 Additional Related Work

Summarizing long inputs has been investigated in many domains, including books (Mihalcea and Ceylan, 2007), patents (Trappey et al., 2009), movie scripts (Gorinski and Lapata, 2015), and scientific publications (Qazvinian and Radev, 2008). However, the datasets are often too small to train neural models. Cohan et al. (2018) publish two large-scale datasets by collecting articles from ARXIV and PUBMED. Popular methods rely on extractive summarizers that identify salient sentences based on positional information (Dong et al., 2020) or combined global and local contexts (Xiao and Carenini, 2019), where each sentence is represented as aggregated word embeddings. However, extractive summaries are often redundant and incoherent, highlighting the need for handling long documents via abstractive summarization.

To that end, extract-then-abstract methods are proposed. For example, Pilault et al. (2020) first extract relevant sentences and then rewrite them into paper abstracts. Our work is in line with building end-to-end abstractive summarization models for long input. Cohan et al. (2018) design a hierarchical encoder to read different sections separately, and then use combined attentions over words and sections to generate the summary. Multiple agents are created to read segments separately, and then collaboratively write an abstract (Celikyilmaz et al., 2018). However, both work truncates articles to 2K words. Although efficient encoder attentions have been studied in Zaheer et al. (2020) for abstractive summarization, at most 3K tokens can be consumed by their models. Our HEPOS encoder-decoder attention are able to process more than 10K tokens, significantly improving summary informativeness and faithfulness.

8 Conclusion

We investigate efficient attentions for long document summarization. We propose a novel encoder-decoder attention, HEPOS, based on head-wise positional strides that can effectively identify salient content. Models based on HEPOS attention can process at least twice as many words and produce more informative summaries with less unfaithful errors, according to both automatic evaluation and human evaluation. We further show that our new cloze QA metric better correlates with human judgment than prior faithfulness evaluation metrics.

Acknowledgements

This research is supported in part by Oracle for Research Cloud Credits, National Science Foundation through Grant IIS-1813341, and by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via contract # FA8650-17-C-9116. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein. We thank three anonymous reviewers for their valuable suggestions and comments.

References

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer.

BioNLP. 2011. Genia event extraction (genia).

BioNLP. 2013. Genia event extraction for nfkb knowledge base.

Asli Celikyilmaz, Antoine Bosselut, Xiaodong He, and Yejin Choi. 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)*, pages 1662–1675.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy Colwell, and Adrian Weller. 2020. Rethinking attention with performers.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621, New Orleans, Louisiana. Association for Computational Linguistics.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, Florence, Italy. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Yue Dong, Andrei Romascanu, and Jackie CK Cheung. 2020. Hiporank: Incorporating hierarchical and positional information into graph-based unsupervised long document extractive summarization. *arXiv preprint arXiv:2005.00513*.

Esin Durmus, He He, and Mona Diab. 2020. FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5055–5070, Online. Association for Computational Linguistics.

Matan Eyal, Tal Baumel, and Michael Elhadad. 2019. Question answering as an automatic evaluation metric for news article summarization. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3938–3948, Minneapolis, Minnesota. Association for Computational Linguistics.

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

Alexios Gidiotis and Grigorios Tsoumakas. 2020. A divide-and-conquer approach to the summarization of long documents. *arXiv: Computation and Language*.

Philip John Gorinski and Mirella Lapata. 2015. Movie script summarization as graph-based scene extraction. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1066–1076, Denver, Colorado. Association for Computational Linguistics.

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 708–719, New Orleans, Louisiana. Association for Computational Linguistics.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. 2020. Transformers are rnns: Fast autoregressive transformers with linear attention.

Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. 2020. Reformer: The efficient transformer. In *International Conference on Learning Representations*.

Anastassia Kornilova and Vladimir Eidelman. 2019. BillSum: A corpus for automatic summarization of US legislation. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 48–56, Hong Kong, China. Association for Computational Linguistics.

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online. Association for Computational Linguistics.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. 2019. Set transformer: A framework for attention-based permutation-invariant neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3744–3753, Long Beach, California, USA. PMLR.

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*, pages 7871–7880, Online. Association for Computational Linguistics.

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

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *Proceedings of The 58th Annual Meeting of the Association for Computational Linguistics*.

Peter J. Liu*, Mohammad Saleh*, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. 2018. Generating wikipedia by summarizing long sequences. In *International Conference on Learning Representations*.

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. Association for Computational Linguistics.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online. Association for Computational Linguistics.

Rada Mihalcea and Hakan Ceylan. 2007. Explorations in automatic book summarization. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 380–389, Prague, Czech Republic. Association for Computational Linguistics.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Jonathan Pilault, Raymond Li, Sandeep Subramanian, and Chris Pal. 2020. On extractive and abstractive neural document summarization with transformer language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9308–9319, Online. Association for Computational Linguistics.

Vahed Qazvinian and Dragomir R Radev. 2008. Scientific paper summarization using citation summary networks. *arXiv preprint arXiv:0807.1560*.

Eva Sharma, Chen Li, and Lu Wang. 2019. BIGPATENT: A large-scale dataset for abstractive and coherent summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2204–2213, Florence, Italy. Association for Computational Linguistics.

Noam Shazeer and Mitchell Stern. 2018. Adafactor: Adaptive learning rates with sublinear memory cost. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4596–4604, Stockholmsmässan, Stockholm Sweden. PMLR.

Sainbayar Sukhbaatar, Edouard Grave, Piotr Bojanowski, and Armand Joulin. 2019. Adaptive attention span in transformers. In *Proceedings of the*

57th Annual Meeting of the Association for Computational Linguistics, pages 331–335, Florence, Italy. Association for Computational Linguistics.

Yi Tay, Dara Bahri, Liu Yang, Donald Metzler, and Da-Cheng Juan. 2020a. Sparse sinkhorn attention.

Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. 2020b. Efficient transformers: A survey. *arXiv preprint arXiv:2009.06732*.

Amy JC Trappey, Charles V Trappey, and Chun-Yi Wu. 2009. Automatic patent document summarization for collaborative knowledge systems and services. *Journal of Systems Science and Systems Engineering*, 18(1):71–94.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008. Curran Associates, Inc.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5797–5808, Florence, Italy. Association for Computational Linguistics.

Christopher Walker, Stephanie Strassel, Stephanie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020a. Asking and answering questions to evaluate the factual consistency of summaries. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5020, Online. Association for Computational Linguistics.

Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Darrin Eide, Kathryn Funk, Rodney Kinney, Ziyang Liu, William Merrill, et al. 2020b. Cord-19: The covid-19 open research dataset. *ArXiv*.

Sinong Wang, Belinda Z. Li, Madian Khabsa, Han Fang, and Hao Ma. 2020c. Linformer: Self-attention with linear complexity.

Chih-Hsuan Wei, Alexis Allot, Robert Leaman, and Zhiyong Lu. 2019. Pubtator central: automated concept annotation for biomedical full text articles. *Nucleic acids research*, 47(W1):W587–W593.

E.J. Williams. 1959. *Regression Analysis*. WILEY SERIES in PROBABILITY and STATISTICS: APPLIED PROBABILITY and STATIST ICS SECTION Series. Wiley.

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. *arXiv preprint arXiv:1609.08144*.

Wen Xiao and Giuseppe Carenini. 2019. Extractive summarization of long documents by combining global and local context. 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 3011–3021, Hong Kong, China. Association for Computational Linguistics.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. 2020. Big bird: Transformers for longer sequences. arxiv e-prints, art. *arXiv preprint arXiv:2007.14062*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2019. PEGASUS: pre-training with extracted gap-sentences for abstractive summarization. *CoRR*, abs/1912.08777.

Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020. Seal: Segment-wise extractive-abstractive long-form text summarization.

A GovReport Dataset Collection and Processing

For GAO reports, their summaries are organized as highlights. We collect GAO reports that include corresponding highlights and were published before Jul 7, 2020 . The reports and highlights are published in PDF files. Most of the highlights are also reorganized and shown on the web page as HTML. Since PDF parsing is more prone to errors than web parsing, we only keep the reports whose highlights can be obtained on the corresponding web page to ensure the quality of extracted gold-standard summaries. For reports, we first convert the PDF files to HTML using PDFMiner[6]. We then parse the HTML into text into sections and paragraphs with handcrafted parsing rules. We remove the reports that do not have cover pages, as our rules are constructed for documents with then. We further remove parsed documents with empty sections, non-capitalized section titles, or a single section, since these are common patterns of incorrectly parsed documents. Failed parsing would also result in short documents. Therefore, we examine the reports with shorter length and then filter out 10% of the shortest reports.

[6] https://github.com/euske/pdfminer

We collect CRS reports that were published before May 20, 2020 from EveryCRSReport[7] where the original PDF files are already parsed into HTML. We only keep documents with expert-written summaries. We then gather texts from the html files.

B Experiment Details

FactCC Training Data Construction. Kryscinski et al. (2020) generate training data by applying rule-based transformations to sentences from source documents. We leverage reference summaries, where we train a FactCC model by reading a summary sentence (i.e., the claim) and a context to predict the corresponding label. A context is constructed by greedily selecting sentences that maximize the improvement of its ROUGE-2 when compared against the reference summary sentence. Following FactCC, we apply *sentence negation*, *entity swap*, and *number swap* to summary sentences to construct negative claims and use the original sentences as positive claims. During testing, we first find the context for each system summary sentence. The model then predicts a sentence-level faithfulness score by reading the system summary sentence and the context.

Evaluation Model Training. We fine-tune BERT (Devlin et al., 2019) for both FactCC and QA models. We include an additional classification head to predict entailment label or answer spans based on the [CLS] token. For GovReport dataset, we consider a base version of BERT with uncased tokens. For PubMed, we use a BERT model which is fine-tuned on PubMed abstracts to obtain better performance[8].

Entity Extraction Model. We use OneIE to extract entities from the reference summary (Lin et al., 2020). OneIE is a unified framework that combines entities, relations, and events extraction in one model. The model leverages the BERT pre-trained weights as the sentence embedding to produce entities, relations, and events from a sentence. Two OneIE models are built.

	Genia 2011	Genia 2013	PubMed
Entity Type			
Anaphora	-	105	-
Entity	480	121	-
CellLine	-	-	614
Chemical	-	-	14,051
Disease	-	-	62,228
Mutation	-	-	164
Protein	11,539	3,562	15,577
Species	-	-	52,954
Event Type			
Binding	880	167	-
Gene Expression	2,076	666	-
Localization	264	44	-
Negative Regulation	338	273	-
Phosphorylation	175	105	-
Positive Regulation	1,123	311	-
Protein Catabolism	100	23	-
Protein Modification	-	8	-
Regulation	292	72	-
Transcription	580	97	-
Ubiquitination	-	4	-

Table 9: Dataset description for training OneIE Biomedical extraction. While Genia 2011 and 2013 datasets focus more on event extraction, PubMed covers more entities.

The first model for government reports is trained on the Automatic Content Extraction (ACE) 2005 dataset (Walker et al., 2006). This model can extract entities from general conversation contexts such as *People*, *Location*, or *Organization*, and events such as *Movement*, *Conflict*, or *Justice*, etc.

The second model for scientific domain information extraction is trained on the Genia 2011 (BioNLP, 2011), Genia 2013 (BioNLP, 2013), and PubMed (Wei et al., 2019) datasets. It extracts entity such as *Gene*, *Variant*, *Disease*, *Chemical*, or *Species*, and events such as *Gene Expression*, *Binding*, *Protein Modification*, or *Positive Regulation*, etc. The full list of entity and event types can be found in Table 9. To train this model, we fine-tune the BioBERT pre-trained model (Lee et al., 2020) on the COVID-19 Open Research (CORD-19) dataset (Wang et al., 2020b). As we proposed, this model is applied to the PubMed data.

C Additional Sample Outputs

We include two samples from GovReport and PubMed to further illustrate that our model with HEPOS attention generates more faithful and informative summaries in Fig. 7 and Fig. 8.

D Human Evaluation Guideline

In human evaluation, annotators are asked to evaluate the system summaries generated for a report or a paper. In addition to the summaries, annotators are provided with *the report or the paper to be summarized* and a corresponding *human-written*

[7] https://www.everycrsreport.com

[8] https://huggingface.co/monologg/biobert_v1.0_pubmed_pmc

reference. Human judges evaluate each system summary **sentence by sentence**. The annotation consists of three tasks, which are described below.

Task 1: Aspect Labeling. First, annotators are asked to decide which **aspect** each sentence belongs to. For government reports, each sentence should be categorized into three aspects: (1) Why GAO did this study, (2) What GAO found, and (3) What GAO recommends. For scientific papers, summaries have four aspects: (1) Introduction and Literature, (2) Methods, (3) Results, and (4) Discussion and Conclusion. Table 10 and Table 11 contain example reference summaries with labeled aspects.

Task 2: Sentence-level Faithfulness Error Labeling. Next, annotators will judge whether each sentence contains any **unfaithful content**. Unfaithful content is categorized into three types. A "0" or "1" label will be given to each type, where "0" indicates the sentence is free of such type of error, and "1" otherwise.

Concretely, unfaithful content is the fabricated or contradictory content which *is not present or contradicts the facts* in the source article. It can also be *ambiguous expression* which distorts the meaning. Here are detailed descriptions for the three types of errors:

- **Hallucination** error refers to fabricated content that cannot be found or inferred from the source.
- Misconstruction error that is due to **deletion** of entities, events, or clauses, resulting in sentences that are incomplete, missing context, or ungrammatical.
- Misconstruction error that is caused by **false concatenation** of content from different places in the source.

Task 3: Aspect-level Summary Quality Rating. After reading the full summary, annotators will evaluate the **informativeness** of the summary for each aspect— whether the summary provides *a necessary and enough coverage of information* in the *reference*. For instance, whether the summary covers all the salient points in "Why GAO did this study".

Here are detailed descriptions of informativeness:

- **5**: Summary covers enough key points in the reference (only misses minor topics), and is free of unfaithful errors.
- **4**: Summary covers major key points (e.g., 80 percent) and may miss one or two key points in the reference. Summary can contain one unfaithful error.
- **3**: Summary covers roughly half of the key points in the reference or contains 2 or 3 unfaithful errors.
- **2**: Summary only covers 1 or 2 key points and misses many important topics (e.g. > 80 percent) in the reference, or contains more than 3 major unfaithful errors, e.g. summary fabricates or distorts some facts.
- **1**: Summary is irrelevant and does not cover any content in the reference.

Summary by Human:

What GAO found

The Food and Drug Administration (FDA) and U.S. Department of Agriculture (USDA) have begun collaborating on regulatory oversight of cell-cultured meat. For example, in 2019, the agencies signed an interagency agreement and created three working groups to carry out the terms of the agreement. However, the agreement and working groups could more fully incorporate practices to enhance and sustain collaboration, such as defining outcomes. For example, the agreement identifies the development of labeling principles as an outcome, but does not describe how the agencies will track and monitor progress toward this outcome, and the working groups identify a lead agency but not members' roles. (...)

Developing and updating written guidance and agreements is also a leading practice for interagency collaboration. By fully incorporating leading practices into their efforts to collaborate, the agencies could minimize potential overlap and fragmentation, use resources in a more efficient manner, and better ensure the public and other key stakeholders have clarity about the agencies' oversight responsibilities. (...)

Model w/ full attn.:

What GAO found

The frequency of inspections the agencies conduct varies, as follows: FDA. FDA conducts risk-based inspections of high- risk and non-high-risk food facilities. FDA's authority requires a risk- based approach, in which inspection rates vary depending on the level of risk associated with a food product. (...)

In 2010, FDA began conducting inspections of meat slaughter facilities to determine whether they were complying with the agency's longstanding interpretation of its statutes requiring it to do so. In addition,USDA conducted inspections of slaughter facilities that process catfish to determine if they are complying with USDA's requirement to inspect catfish slaughter facilities. USDA also conducts inspections of processing facilities to verify that they are following USDA's requirements to inspect these facilities. In 2011, USDA began inspecting processing facilities for catfish for the first time. (...)

Model w/ HEPOS enc-dec attn.:

What GAO found

The Departments of Health and Human Services (HHS) and Agriculture have taken several steps to coordinate their oversight of the development, commercialization, and regulation of cell cultured meat and seafood, but they have not fully incorporated leading practices for interagency collaboration. For example, HHS and USDA have established interagency working groups that have met to discuss issues such as coordination and coordination. However, the agencies have not yet fully incorporated some leading practices, such as establishing clear roles and responsibilities, into their interagency agreements. (...)

Figure 7: Sample summaries for a government report. Model with truncated input generates unfaithful content. Our HEPOS encoder-decoder attention with Sinkhorn encoder attention covers more salient information in "What GAO found" aspect.

Original Paper:

(...) anatomical abnormalities included complex pelvicalyceal anatomy in 98, pelviureteric junction obstruction (pujo) in 35, ectopic, horseshoe, cross-fused ectopic kidneys in 40, and duplex system with nonfunctioning upper or lower moiety in 14 patients. (...)

of the 1404 patients from rural area , 547 (39%) were managed by open surgery as compared to 362 (22%) of the patients from urban areas. (...)

the mean size of the kidney stone was 5.05 5.88 cm, ureter 1.95 1.33 cm, and bladder 7.6 3.2 cm. (...)

overall success rate was 90% in pcnl, 96% in urs, 100% in cystolithotripsy, and 81% in swl.

Summary by Human:

Results

Results of 3969 surgeries, 2794 (70%) were minimally invasive surgery (mis) techniques to include swl 19%, pcnl 16 % , urs 18.9% , and pucl + pccl 16% and 1175 (30%) were open surgeries .

the main factors necessitating open surgery were large stone burden 37%, anatomical abnormalities 16%, stones with renal failure 34%, gross hydronephrosis with thin cortex 58% , urinary tract infection (uti) 25%, and failed mis 18%. nearly 50% of the surgeries were necessitated by economic constraints and long distance from center where one-time treatment was preferred by the patient. tone-free rates by open surgeries were pyelolithotomy 91%, ureterolithotomy 100%, and cystolithotomy 100% with complication rate of up to 3%.

Model w/ full attn.:

Results

results: the mean age of the patients was 6.5 years. the mean follow-up period was 12.5 months. mean stone size was 1.5cm. the majority of the stones were lager stones (n=454). the stone free rate was 100%. open surgery was performed in children with large bulk of stones, anatomical abnormalities, marked obstructive cortical atrophy and scarring, gross hydronephrosis, or urinary tract infection (uti).

Model w/ HEPOS enc-dec attn.:

Results

the mean age of the patients was 6.35, 3.7 years with a male-to-female ratio of 2.8:1, and a range of 25 days (<1 month) to 15 years. anatomical abnormalities included complex pelvicalyceal anatomy in 98, pelviureteric junction obstruction (pujo) in 35, ectopic, horseshoe, cross-fused ectopic kidneys in 40, and duplex system with nonfunctioning upper or lower moiety in 14 patients. the mean size of the kidney stone was 5.05 5.88 cm3. of the 1404 patients from rural areas, 547 (39%) were managed by surgery as compared to 362 (22%) patients from urban areas. overall success rate was 90% in pcnl , 96% in urs , 100% in cystolithotripsy , and 81% in swl.

Figure 8: Sample summaries for a scientific paper. Model with truncated input generates fabricated facts. Our HEPOS encoder-decoder attention with LSH encoder attention are more faithful for the aspect of "results".

Aspect	**Example**
Why GAO Did This Study	To protect data that are shared with state government agencies, federal agencies have established cybersecurity requirements and related compliance assessment programs. Specifically, they have numerous cybersecurity requirements for states to follow when accessing, storing, and transmitting federal data. GAO was asked to evaluate federal agencies' cybersecurity requirements and related assessment programs for state agencies. The objectives were to determine the extent to which (...)
What GAO Found	Although the Centers for Medicare and Medicaid Services (CMS), Federal Bureau of Investigation (FBI), Internal Revenue Service (IRS), and Social Security Administration (SSA) each established requirements to secure data that states receive, these requirements often had conflicting parameters. Such parameters involve agencies defining specific values like the number of consecutive unsuccessful logon attempts prior to locking out the user. Among the four federal agencies, the percentage of total requirements with conflicting parameters ranged from 49 percent to 79 percent. Regarding variance with National Institute of Standards and Technology guidance, GAO found that the extent to which the four agencies did not fully address guidance varied from 9 percent to 53 percent of total requirements. The variances were due in part to the federal agencies' insufficient coordination in establishing requirements. (...)
What GAO Recommends	GAO is making 12 recommendations to the four selected agencies and to OMB. Three agencies agreed with the recommendations and one agency (IRS) partially agreed or disagreed with them. OMB did not provide comments. GAO continues to believe all recommendations are warranted.

Table 10: Sample reference summary with aspects in a GAO report.

Aspect	**Keywords**	**Example**
Introduction and Literature	introduction, case, objectives, purposes, objective, purpose, background, literature, related work	background : the present study was carried out to assess the effects of community nutrition intervention based on advocacy approach on malnutrition status among school - aged children in shiraz , iran . introduction . low serum vitamin d levels are associated with increased postural sway . vitamin d varies seasonally . this study investigates whether postural sway varies seasonally and is associated with serum vitamin d and falls .
Methods	materials and methods, techniques, methodology, materials, research design, study design	materials and methods : this case - control nutritional intervention has been done between 2008 and 2009 on 2897 primary and secondary school boys and girls (7 - 13 years old) based on advocacy approach in shiraz , iran . the project provided nutritious snacks in public schools over a 2 - year period along with advocacy oriented actions in order to implement and promote nutritional intervention . for evaluation of effectiveness of the intervention growth monitoring indices of pre- and post - intervention were statistically compared .
Results	results, experiments, observations	results : the frequency of subjects with body mass index lower than 5% decreased significantly after intervention among girls (p = 0. 02) . however , there were no significant changes among boys or total population . (...)
Discussion and Conlusion	discussion, limitation, conclusions, concluding	conclusion : this study demonstrates the potential success and scalability of school feeding programs in iran . community nutrition intervention based on the advocacy process model is effective on reducing the prevalence of underweight specifically among female school aged children .

Table 11: Sample reference summary with aspects labeled in a PubMed article. Keywords are used to match different parts of the summaries to the four aspects.