

Hierarchical Learning for Generation with Long Source Sequences

Tobias Rohde^{†‡}, Xiaoxia Wu^{†*}, Yinhan Liu[†]

[†] Birch AI, Seattle, WA

[‡] Paul G. Allen School of Computer Science & Engineering,
University of Washington, Seattle, WA
{tobiasr, xwu, yinhan}@birch.ai

Abstract

One of the challenges for current sequence to sequence (seq2seq) models is processing long sequences, such as those in summarization and document level machine translation tasks. These tasks require the model to reason at the token level as well as the sentence and paragraph level. We design and study a new Hierarchical Attention Transformer-based architecture (HAT) that outperforms standard Transformers on several sequence to sequence tasks. In particular, our model achieves state-of-the-art results on four summarization tasks, including ArXiv, CNN/DM, SAMSum, and AMI, and we push PubMed R1 & R2 SOTA further. Our model significantly outperforms our document-level machine translation baseline by 28 BLEU on the WMT19 EN-DE document translation task. We also investigate what the hierarchical layers learn by visualizing the hierarchical encoder-decoder attention. Finally, we study hierarchical learning on encoder-only pre-training and analyze its performance on classification downstream tasks.¹

1 Introduction

Sequence to sequence (seq2seq) models have been successfully used for a variety of natural language processing (NLP) tasks, including text summarization, machine translation and question answering. Often sequence to sequence models consist of an encoder that processes some source sequence and a decoder that generates the target sequence. Originally, Sutskever et al. (2014) used recurrent neural networks as encoder and decoder for machine translation on the WMT-14 dataset. Bahdanau et al. (2016) introduced the attention mechanism, where the decoder computes a distribution over the hidden states of the encoder and uses it to weigh the

hidden states of the input tokens differently at each decoding step. Vaswani et al. (2017) then introduced a new architecture for sequence to sequence modeling – the Transformer, which is based on the attention mechanism but not recurrent allowing for more efficient training.

While successful, both recurrent neural networks and Transformer-based models have limits on the input sequence length. When the input is long, the learning degrades particularly for tasks which require a comprehensive understanding of the entire paragraph or document. One of the main learning challenges for seq2seq models is that the decoder needs to attend to token level representations from the encoder to predict the next token, while at the same time it must learn from a large context.

A commonly used method for attempting to solve the long-sequence problem is hierarchical attention (Yang et al., 2016). This method was studied primarily on long sequence classification tasks, where the model learns a document representation which is used as the input to a classifier. Since then, many successful papers proposed methods using hierarchical attention (see Section 2 for full details). While hierarchical attention has been successfully applied to classification tasks, its potential for being applied to large document sequence to sequence tasks remains an interesting and open question.

In this paper, we present a hierarchical attention model based on the standard Transformer (Vaswani et al., 2017) that produces sentence level representations, and combine them with token level representations to improve performance on long document sequence to sequence tasks. Our main contributions include

- 1) We design a hierarchical seq2seq attention network architecture named HAT (Hierarchical Attention Transformer) (See Figure 1). We conduct extensive experiments on various generation tasks for our proposed model in Sections 4 and 5 and achieve new state-of-the-art

^{*}Work performed while Xiaoxia Wu interned at Birch AI. She is currently a postdoc fellow at The University of Chicago and Toyota Technological Institute at Chicago.

¹<https://github.com/birch-research/hierarchical-learning>

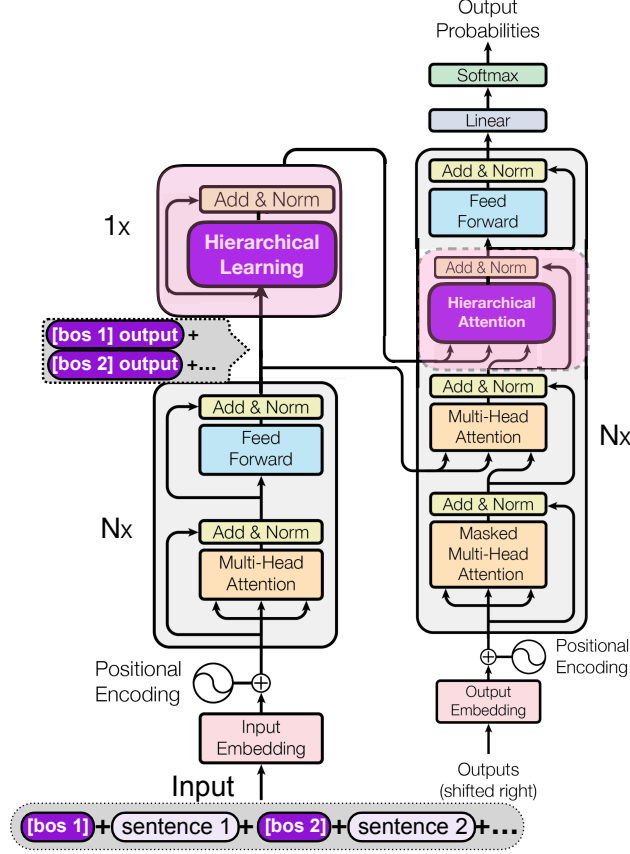


Figure 1: **The architecture of HAT: Hierarchical Attention Transformer.** The blocks shaded in pink are the additional hierarchical layers added to the existing Transformer (Vaswani et al., 2017) architecture. The extra BOS tokens used for hierarchical learning are highlighted in purple. The figure is based on Figure 1 in Vaswani et al. (2017).

results on several datasets.

- 2) In Sections 6 and 7, we study the generated output of our architecture to further understand the benefits of hierarchical attention and compare it with those generated by plain seq2seq Transformer models. Furthermore, we analyze how the decoder makes use of the sentence level representations of the encoder by visualizing the hierarchical encoder-decoder attention.
- 3) Finally, we apply hierarchical attention to an encoder-only architecture and pre-train it on a Books and Wiki corpus similar to the one used in RoBERTa (Liu et al., 2019). We fine-tune our pre-trained model on several downstream tasks and analyze the performance in Section 7.

2 Background

Transformer models. The attention-based Transformer architecture (Vaswani et al., 2017) currently dominates the state-of-the-art performance in many NLP tasks. This is largely due to the success of language model pre-training prior to fine-tuning the Transformer on the actual downstream task of interest. Pre-training has been applied in different ways to different variations of the Transformer architecture, including encoder-only pre-training (BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), XLNet (Yang et al., 2019)), decoder-only pre-training (GPT (Radford et al., 2019)), encoder-decoder pre-training (T5 (Raffel et al., 2020), BART (Lewis et al., 2020)) and multilingual pre-training (MBART (Liu et al., 2020), Roberta-XLM (Conneau et al., 2020)). Both classification and generation downstream task performance improves significantly by initializing parameters from pre-trained models. To use a pre-trained model, the

downstream task model architecture needs to be the same or similar to the one used in pre-training. Transformers have become the most popular architectures in NLP. However, one disadvantage of Transformers is that when the input sequence is long, the performance of the attention mechanism can become worse.

Long document modeling. Understanding and modeling large documents has been a longstanding challenge that has become increasingly demanded in practice (Nakao, 2000; Mihalcea and Ceylan, 2007; Iyer et al., 2016; Zhang et al., 2017; Çelikyilmaz et al., 2018; Paulus et al., 2018; Tasnim et al., 2019; Zhao et al., 2019; Gunel et al., 2019; Ye et al., 2019; Beltagy et al., 2020; Zhang et al., 2020). Many methods have been proposed to improve performance on abstractive summarization tasks, which often have long source sequences. Recent work includes graph-based attention neural networks (Tan et al., 2017), discourse-aware attention models Cohan et al. (2018) and decoder decomposition with policy learning Kryscinski et al. (2018). In addition, Xu et al. (2020a) suggest a discourse-aware neural summarization model with structural discourse graphs to capture the long-range dependencies among discourse units. Zhu et al. (2020) design a hierarchical structure on speaker turns and roles to improve performance on meeting summarization.

Hierarchical learning. Hierarchical learning has been suggested by many researchers and it has been empirically shown to be effective in numerous diverse tasks outside of natural language processing, including policy learning (Ryu et al., 2020), visual relationship detection (Mi and Chen, 2020), part-aware face detection (Wu et al., 2019), visually-aware food recommendation (Gao et al., 2020), urban anomaly prediction (Huang et al., 2020), online banking fraud detection (Achituve et al., 2019) and discourse parsing (Li et al., 2016). Within the natural language processing domain, Yang et al. (2016) propose a hierarchical structure with word and sentence-level attention mechanisms for document classification. Xing et al. (2018) designed a hierarchical recurrent attention network to model the hierarchical structure of context, word importance, and utterance importance. Gao et al. (2018) show that hierarchical attention networks perform significantly better than conventional models for information extraction from cancer pathology reports. Xu et al. (2020b) present a stacked hierarchical at-

tention mechanism for text-based games in deep reinforcement learning. Song et al. (2017) design a hierarchical contextual attention recurrent neural network to capture the global long range contextual dependencies among map query sessions. Zhao et al. (2018) detect events in sentences by constructing a hierarchical and supervised attention-based recurrent neural network. Han et al. (2018) propose a hierarchical attention scheme for distantly supervised relation extraction. (Miculicich et al., 2018) apply hierarchical learning on data for document level machine translation.

In this work, we apply hierarchical learning to Transformer models for improving performance on generation tasks with long documents, including summarization and document-level machine translation.

3 Model

We modify the standard sequence to sequence transformer architecture (Vaswani et al., 2017) by adding hierarchical attention for improved processing of long documents (Figure 1).

We use 12 encoder and decoder layers, a hidden size of 1024, 4096 for the dimension of the fully-connected feed-forward networks and 16 attention heads in both the encoder and the decoder. Unlike the original Transformer, we use GELU activation instead of ReLU.

3.1 Data pre-processing

During pre-processing, we insert BOS tokens at the start of every sentence in each source document as shown in Figure 1. We simply determine sentence boundaries by punctuation or by using prior sentence segmentation present in the documents. By using BOS tokens as hierarchical tokens, the hierarchical attention can benefit from the representations learned for the BOS token during pre-training.

3.2 Encoder

We use the same encoder as in Transformer Vaswani et al. (2017). This produces an embedding for each input token. After those, we add an additional encoder layer (pink block in Figure 1) which only attends to the embeddings of the BOS tokens that we inserted during data-preprocessing. We refer to this layer as the hierarchical encoder layer, which produces another level of contextual representations for each of the BOS tokens, which can be interpreted

as sentence level representations. We find that a single hierarchical layer works the best, although multiple hierarchical layers may be used.

3.3 Decoder

As in Vaswani et al. (2017), each layer first performs self attention over the previously generated tokens and then attends over the outputs of the final token level encoder layer, similarly to the vanilla Transformer. We add an attention module that attends over the BOS token embeddings from the hierarchical encoder layer.

4 Experiments

Our architecture is specifically designed to better handle long sequences, thus we evaluate it on summarization and document level translation tasks, which tend to have long source sequences as the input. We also run experiments with non-generation tasks with an encoder-only hierarchical attention architecture (Section 7).

4.1 Summarization Tasks

We characterize all the summarization tasks into several categories, test our architectures with different weight initializations and compare them with their non-hierarchical counterparts using the same weight initializations.

Long sequence datasets The PubMed and arXiv datasets (Cohan et al., 2018) contain scientific articles from PubMed and arXiv respectively, and use the abstract of the articles as the target summary. The sequences in each of these datasets are long and need to be truncated to be processed by most transformers. Statistics on the PubMed and arXiv datasets are given in Table 1.

We add a BOS token at the beginning of each sentence in the source sequences during data preprocessing. We use 3072 as the maximum source length, 512 as the maximum target length. Longer sequences are truncated. We followed BART (Lewis et al., 2020) and use GPT2 (Radford et al., 2019) byte-pair encoding (BPE). We random initialize the hierarchical encoder layer, the hierarchical attention modules in the decoder layers and the additional positional embedding from 512 to 3072. We initialize all the remaining weights with pre-trained weights from BART. We initialize also initialize a plain seq2seq model with BART pre-trained weights for direct comparison.

For training, we use a batch-size of 128. We set weight decay to 0.01 and use the Adam optimizer with $(\beta_1, \beta_2) = (0.9, 0.999)$ and $\varepsilon = 10^{-8}$ (Kingma and Ba, 2015). We train for 30000 steps and warm up the learning rate for 900 steps to 3×10^{-5} and decay it linearly afterwards. We use a dropout rate of 0.1 for attention and all layers. We also use label smoothing with smoothing constant 0.1. We use mixed precision for training. We complete 30000 steps in approximately 115 hours (without validation and checkpoint saving) on 2 A100 GPUs.

For generation, we use a beam width of 2, length penalty of 1 and minimum and maximum generation lengths of 72 and 966 respectively.

News datasets CNN/DailyMail (Hermann et al., 2015) and XSum (Narayan et al., 2018) are commonly used news summarization datasets. Both datasets are sourced from news articles, which frequently include a short summary in the article. Statistics for CNN/DM and Xsum are in Table 1.

We use 1024 as maximum source length for article and 256 as maximum target length for the summary. Longer sequences are truncated. We apply the same data-processing and model initialization for these two datasets as we did for the long sequence datasets. We train on CNN/DM for 20000 steps with a batch-size of 64 and a peak learning rate of 3×10^{-5} with linear decay afterwards. During generation, we use beam size 1, no length penalty and minimum and maximum generation lengths of 40 and 120, respectively. We use the same training and generation parameters for both the hierarchical seq2seq model and the plain seq2seq model.

Conversational datasets Since conversational summarization datasets are rare, we only consider the SAMSum (Gliwa et al., 2019) corpus. It consists of staged chat conversations between two people and corresponding abstractive summaries written by linguists. Statistics for SAMSum dataset are presented in Table 1.

During data-processing, we concatenate the role string with its utterance and add a BOS token at the beginning of each speaker turn. Then we concatenate all the turns together and use this as the source sequence. We add segmentation embeddings to our model, where we map the same role along with its utterance to the same segment embedding. The input to the encoder layers is the sum of the

Dataset	Instances			Input Length		Output Length	
	Train	Valid	Test	Words	Tokens	Words	Tokens
PubMed	112K	6.6K	6.7K	3016	4016	178	258
ArXiv	203K	6.4K	6.4K	6055	8613	280	362
CNN-DM	287K	13K	11K	781	906	56	63
XSum	204K	11K	11K	431	488	23	27
SAMSum	14.7K	818	819	94	147	20	26
AMI	100	17	20	3156	4081	280	321
ISCI	43	10	6	6228	7913	466	576

Table 1: Stats for each of the summarization datasets. Both number of words and number of BPE tokens are presented. Words are counted before tokenization.

Dataset	Documents			Source Length (EN)		Target Length (DE)	
	Train	Valid	Test	Words	Tokens	Words	Tokens
WMT-19 EN-DE	358K	488	123	366	459	343	474

Table 2: Statistics for WMT-19 EN-DE document translation task. Both average number of words and number of BPE tokens are presented.

token embeddings, position embeddings and segment embeddings. We randomly initialize segment embedding parameters, and initialize the remaining parameters with the hierarchical seq2seq model trained on CNN-DM. For comparison, we also use train a plain seq2seq model initialized with weights from a plain seq2seq trained on CNN-DM.

Meeting datasets The AMI (Carletta et al., 2006) and ISCI (Janin et al., 2003) corpora consist of staged meetings which are transcribed and annotated with abstractive summaries. The meeting transcripts are extremely long and the turn-based structure of the meetings makes these datasets particularly suitable for the hierarchical architecture, since speaker turns can be marked by hierarchical tokens. Statistics for AMI and ISCI datasets are illustrated in Table 1. We followed (Shang et al., 2018) for splitting the data.

Since the meeting transcripts are transcribed from a speech to text model, we first built a small lexicon that filters out meaningless words. Then we add BOS tokens to each turn and concatenate all the turns together, which we use as input to the model. Following the conversational dataset approach, we add segment embeddings for each role. We follow the same weight initialization procedure as with the conversational datasets.

4.2 Document level Machine Translation.

Historically, sentence level translation approaches have outperformed document level approaches for translating entire documents. This indicates that

document level approaches are not able to incorporate the larger context of the document during translation. We test our hierarchical model on a translation task and see significant improvements over the non-hierarchical counterpart.

Dataset We evaluate our architecture on the WMT-19 English to German document translation task. For the Europarl corpus, we use a newer version from WMT-20, since it contains metadata useful for finding document boundaries. Dataset stats for EN-DE document pairs are shown in Table 2. We only process 512 tokens at once due to memory constraints. Thus we split documents into chunks of at most 512 tokens. We only split at sentence boundaries to maintain alignment between the source and target languages. We translate each chunk separately and afterwards concatenate the translated chunks to compute the BLEU score for the entire document. We use Moses for preprocessing the data and the fastBPE implementation of Sennrich et al. (2016) for byte-pair encoding with 40000 bpe codes. We build a joined dictionary between English and German.

Model and Optimization We use the Transformer architecture (Vaswani et al., 2017) with 6 encoder/decoder layers, a hidden size of 1024, 16 attention heads and 4096 for the dimension of the feedforward networks. For training, we use an effective batch size of approximately 400 (32 gradient accumulation steps on 4 V100 GPUs with at most 1024 tokens per batch). We use the Adam op-

timizer with $(\beta_1, \beta_2) = (0.9, 0.98)$ and $\varepsilon = 10^{-6}$. We minimize the label smoothed cross entropy loss with smoothing constant 0.1. We train with mixed precision for a total of 50000 steps and warm up the learning rate for 1250 steps to 10^{-4} and decay it linearly afterwards. We use one hierarchical layer. During generation, we use a beam width of 4, no length penalty and we generate until we encounter an EOS token. Note that the above parameters were not extensively tuned and we do not use a monolingual pre-train.

5 Results

As shown in Table 3, we achieve state-of-the-art results on the PubMed and arXiv datasets. As shown in Table 4, our hierarchical seq2seq architecture outperforms its plain seq2seq peer initialized with the same pretrained weights on news datasets. We also achieve state-of-the-art results on the CNN/DM dataset. We outperform the previous baseline by 7 Rouge scores on SAMSum as shown in Table 5. We also achieve state-of-the-art on ROUGE R2 on the AMI dataset, shown in Table 6. Our hierarchical seq2seq architecture outperforms the plain seq2seq baseline by 28 BLEU score on EN-DE document level machine translation as shown in table 7.

6 Analysis

The addition of hierarchical learning improves rouge scores over prior state-of-the-art methods for several summarization datasets. Here we compare the generated output of our hierarchical model with the non-hierarchical counterpart for three instances from the arXiv test data. We also include the introduction of each article. These can be found in Appendix A. Since there is often overlap between the abstract and the introduction, the models have learned to extract sentences from the introduction. We highlight the sentences extracted from the introduction by the hierarchical model in blue and the sentences extracted by the standard model in boldface. We observe that the hierarchical model extracts sentences throughout multiple paragraphs of the introduction, while the plain model generally extracts only from the first paragraph and sometimes does not extract entire sentences. In addition, we include our document level machine translation results in Appendix B. The quality of generated German text matches sentence level translation.

7 Ablation

Encoder-only transformer hierarchical attention. We evaluate our hierarchical attention model on several classification tasks. Instead of using our seq2seq architecture, we design a similar encoder-only architecture with hierarchical attention. This also allows us to easily pre-train the hierarchical layers.

Our architecture is based on the encoder of (Vaswani et al., 2017). We add a hierarchical attention module after the self attention module in each of the encoder layers. Similarly to how we preprocessed the summarization and translation datasets, we insert BOS tokens at the beginning of all sentences.

We follow RoBERTa (Liu et al., 2019) for pre-training our model by using the same dataset, pre-processing steps and pre-training objective. We evaluate the pre-trained model on three downstream tasks: SQuAD 2.0 (Rajpurkar et al., 2018), MNLI-m (Williams et al., 2018) and RACE (Lai et al., 2017). We observe that the pre-training converges faster to a better optimum with lower complexity than RoBERTa with the same hyperparameters. However, downstream task performance does not improve.

	SQuAD 2.0	MNLI-m	RACE
	F1	Acc	Acc
RoBERTa-Base ¹	79.7	84.7	65.6
HAT (Encoder)	79.4	84.7	67.3

Table 8: Hierarchical learning for encoder only.

¹ The results are taken from Table 1 (DOC-SENTENCES) in Liu et al. (2019)

The results are given in Table 8. We observe that for SQuAD 2.0 and MNLI-m our hierarchical model does not perform better than the non-hierarchical model. However, the performance for RACE is significantly better, which suggests that there are some benefits to using hierarchical attention for classification tasks with long source sequences. Note that when fine-tuning on RACE, we had to disable dropout for the first epoch and then set it to 0.1, otherwise the model did not converge.

What has the hierarchical attention learned?

In order to better understand what the hierarchical model has learned, we plot a heatmap of the hierarchical attention between the decoder and the encoder. We use the best performing hierarchical model for this experiment (Table 3). We gener-

	PubMed			arXiv		
	R1	R2	RL	R1	R2	RL
PEGASUS	45.09	19.56	27.42	44.67	17.18	25.73
BigBird	46.32	20.65	42.33	46.63	19.02	41.77
LSH	48.12	21.06	42.72	-	-	-
Transformer-BART	45.54	19.1	34.65	43.92	16.36	39.16
HAT-BART	48.25	21.35	36.69	46.74	19.19	42.20

Table 3: Results on summarization tasks with long source sequences. PEGASUS (Zhang et al., 2019) results are from BigBird (Zaheer et al., 2020) paper. BigBird uses source sequence length of 4096, LSH (Huang et al., 2021) uses 7168, while Transformer-BART and HAT-BART use 3072 due to memory constraints. Transformer-BART and HAT-BART were trained using the same parameter settings.

	CNN/DailyMail			XSum		
	R1	R2	RL	R1	R2	RL
seq2seq-PEGASUS	44.17	21.47	41.11	47.21	24.56	39.25
BigBird	43.84	21.11	40.74	47.12	24.05	38.80
BART	44.16	21.28	40.9	45.14	22.27	37.25
Transformer-BART	44.45	21.27	41.51	45.26	22.19	37.04
HAT-BART	44.48	21.31	41.52	45.92	22.79	37.84

Table 4: Results on standard news summarization tasks. PEGASUS results are from Zaheer et al. (2020). BigBird (Zaheer et al., 2020), Transformer-BART and HAT-BART use a source sequence length of 1024. BART and HAT-BART were trained on the same parameters settings.

	SAMSum		
	R1	R2	RL
DynamicConv + GPT2	45.41	20.65	41.45
Transformer-CNNDM	53.00	28.03	48.59
HAT-CNNDM	53.01	28.27	48.84

Table 5: Results on conversational summarization tasks. The plain Transformer model (Transformer-CNNDM) was initialized by the BART CNN/DM model. The hierarchical model (HAT-CNNDM) was initialized by the hierarchical seq2seq model trained on CNN/DM.

	AMI			ISCI		
	R1	R2	RL	R1	R2	RL
HMNet	53.02	18.57	-	46.28	10.60	-
Transformer-CNNDM	52.06	19.27	50.02	43.77	11.65	41.64
HAT-CNNDM	52.27	20.15	50.57	43.98	10.83	41.36

Table 6: Results on meeting summarization tasks. The plain Transformer model (Transformer-CNNDM) was initialized by the BART CNN/DM model. The hierarchical model (HAT-CNNDM) was initialized by the hierarchical Transformer model trained on CNN/DM.

WMT-19 EN-DE	
d-BLEU	
Transformer (no-pretrain)	7.7
HAT (no-pretrain)	35.7

Table 7: Results on WMT-19 EN-DE document translation task. The plain Transformer result was obtained from MBART (Liu et al., 2020). Both models are initialized randomly.

ate summaries for each of the sample articles and record the hierarchical attention between the decoder and the BOS embeddings from the encoder at each step of generating the summary. For each of the 12 decoder layers and each of the 16 attention heads we get a distribution over the BOS tokens for each of the generated summary tokens. To visualize the attention more concisely, we aggregate across the attentions heads by choosing only the 16 BOS tokens with the highest weight for each generated token. We normalize the distribution such that the weights sum to 1. The hierarchical attention heatmaps for each layer are shown in Figures 2, 3 and 4.

We see that across different layers the model attends to different BOS tokens across the entire document. For each layer there are several horizontal lines, which indicates that some BOS tokens are assigned large weights at each step during generation. However, we also observe many individual points and discontinuous horizontal lines on the heatmaps, indicating that the model selectively attends to different BOS tokens across the entire document. We note that in the first few layers, the model seems to attend to the layers more uniformly while in the last few layers the model attends more heavily to certain BOS tokens.

8 Conclusion and Future Work

We designed a transformer based architecture with hierarchical attention and obtained improvements on several sequence to sequence generation tasks, especially summarization datasets with long source documents. We showed significant improvements on document-level machine translation on the WMT-19 EN-DE translation task, as compared to our baseline. We did not see significant gains when applying hierarchical attention to encoder-only classification tasks.

Future work might include further investigating the benefits of hierarchical attention for document-level machine translation, since we only experimented with a single dataset and language pair; we did not tune the hyperparameters extensively. We believe that hierarchical attention document-level translation could outperform sentence-level translation. In addition, we initialize the hierarchical components randomly since pre-training similar to BART where permuted sentences might be difficult. However, as the hierarchical attention operates on the rich pre-trained BOS embeddings, we believe

that would not be a major issue, and pre-training on the hierarchical components could further increase the performance on sequence to sequence tasks.

Finally, although we did not see significant gains when using a hierarchical encoder-only model for classification, we believe that some modifications on hierarchical attention might improve over current non-hierarchical models. Particularly, the impact of dropout on hierarchical layers should be investigated more closely.

Acknowledgments

We would like to thank Kevin Terrell for reading the first version of our draft. We acknowledge the support from the Google Cloud team and the PyTorch/XLA team, particularly Taylan Bilal.

References

- Idan Achituve, Sarit Kraus, and Jacob Goldberger. 2019. [Interpretable online banking fraud detection based on hierarchical attention mechanism](#). In *MLSP*, pages 1–6.
- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2016. [Neural machine translation by jointly learning to align and translate](#).
- Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv:2004.05150*.
- Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Mael Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, Guillaume Lathoud, Mike Lincoln, Agnes Lisowska, Iain McCowan, Wilfried Post, Dennis Reidsma, and Pierre Wellner. 2006. The ami meeting corpus: A pre-announcement. In *Machine Learning for Multimodal Interaction, Second International Workshop*, number 10 in Lecture Notes in Computer Science, pages 28–39. Springer.
- 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.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. [Unsupervised cross-lingual representation learning at scale](#). In

- Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. 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.
- Shang Gao, Michael T Young, John X Qiu, Hong-Jun Yoon, James B Christian, Paul A Fearn, Georgia D Tourassi, and Arvind Ramanathan. 2018. Hierarchical attention networks for information extraction from cancer pathology reports. *Journal of the American Medical Informatics Association*, 25(3):321–330.
- Xiaoyan Gao, Fuli Feng, Xiangnan He, Heyan Huang, Xinyu Guan, Chong Feng, Zhaoyan Ming, and Tat-Seng Chua. 2020. Hierarchical attention network for visually-aware food recommendation. *IEEE Trans. Multim.*, 22(6):1647–1659.
- Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. SAMSum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 70–79, Hong Kong, China. Association for Computational Linguistics.
- Beliz Gunel, Chenguang Zhu, Michael Zeng, and Xuedong Huang. 2019. Mind the facts: Knowledge-boosted coherent abstractive text summarization. In *NeurIPS 2019, Knowledge Representation Reasoning Meets Machine Learning (KR2ML) Workshop*.
- Xu Han, Pengfei Yu, Zhiyuan Liu, Maosong Sun, and Peng Li. 2018. Hierarchical relation extraction with coarse-to-fine grained attention. In *EMNLP*, pages 2236–2245.
- Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.
- Chao Huang, Chuxu Zhang, Peng Dai, and Liefeng Bo. 2020. Cross-interaction hierarchical attention networks for urban anomaly prediction. In *IJCAI*, pages 4359–4365.
- Luyang Huang, Shuyang Cao, Nikolaus Parulian, Heng Ji, and Lu Wang. 2021. Efficient attentions for long document summarization.
- Srinivasan Iyer, Ioannis Konstas, Alvin Cheung, and Luke Zettlemoyer. 2016. Summarizing source code using a neural attention model. In *ACL (1)*.
- Adam Janin, Don Baron, Jane Edwards, Dan Ellis, David Gelbart, Nelson Morgan, Barbara Peskin, Thilo Pfau, Elizabeth Shriberg, Andreas Stolcke, and Chuck Wooters. 2003. The icsi meeting corpus. pages 364–367.
- Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR (Poster)*.
- Wojciech Kryscinski, Romain Paulus, Caiming Xiong, and Richard Socher. 2018. Improving abstraction in text summarization. In *EMNLP*, pages 1808–1817.
- Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard H. Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. In *EMNLP*, pages 785–794.
- 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.
- Qi Li, Tianshi Li, and Baobao Chang. 2016. Discourse parsing with attention-based hierarchical neural networks. In *EMNLP*, pages 362–371.
- Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- Li Mi and Zhenzhong Chen. 2020. Hierarchical graph attention network for visual relationship detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13886–13895.
- Lesly Miculicich, Dhananjay Ram, Nikolaos Pappas, and James Henderson. 2018. Document-level neural machine translation with hierarchical attention networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2947–2954, Brussels, Belgium. Association for Computational Linguistics.
- Rada Mihalcea and Hakan Ceylan. 2007. Explorations in automatic book summarization. In *EMNLP-CoNLL*, pages 380–389.
- Yoshio Nakao. 2000. An algorithm for one-page summarization of a long text based on thematic hierarchy detection. In *ACL*.

- 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. Association for Computational Linguistics.
- Romain Paulus, Caiming Xiong, and Richard Socher. 2018. [A deep reinforced model for abstractive summarization](#). In *International Conference on Learning Representations*.
- 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.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. [Exploring the limits of transfer learning with a unified text-to-text transformer](#). *Journal of Machine Learning Research*, 21(140):1–67.
- Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789.
- Heechang Ryu, Hayong Shin, and Jinkyoo Park. 2020. Multi-agent actor-critic with hierarchical graph attention network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7236–7243.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. [Neural machine translation of rare words with subword units](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.
- Guokan Shang, Wensi Ding, Zekun Zhang, Antoine Tixier, Polykarpos Meladianos, Michalis Vazirgiannis, and Jean-Pierre Lorré. 2018. [Unsupervised abstractive meeting summarization with multi-sentence compression and budgeted submodular maximization](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 664–674, Melbourne, Australia. Association for Computational Linguistics.
- Jun Song, Jun Xiao, Fei Wu, Haishan Wu, Tong Zhang, Zhongfei Mark Zhang, and Wenwu Zhu. 2017. [Hierarchical contextual attention recurrent neural network for map query suggestion](#). *IEEE Trans. Knowl. Data Eng.*, 29(9):1888–1901.
- Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 3104–3112, Cambridge, MA, USA. MIT Press.
- Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. 2017. Abstractive document summarization with a graph-based attentional neural model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1171–1181.
- Mayesha Tasnim, Diego Collarana, Damien Graux, Fabrizio Orlandi, and Maria-Esther Vidal. 2019. [Summarizing entity temporal evolution in knowledge graphs](#). In *WWW (Companion Volume)*, pages 961–965.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6000–6010.
- Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2018. [A broad-coverage challenge corpus for sentence understanding through inference](#). In *NAACL-HLT*, pages 1112–1122.
- Shuzhe Wu, Meina Kan, Shiguang Shan, and Xilin Chen. 2019. Hierarchical attention for part-aware face detection. *International Journal of Computer Vision*, 127(6-7):560–578.
- Chen Xing, Yu Wu, Wei Wu, Yalou Huang, and Ming Zhou. 2018. Hierarchical recurrent attention network for response generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.
- Jiacheng Xu, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020a. [Discourse-aware neural extractive text summarization](#). In *ACL*, pages 5021–5031.
- Yunqiu Xu, Meng Fang, Ling Chen, Yali Du, Joey Tianyi Zhou, and Chengqi Zhang. 2020b. Deep reinforcement learning with stacked hierarchical attention for text-based games. *Advances in Neural Information Processing Systems*, 33.
- Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. [Xlnet: Generalized autoregressive pretraining for language understanding](#). In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.
- Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489.

- Zihao Ye, Qipeng Guo, Quan Gan, Xipeng Qiu, and Zheng Zhang. 2019. Bp-transformer: Modelling long-range context via binary partitioning. *arXiv preprint arXiv:1911.04070*.
- Manzil Zaheer, Guru Guruganesh, Kumar 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. *Advances in Neural Information Processing Systems*, 33.
- Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2019. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization.
- Ningyu Zhang, Shumin Deng, Juan Li, Xi Chen, Wei Zhang, and Huajun Chen. 2020. Summarizing chinese medical answer with graph convolution networks and question-focused dual attention. In *EMNLP (Findings)*, pages 15–24.
- Yizhe Zhang, Dinghan Shen, Guoyin Wang, Zhe Gan, Ricardo Henao, and Lawrence Carin. 2017. Deconvolutional paragraph representation learning. In *NIPS*, pages 4172–4182.
- Yue Zhao, Xiaolong Jin, Yuanzhuo Wang, and Xueqi Cheng. 2018. Document embedding enhanced event detection with hierarchical and supervised attention. In *ACL (2)*, pages 414–419.
- Zhou Zhao, Haojie Pan, Changjie Fan, Yan Liu, Linlin Li, and Min Yang. 2019. Abstractive meeting summarization via hierarchical adaptive segmental network learning. In *WWW*, pages 3455–3461.
- Chenguang Zhu, Ruochen Xu, Michael Zeng, and Xuedong Huang. 2020. A hierarchical network for abstractive meeting summarization with cross-domain pretraining. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 194–203.
- Asli Çelikyilmaz, Antoine Bosselut, Xiaodong He, and Yejin Choi. 2018. Deep communicating agents for abstractive summarization. In *NAACL-HLT*, pages 1662–1675.

A Appendix

deep neural networks (dnns) have been receiving ubiquitous success in wide applications , ranging from computer vision @xcite , to speech recognition @xcite , natural language processing @xcite , and domain adaptation @xcite . as the sizes of data mount up , people usually have to increase the number of parameters in dnns so as to absorb the vast volume of supervision . high performance computing techniques are investigated to speed up dnn training , concerning optimization algorithms , parallel synchronisations on clusters w / o gpus , and stochastic binarization / ternarization , etc @xcite . on the other hand the memory and energy consumption is usually , if not always , constrained in industrial applications @xcite . for instance , for commercial search engines (e.g. , google and baidu) and recommendation systems (e.g. , netflix and youtube) , the ratio between the increased model size and the improved performance should be considered given limited online resources . compressing the model size becomes more important for applications on mobile and embedded devices @xcite . having dnns running on mobile apps owns many great features such as better privacy , less network bandwidth and real time processing . however , the energy consumption of battery - constrained mobile devices is usually dominated by memory access , which would be greatly saved if a dnn model can fit in on - chip storage rather than dram storage (c.f. @xcite for details) . a recent trend of studies are thus motivated to focus on compressing the size of dnns while mostly keeping their predictive performance @xcite . **with different intuitions** , there are mainly two types of dnn compression methods , which could be used in conjunction for better parameter savings . the first type tries to revise the training target into more informative supervision using _ dark knowledge _ . in specific , hinton _ @xcite suggested to train a large network ahead , and distill a much smaller model on a combination of the original labels and the soft - output by the large net . the second type observes the redundancy existence in network weights @xcite , and exploits techniques to constrain or reduce the number of free - parameters in dnns during learning . this paper focuses on the latter type . to constrain the network redundancy , efforts @xcite formulated an original weight matrix into either low - rank or fast - food decompositions . moreover @xcite proposed a simple - yet - effective pruning - retraining iteration during training , followed by quantization and fine - tuning . @xcite proposed hashednets to efficiently implement parameter sharing prior to learning , and showed notable compression with much less loss of accuracy than low - rank decomposition . more precisely , prior to training , a hash function is used to randomly group (virtual) weights into a small number of buckets , so that all weights mapped into one hash bucket directly share a same value . hashednets was further deliberated in frequency domain for compressing convolutional neural networks in @xcite . in applications , we observe hashednets compresses model sizes greatly at marginal loss of accuracy for some situations , whereas also significantly loses accuracy for others . after revisiting its mechanism , we conjecture this instability comes from at least three factors . first , hashing and training are disjoint in a two - phase manner , i.e. , once inappropriate collisions exist , there may be no much optimization room left for training . second , _ one single hash function _ is used to fetch a single value in the compression space , whose collision risk is larger than multiple hashes @xcite . third , parameter sharing within a buckets implicitly uses _ identity mapping _ from the hashed value to the virtual entry . this paper proposes an approach to relieve this instability , still in a two - phase style for preserving efficiency . specifically , we use _ multiple hash functions _ @xcite to map per virtual entry into multiple values in compression space . then an additional network plays in a _ mapping function _ role from these hashed values to the virtual entry before hashing , which can be also regarded as “ reconstructing ” the virtual entry from its multiple hashed values . plugged into and jointly trained within the original network , the reconstruction network is of a comparably ignorable size , i.e. , at low memory cost . this functional hashing structure includes hashednets as a degenerated special case , and facilitates less value collisions and better value reconstruction . shortly denoted as funhashnn , our approach could be further extended with dual space hashing and multi - hops . since it imposes no restriction on other network design choices (e.g. dropout and weight sparsification) , funhashnn can be considered as a standard tool for dnn compression . experiments on several datasets demonstrate promisingly larger reduction of model sizes and/or less loss on prediction accuracy , compared with hashednets .

Table 9: Introduction of sample article from arXiv test data (1).

HAT	<p>deep neural networks (dnns) have been receiving ubiquitous success in wide applications , ranging from computer vision , to speech recognition , natural language processing , and domain adaptation . as the sizes of data mount up , people usually have to increase the number of parameters in dnn so as to absorb the vast volume of supervision . however , the ratio between the increased model size and the improved performance should be considered given limited online resources . compressing the model size becomes more important for applications on mobile and embedded devices . with different intuitions , there are mainly two types of dnn compression methods , which could be used in conjunction for better parameter savings . the first type tries to revise the training target into more informative supervision using _ dark knowledge_ . in specific , hinton _ _ suggested to train a large network ahead , and distill a much smaller model on a combination of the original labels and the soft - output by the large net . the second type observes the redundancy existence in network weights , and exploits techniques to constrain or reduce the free - parameters during learning . hashednets is one of the two types , and has been studied as a dimension reduction method for reducing model storage size without maintaining the mapping matrices like random projection . in applications , we observe hashednet compresses model sizes greatly at marginal loss of accuracy for some situations , whereas also significantly loses accuracy for others . after revisiting its mechanism , we conjecture this instability comes from at least three factors . first , hashing and training are disjoint in a two - phase manner , i.e. , once inappropriate collisions exist , there may be no much optimization room left for training . second , one single hash function is used to fetch a single value in the compression space , whose collision risk is larger than multiple hashes . third , parameter sharing within a buckets implicitly uses _ identity mapping _ from the hashed value to the virtual entry . this paper proposes an approach to relieve this instability . specifically , we use _ multiple hash functions _ to map per virtual entry into multiple values in compression space . then an additional network plays in a _ mapping function _ role from these hashed values to the virtual entry before hashing , which can be also regarded as “ reconstructing ” the virtual entries from its multiple hashed values . plugged into and jointly trained within the original network , the reconstruction network is of a comparably ignorable size , and requires little additional memory . this functional hashing structure includes hashed nets as a degenerated special case , and facilitates less value collisions and better value reconstruction . shortly denoted as funhashnn , our approach could be further extended with dual space hashing and multi - hops . experiments on several datasets demonstrate promisingly larger reduction of model sizes and/or less loss on prediction accuracy , compared with hashed net .</p>
Transformer	<p>deep neural networks (dnns) have been receiving ubiquitous success in wide applications , ranging from computer vision , to speech recognition , natural language processing , and domain adaptation . as the sizes of data mount up , people usually have to increase the number of parameters in dnns so as to absorb the vast volume of supervision . high performance computing techniques are investigated to speed up dnn training , concerning optimization algorithms , parallel synchronisations on clusters w / o gpus , and stochastic binarization / ternarization . on the other hand , the memory and energy consumption is usually , if not always , constrained in industrial applications . for instance , for commercial search engines (e.g. , google and baidu) and recommendation systems (e.g. , netflix and youtube) , the ratio between the increased model size and the improved performance should be considered given limited online resources . compressing the model size becomes more important for applications on mobile and embedded devices . with different intuitions</p>
Target	<p>as the complexity of deep neural networks (dnns) trend to grow to absorb the increasing sizes of data , memory and energy consumption has been receiving more and more attentions for industrial applications , especially on mobile devices . this paper presents a novel structure based on functional hashing to compress dnns , namely funhashnn . for each entry in a deep net , funhashnn uses multiple low - cost hash functions to fetch values in the compression space , and then employs a small reconstruction network to recover that entry . the reconstruction network is plugged into the whole network and trained jointly . funhashnn includes the recently proposed hashednets @xcite as a degenerated case , and benefits from larger value capacity and less reconstruction loss . we further discuss extensions with dual space hashing and multi - hops . on several benchmark datasets , funhashnn demonstrates high compression ratios with little loss on prediction accuracy .</p>

Table 10: ROUGE(HAT): 36.91/13.47/34.90; ROUGE(Transformer): 37.11/10.38/34.36

convolutional neural networks typically consist of an input layer , a number of hidden layers , followed by a softmax classification layer . the input layer , and each of the hidden layers , is represented by a three - dimensional array with size , say , x_0 . the second and third dimensions are spatial . the first dimension is simply a list of features available in each spatial location . for example , with rgb color images x_1 is the image size and x_2 is the number of color channels . the input array is processed using a mixture of convolution and pooling operations . as you move forward through the network , x_3 decreases while x_4 is increased to compensate . when the input array is spatially sparse , it makes sense to take advantage of the sparsity to speed up the computation . more importantly , knowing you can efficiently process sparse images gives you greater freedom when it comes to preparing the input data . consider the problem of online isolated character recognition ; `_online_` means that the character is captured as a path using a touchscreen or electronic stylus , rather than being stored as a picture . recognition of isolated characters can be used as a building block for reading cursive handwriting , and is a challenging problem in its own right for languages with large character sets . each handwritten character is represented as a sequence of strokes ; each stroke is stored as a list of x_5 - and x_6 -coordinates . we can draw the characters as x_1 binary images : zero for background , one for the pen color . the number of pixels is x_7 , while the typical number of non - zero pixels is only x_8 , so the first hidden layer can be calculated much more quickly by taking advantage of sparsity . another advantage of sparsity is related to the issue of spatial padding for convolutional networks . convolutional networks conventionally apply their convolutional filters in `_valid_` mode they are only applied where they fit completely inside the input layer . this is generally suboptimal as makes it much harder to detect interesting features on the boundary of the input image . there are a number of ways of dealing with this . padding the input image `_xcite_` with zero pixels . this has a second advantage : training data augmentation can be carried out in the form of adding translations , rotations , or elastic distortions to the input images . adding small amounts of padding to each of the convolutional layers of the network ; depending on the amount of padding added this may be equivalent to applying the convolutions in `_full_` mode . this has a similar effect to adding lots of padding to the input image , but it allows less flexibility when it comes to augmenting the training data . applying the convolutional network to a number of overlapping subsets of the image `_xcite_` ; this is useful if the input images are not square . this can be done relatively computationally efficiently as there is redundancy in the calculation of the lower level convolutional filters . however , the (often large) fully connected classification layers of the network must be evaluated several times . sparsity has the potential to combine the best features of the above . the whole object can be evaluated in one go , with a substantial amount of padding added at no extra cost . in section [`deepcnn`]-[`deepcnn`] we describe a family of convolutional networks with many layers of max - pooling . in section [`sparsity`]-[`nn`] we describe how sparsity applies to character recognition and image recognition . in section [`results`] we give our results . in section [`sec : conclusion`] we discuss other possible uses of sparse cnns .

Table 11: Introduction of sample article from arXiv test data (2).

arXiv prediction and target - 2	
HAT	spatial sparsity is an important feature of convolutional neural networks (cnns) . when the input array is sparse , it makes sense to take advantage of the sparsity to speed up the computation . more importantly , knowing you can efficiently process sparse images gives you greater freedom when it comes to preparing the input data . we describe a family of cnn architectures with many layers of max - pooling , and show how sparsity can be used to improve the performance of online character recognition and image recognition .
Transformer	convolutional neural networks (cnns) typically consist of an input layer , a number of hidden layers , followed by a softmax classification layer . the input layer is represented by a three - dimensional array with size , say , x_0 . the second and third dimensions are spatial . the first dimension is simply a list of features available in each spatial location . for example , with rgb color images x_1 is the image size and x_2 is the number of color channels . the input array is processed using a mixture of convolution and pooling operations . as you move forward through the network , x_3 decreases while x_4 is increased to compensate . when the input array is spatially sparse , it makes sense to take advantage of the sparsity to speed up the computation . more importantly , knowing you can efficiently process sparse images , at the top of the network is an output layer . if the input layer has spatial size x_1 with
Target	convolutional neural networks (cnns) perform well on problems such as handwriting recognition and image classification . however , the performance of the networks is often limited by budget and time constraints , particularly when trying to train deep networks . motivated by the problem of online handwriting recognition , we developed a cnn for processing spatially - sparse inputs ; a character drawn with a one - pixel wide pen on a high resolution grid looks like a sparse matrix . taking advantage of the sparsity allowed us more efficiently to train and test large , deep cnns . on the casia - olhwdb1.1 dataset containing 3755 character classes we get a test error of 3.82% . although pictures are not sparse , they can be thought of as sparse by adding padding . applying a deep convolutional network using sparsity has resulted in a substantial reduction in test error on the cifar small picture datasets : 6.28% on cifar-10 and 24.30% for cifar-100 . * keywords : * online character recognition , convolutional neural network , sparsity , computer vision

Table 12: ROUGE(HAT): 37.25/12.25/34.01; ROUGE(Transformer): 33.54/7.01/32.28

question answering (qa) aims to automatically understand natural language questions and to respond with actual answers . the state - of - the - art qa systems usually work relatively well for factoid , list and definition questions , but they might not necessarily work well for real world questions , where more comprehensive answers are required . frequently asked questions (faq) based qa is an economical and practical solution for general qa @xcite . instead of answering questions from scratch , faq - based qa tries to search the faq archives and check if a similar question was previously asked . if a similar question is found , the corresponding answer is returned to the user . the faq archives are usually created by experts , so the returned answers are usually of higher - quality . the core of faq - based qa is to calculate semantic similarities between questions . this is a very challenging task , because two questions , which share the same meaning , may be quite different at the word or syntactic level . for example , how do i add a vehicle to this policy ? ” and what should i do to extend this policy for my new car ? ” have few words in common , but they share the same answer . in the past two decades , many efforts have been made to tackle this lexical gap problem . one type of methods tried to bridge the lexical gap by utilizing semantic lexicons , like wordnet @xcite . another method treated this task as a statistical machine translation problem , and employed a parallel question set to learn word - to - word or phrase - to - phrase translation probabilities @xcite . both of these methods have drawbacks . the first method is hard to adapt to many other languages , because the semantic lexicon is unavailable . for the second method , a large parallel question set is required to learn the translation probabilities , which is usually hard or expensive to acquire . to overcome these drawbacks , we utilize distributed word representations to calculate the similarity between words , which can be easily trained by only using amount of monolingual data . **in this paper , we propose a novel word - alignment - based method to solve the faq - based qa tasks .** the characteristics of our method include : (1) a neural network model for calculating question similarity with word alignment features . for an input question and a candidate question , the similarities of each word pairs (between the two questions) are calculated first , and then the best word alignment for the two questions is computed . we extract a vector of dense features from the word alignment , then import the feature vector into a neural network and calculate the question similarity in the network s output layer . (2) a bootstrap - based feature extraction method . the faq archives usually contain less than a few hundred questions , and in order to avoid overfitting , we are unable to use too many sparse features . therefore , we come up with this method to extract a small set of effective sparse features according to our system s ranking results . (3) a learning - to - rank algorithm for training . the faq - based qa task is essentially a ranking task , our model not only needs to calculate a proper similarity for each question pair , but also needs to rank the most relevant one on top of the other candidates . so we propose a learning - to - rank method to train parameters more suitable for ranking . experimental results , conducted on faq archives from three languages , demonstrate that our method is very effective . we also evaluate our method on the answer sentence selection task . experimental results on the standard trec data set show that our method outperforms all previous state - of - the - art systems .

Table 13: Introduction of sample article from arXiv test data (3).

HAT	<p>in this paper , we propose a novel word - alignment - based method to solve the frequently asked questions (faq) - based question answering (qa) tasks . the characteristics of our method include : (1) a neural network model for calculating question similarity with word alignment features . for an input question and a candidate question , the similarities of each word pairs (between the two questions) are calculated first , and then the best word alignment for the two question is computed . we extract a vector of dense features from the word alignment , then import the feature vector into a neural networks and calculate the question similarity in the network s output layer . (2) a bootstrap - based feature extraction method . the faq archives usually contain less than a few hundred questions , and in order to avoid overfitting , we are unable to use too many sparse features . therefore , we come up with this method to extract a small set of effective sparse features according to our system s ranking results . (3) a learning - to - rank algorithm for training . since faq - based qa task is essentially a ranking task , our model not only needs to calculate a proper similarity for each question pair , but also needs to rank the most relevant candidate on top of the other candidates . so we propose an explicit learning -to - rank method to train parameters more suitable for ranking . experimental results , conducted on faq archive from three languages , demonstrate that our method is very effective . we also evaluate our method on the answer sentence selection task , and experimental results on the standard trec data set show that our system outperforms all previous state - of - the - art systems .</p>
Transformer	<p>faq - based question answering (qa) aims to automatically understand natural language questions and to respond with actual answers . the state - of - the - art systems usually work relatively well for factoid , list and definition questions , but they might not necessarily work well for real world questions , where more comprehensive answers are required . frequently asked questions (faq) based qa is an economical and practical solution for general qa . instead of answering questions from scratch , faq - based qa tries to search the faq archives and check if a similar question was previously asked . if a similar question is found , the corresponding answer is returned to the user . the core of faq - based qa is to calculate semantic similarities between questions . this is a very challenging task , because two questions , which share the same meaning , may be quite different at the word or syntactic frequency level . in this paper , we propose a novel word - alignment - based method to solve the faq - based</p>
Target	<p>in this paper , we propose a novel word - alignment - based method to solve the faq - based question answering task . first , we employ a neural network model to calculate question similarity , where the word alignment between two questions is used for extracting features . second , we design a bootstrap - based feature extraction method to extract a small set of effective lexical features . third , we propose a learning - to - rank algorithm to train parameters more suitable for the ranking tasks . experimental results , conducted on three languages (english , spanish and japanese) , demonstrate that the question similarity model is more effective than baseline systems , the sparse features bring 5% improvements on top-1 accuracy , and the learning - to - rank algorithm works significantly better than the traditional method . we further evaluate our method on the answer sentence selection task . our method outperforms all the previous systems on the standard trec data set .</p>

Table 14: ROUGE(HAT): 58.71/37.41/55.37; ROUGE(Transformer): 38.16/13.91/35.53

B Appendix

WMT19 EN-DE source - 1

Maxine Waters denies staffer leaked GOP senators' data, blasts 'dangerous lies' and 'conspiracy theories' U.S. Rep. Maxine Waters on Saturday denounced allegations that a member of her staff had posted the personal information of three Republican U.S. senators onto the lawmakers' Wikipedia pages. The Los Angeles Democrat asserted that the claims were being pedaled by "ultra-right wing" pundits and websites. "Lies, lies, and more despicable lies," Waters said in a statement on Twitter. The released information reportedly included the home addresses and phone numbers for U.S. Sens. Lindsey Graham of South Carolina, and Mike Lee and Orrin Hatch, both of Utah. The information appeared online Thursday, posted by an unknown person on Capitol Hill during a Senate panel's hearing on the sexual misconduct allegations against Supreme Court nominee Brett Kavanaugh. The leak came sometime after the three senators had questioned Kavanaugh. Conservative sites such as Gateway Pundit and RedState reported that the IP address that identifies the source of the posts was associated with Waters' office and released the information of a member of Waters' staff, the Hill reported. "This unfounded allegation is completely false and an absolute lie," Waters continued. "The member of my staff - whose identity, personal information, and safety have been compromised as a result of these fraudulent and false allegations - was in no way responsible for the leak of this information. This unfounded allegation is completely false and an absolute lie." Waters' statement quickly drew criticism online, including from former White House press secretary Ari Fleischer. "This denial is angry," Fleischer wrote. "This suggests she doesn't have the temperament to be a Member of Congress. When someone is accused of something they didn't do, they must not be angry. They must not be defiant. They must not question the motives of the accuser. They must be calm and serene." Fleischer was appearing to compare Waters' reaction to the Democrats' criticism of Judge Kavanaugh, who was accused by critics of seeming too angry during Thursday's hearing. Omar Navarro, a Republican candidate running to unseat Waters in the midterm elections, also voiced his thoughts on Twitter. "Big if true," he tweeted. In her statement, Waters said her office had alerted "the appropriate authorities and law enforcement entities of these fraudulent claims. "We will ensure that the perpetrators will be revealed," she continued, "and that they will be held legally liable for all of their actions that are destructible and dangerous to any and all members of my staff."

Table 15: English source text of sample document from WMT19 EN-DE test data (1).

HAT	<p>Maxine Waters bestreitet, dass ein Mitglied ihres Personals die persönlichen Informationen von drei republikanischen Senatoren auf die Wikipedia-Seiten der Gesetzgeber gepostet habe. Die Demokraten in Los Angeles behaupteten, dass die Behauptungen von "ultra-right wing" Pundits und Websites manipuliert würden. "Lies, lies und despicable lies," sagte Waters in einer Erklärung auf Twitter. Die veröffentlichten Informationen enthielten angeblich die Home-Adressen und Telefonnummern von U.S. Sens. Lindsey Graham von South Carolina, und Mike Lee und Orrin Hatch, beide von Utah. Die Informationen erschienen am Donnerstag, gepostet von einer unbekannten Person auf Capitol Hill während einer Anhörung des Senats über die sexuellen Missstände gegen den Supreme Court nominee Brett Kavanaugh. Das Blatt kam irgendwann, nachdem die drei Senatoren Kavanaugh in Frage gestellt hatten. Konservative Websites wie Gateway Pundit und RedState berichteten, dass die IP-Adresse, die die Quelle der Beiträge identifiziert, mit Waters "Büro in Verbindung gebracht wurde und die Informationen eines Mitglieds von Waters" Mitarbeitern veröffentlichte, berichtete Hill. "Diese unbegründete Behauptung ist völlig falsch und eine absolute Lüge," fuhr Waters fort. "Das Mitglied meines Personals - dessen Identität, persönliche Informationen und Sicherheit als Folge dieser betrügerischen und falschen Behauptungen beeinträchtigt wurden - war in keiner Weise verantwortlich für das Blatt dieser Informationen. Diese unbegründete Behauptung ist völlig falsch und eine absolute Lüge." Waters "Erklärung zog schnell Kritik online, darunter von der ehemaligen Pressesprecherin des Weißen Hauses, Ari Fleischer." Diese Leugnung ist wütend, "schrieb Fleischer." Dies legt nahe, dass sie nicht das Gefühl hat, Mitglied des Kongresses zu sein. Wenn jemand von etwas beschuldigt wird, das sie nicht getan haben, dürfen sie nicht wütend sein. Sie dürfen nicht wütend sein. Sie dürfen die Motive des Akkreditierenden nicht in Frage stellen. Sie müssen ruhig und heiter sein. "Fleischer erschien, um Waters" Reaktion auf die Kritik der Demokraten an Richter Kavanaugh zu vergleichen, der von Kritikern beschuldigt wurde, während der Anhörung am Donnerstag zu wütend zu sein. Omar Navarro, ein republikanischer Kandidat, der bei den Zwischenwahlen die Waters verlassen wollte, äußerte auch seine Gedanken auf Twitter. "Big if true," so sagte er. In ihrer Erklärung sagte Waters, ihr Büro habe "die zuständigen Behörden und Strafverfolgungsbehörden auf diese betrügerischen Behauptungen aufmerksam gemacht." Wir werden sicherstellen, dass die Täter enthüllt werden, "fuhr sie fort," und dass sie für all ihre Handlungen haftbar gemacht werden, die für alle Mitglieder meines Personals zerstörbar und gefährlich sind. "</p>
Target	<p>Maxine Waters bestreitet, dass ein Angestellter die Daten von GOP-Senatoren geleakt hat, kritisiert die Vorwürfe als "gefährliche Lügen" und "Verschwörungstheorien" Die US-Abgeordnete Maxine Waters hat am Samstag Behauptungen zurückgewiesen, nach denen einer ihrer Mitarbeiter persönliche Daten von drei republikanischen US-Senatoren auf den Wikipedia-Seiten der Gesetzgeber veröffentlicht haben soll. Die in Los Angeles ansässige Demokratin ist der Überzeugung, dass die Behauptungen von Angehörigen des ultralinken Flügels und über Websites lanciert wurden. "Lügen, Lügen und noch mehr widerwärtige Lügen", teilte Waters in einer Stellungnahme auf Twitter mit. Zu den veröffentlichten Informationen gehörten Berichten zufolge die Privatadressen und Telefonnummern der US-Senatorin Lindsey Graham aus South Carolina sowie der US-Senatoren Mike Lee und Orrin Hatch, die beide aus Utah stammen. Die Informationen erschienen Donnerstag online, gepostet von einer unbekannten Person auf dem Kapitolshügel während einer Anhörung des Senats über die Vorwürfe des sexuellen Fehlverhaltens gegenüber des Kandidaten für den Obersten Gerichtshofs Brett Kavanaugh. Die Daten der drei Senatoren wurden kurz nach der Befragung von Kavanaugh veröffentlicht. Konservative Webseiten wie Gateway Pundit und RedState berichteten, dass die IP-Adresse, die die Quelle der Stellen identifiziert, mit Waters ' Büro in Verbindung gebracht wurde und die Informationen eines Mitarbeiters von Waters veröffentlicht habe, berichtet Hill. "Dieser unbegründete Vorwurf ist völlig falsch und eine absolute Lüge", so Waters weiter. "Der Angestellte unter meinen Mitarbeitern - dessen Identität, persönliche Informationen, persönliche Daten und Sicherheit als Ergebnis dieser betrügerischen und falschen Behauptungen beeinträchtigt wurden - war in keiner Weise verantwortlich, dass diese Informationen geleakt wurde. Diese haltlose Behauptung ist vollkommen falsch und eine absolute Lüge. Waters Erklärung führte schnell online zu Kritik, unter anderem von dem ehemaligen Pressesekretär des Weißen Hauses Ari Fleischer. "Dass die Sache geleugnet wird, ist äußerst ärgerlich", schrieb Fleischer. "Dies lässt darauf schließen, dass sie als Kongressmitglied ungeeignet ist. Wenn jemand beschuldigt wird, etwas getan zu haben, darf er nicht böse sein. Sie dürfen nicht aufsässig sein. Sie dürfen die Motive des Klägers nicht in Frage stellen. Sie müssen ruhig und gelassen sein". Fleischer schien Waters Reaktion mit der Kritik der Demokraten an Richter Kavanaugh zu vergleichen, der von Kritikern beschuldigt wurde, während der Anhörung am Donnerstag zu wütend zu erscheinen. Auch Omar Navarro, ein republikanischer Kandidat, der bei den Zwischenwahlen neben dem sitzlosen Waters kandidiert, äußerte seine Gedanken auf Twitter. "Big, if true", hat er getweetet. In ihrem Statement sagte Waters, dass ihr Büro "die zuständigen Behörden und Strafverfolgungsbehörden über die betrügerischen Ansprüche in Kenntnis gesetzt hat. "Wir werden dafür sorgen, dass die Täter gefasst und für ihre Taten, die sich destruktiv und gefährlich auf meine Mitarbeiter auswirken, zur Rechenschaft gezogen werden", fuhr sie fort.</p>

Table 16: BLEU(HAT): 34.3

Shark injures 13-year-old on lobster dive in California A shark attacked and injured a 13-year-old boy Saturday while he was diving for lobster in California on the opening day of lobster season, officials said. The attack occurred just before 7 a.m. near Beacon's Beach in Encinitas. Chad Hammel told KSWB-TV in San Diego he had been diving with friends for about half an hour Saturday morning when he heard the boy screaming for help and then paddled over with a group to help pull him out of the water. Hammel said at first he thought it was just excitement of catching a lobster, but then he "realized that he was yelling, 'I got bit! I got bit!' His whole clavicle was ripped open," Hammel said he noticed once he got to the boy. "I yelled at everyone to get out of the water: 'There's a shark in the water!'" Hammel added. The boy was airlifted to Rady Children's Hospital in San Diego where he is listed in critical condition. The species of shark responsible for the attack was unknown. Lifeguard Capt. Larry Giles said at a media briefing that a shark had been spotted in the area a few weeks earlier, but it was determined not to be a dangerous species of shark. Giles added the victim sustained traumatic injuries to his upper torso area. Officials shut down beach access from Ponto Beach in Casablad to Swami's in Ecinitas for 48 hours for investigation and safety purposes. Giles noted that there are more than 135 shark species in the area, but most are not considered dangerous.

Table 17: English source text of sample document from WMT19 EN-DE test data (2).

HAT	Shark injures 13-jähriger auf Hummertauchen in Kalifornien Ein Hai attackierte und verletzte einen 13-jährigen Jungen Samstag, während er am Eröffnungstag der Hummersaison in Kalifornien für Hummer tauchte. Der Angriff ereignete sich kurz vor 7 Uhr in der Nähe von Beacon 's Beach in Encinitas. Tschad Hammel erzählte KSWB-TV in San Diego, dass er am Samstagmorgen etwa eine halbe Stunde lang mit Freunden tauche, als er hörte, dass der Junge um Hilfe schreibe und dann mit einer Gruppe überhäuft wurde, um ihn aus dem Wasser zu ziehen. Hammel sagte zunächst, dass er es nur als Aufregung ansehe, einen Hummer zu fangen, aber dann "merkte er, dass er gejagt habe:" Ich habe wenig! Ich habe wenig! "Seine ganze Krone wurde offen gerissen," sagte Hammel, dass er bemerkt habe, sobald er zum Jungen kam. "Ich habe alle gezwungen, aus dem Wasser zu kommen:" There' s a shark in the water! "Hammel fügte hinzu. Der Junge wurde in das Rady Children 's Hospital in San Diego gebracht, wo er in kritischem Zustand aufgeführt wird. Die Art von Hai, die für den Angriff verantwortlich ist, war unbekannt. Lifeguard Capt. Larry Giles sagte bei einer Medienmitteilung, dass ein Hai in der Gegend ein paar Wochen zuvor entdeckt worden sei, aber er sei nicht eine gefährliche Art von Hai. Giles fügte dem Opfer hinzu, dass er in seinem oberen Torso-Gebiet traumatische Verletzungen erlitten habe. Offiziere, die den Zugang zum Strand von Ponto Beach in Casablad für 48 Stunden in Swami' s in Ecinitas sperren, um Sicherheitsvorkehrungen zu treffen.
Target	Hai verletzt 13-jährigen Jungen beim Hummertauchen in Kalifornien Am Samstag griff ein Hai einen 13-jährigen Jungen an und verletzte ihn, während er in Kalifornien am Eröffnungstag der Hummersaison nach Hummern tauchte, sagten Beamte. Der Angriff fand kurz vor sieben Uhr morgens nahe dem Strand von Beacon in Encinitas statt. Chad Hammel sagte KSWB-TV in San Diego, er habe mit Freunden für eine halbe Stunde am Samstagmorgen getaucht, als er den Jungen um Hilfe schreien hörte. Er sei dann mit den anderen rübergepaddelt, um ihn aus dem Wasser zu retten. "Zuerst dachte ich, jemand freut sich, weil er einen Hummer gefangen hat", sagte Hammel. Aber dann bemerkte ich, dass jemand schrie: "Ich wurde gebissen!" Ich wurde gebissen! Sein ganzes Schlüsselbein wurde aufgerissen, sagte Hammel, er stellte dies fest als er zu dem Jungen kam. Ich schrie alle an, damit sie aus dem Wasser herauskommen: "Da ist ein Hai im Wasser!" sagte Hammel. Der Junge wurde ins Rady Children's Hospital in San Diego gebracht, wo sein Zustand als kritisch dokumentiert wurde. Die für den Angriff verantwortliche Haiart ist unbekannt. Rettungsschwimmer Kapitän Larry Giles sagte bei einer Medienbesprechung, dass ein Hai einige Wochen zuvor in der Gegend gesichtet worden war, aber es wurde festgestellt, dass es sich nicht um eine gefährliche Haiart handelt. Giles fügte im Oberkörperbereich seines Opfers traumatische Verletzungen hinzu. Beamte schlossen den Zugang zum Strand von Ponto Beach in Casablad zu Swami's in Ecinitas für 48 Stunden aus Sicherheitsgründen. Giles stellte fest, dass es mehr als 135 Haiarten in der Gegend gibt, aber die meisten gelten nicht als gefährlich.

Table 18: BLEU(HAT): 21.8

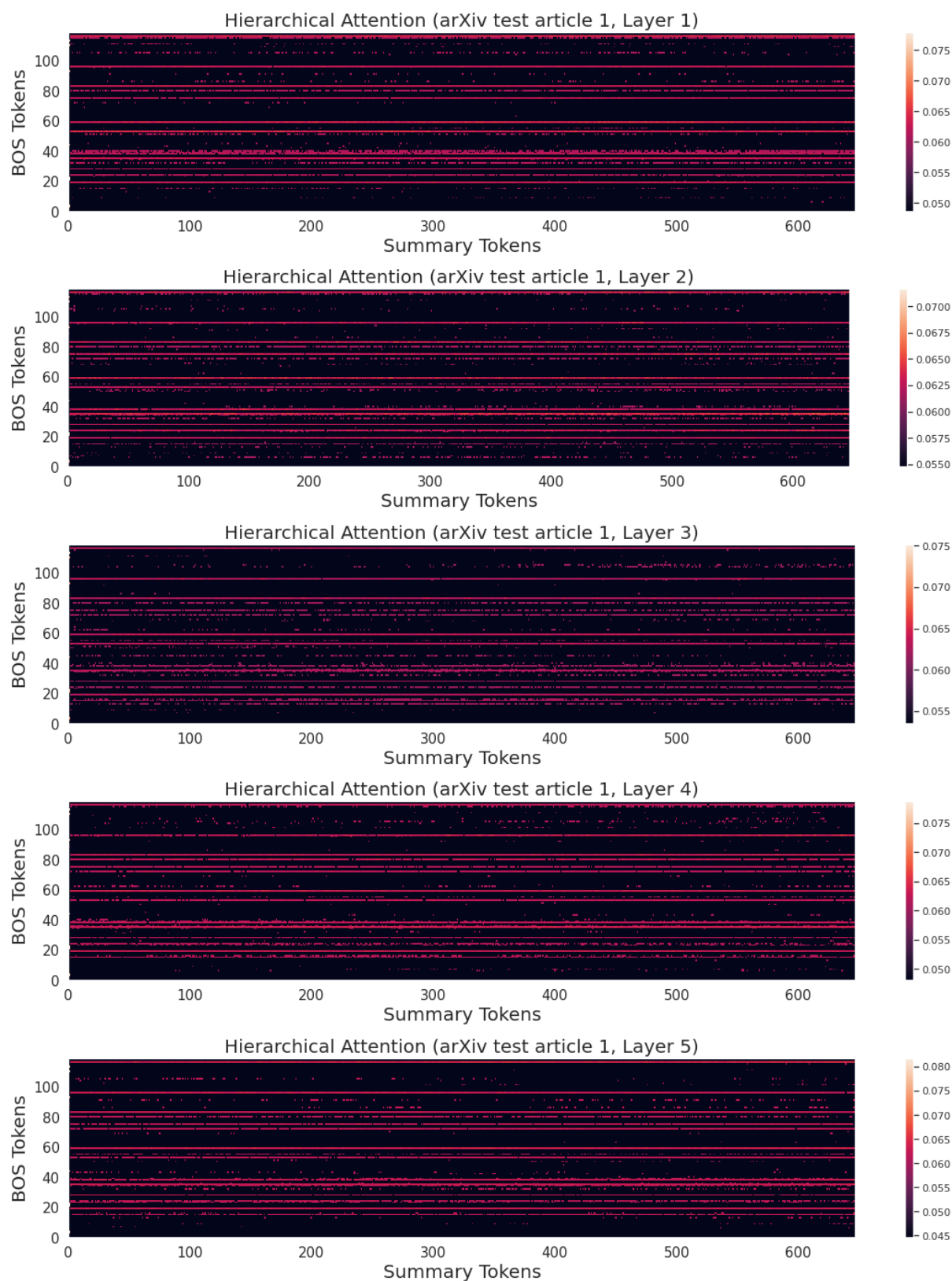
Air Traffic Controller Dies To Ensure Hundreds On Plane Can Escape Earthquake An air traffic controller in Indonesia is being hailed as a hero after he died ensuring that a plane carrying hundreds of people made it safely off the ground. More than 800 people have died and many are missing after a major earthquake hit the island of Sulawesi on Friday, triggering a tsunami. Strong aftershocks continue to plague the area and many are trapped in debris in the city of Palu. But despite his colleagues fleeing for their lives, 21-year-old Anthonius Gunawan Agung refused to leave his post in the wildly swaying control tower at Mutiara Sis Al Jufri Airport Palu airport. He stayed put to make sure that the Batik Air Flight 6321, which was on the runway at the time, was able to take off safely. He then jumped off the traffic control tower when he thought it was collapsing. He died later in hospital. Spokesman for Air Navigation Indonesia, Yohannes Sirait, said the decision may have saved hundreds of lives, Australia's ABC News reported. We prepared a helicopter from Balikpapan in Kalimantan to take him to a bigger hospital in another city. Unfortunately we lost him this morning before the helicopter reached Palu. "Our heart breaks to hear about this," he added. Meanwhile, authorities fear that the death toll could reach the thousands with the country's disaster mitigation agency saying that access to the towns of Donggala, Sigi and Boutong is limited. "The toll is believed to be still increasing since many bodies were still under the wreckage while many have not able to be reached," agency spokesman Sutopo Purwo Nugroho said. Waves that reached up to six meters have devastated Palu which will hold a mass burial on Sunday. Military and commercial aircraft are bringing in aid and supplies. Risa Kusuma, a 35-year-old mother, told Sky News: "Every minute an ambulance brings in bodies. Clean water is scarce. The mini-markets are looted everywhere." Jan Gelfand, head of the International Red Cross in Indonesia, told CNN: "The Indonesian Red Cross is racing to help survivors but we don't know what they'll find there. This is already a tragedy, but it could get much worse." Indonesia's President Joko Widodo arrived in Palu on Sunday and told the country's military: "I am asking all of you to work day and night to complete every tasks related to the evacuation. Are you ready?" CNN reported. Indonesia was hit earlier this year by earthquakes in Lombok in which more than 550 people died.

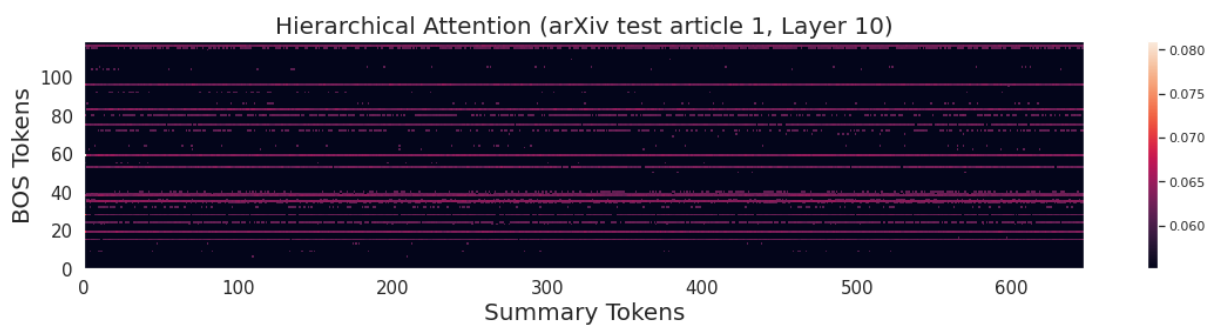
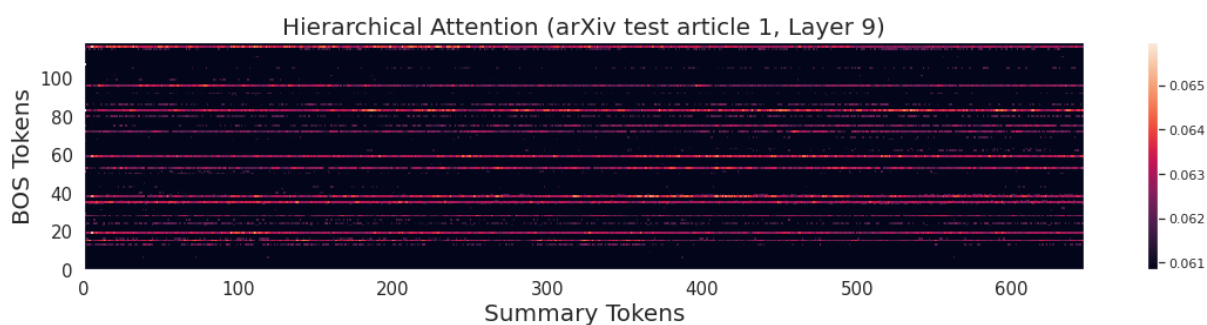
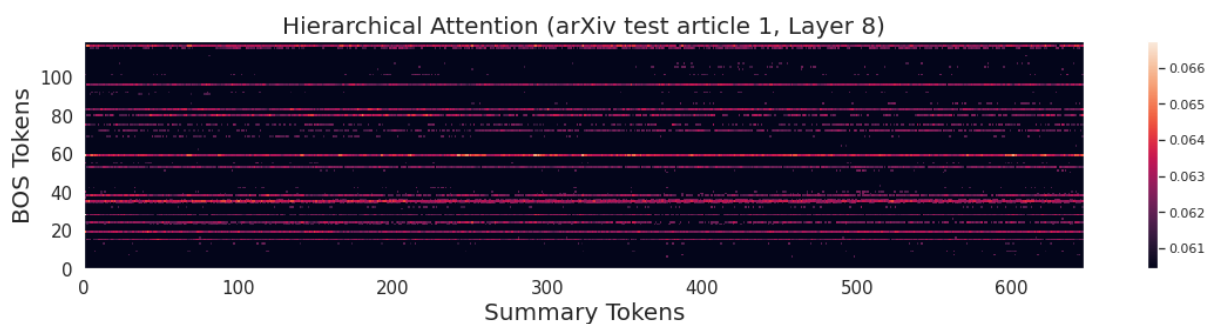
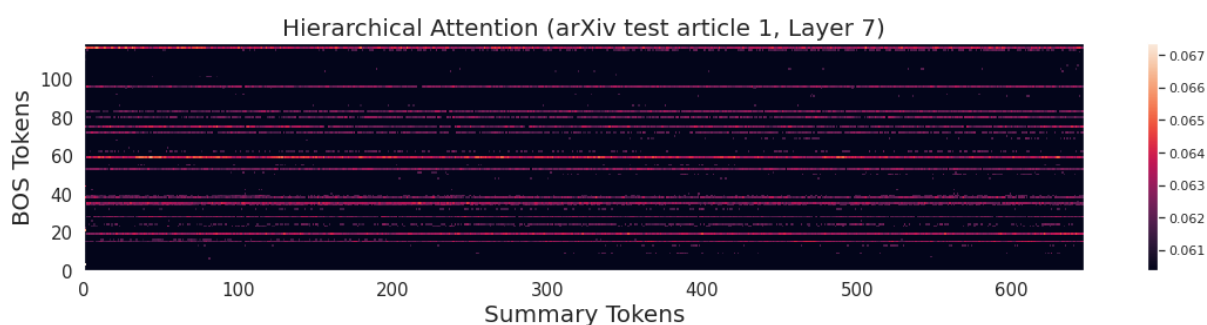
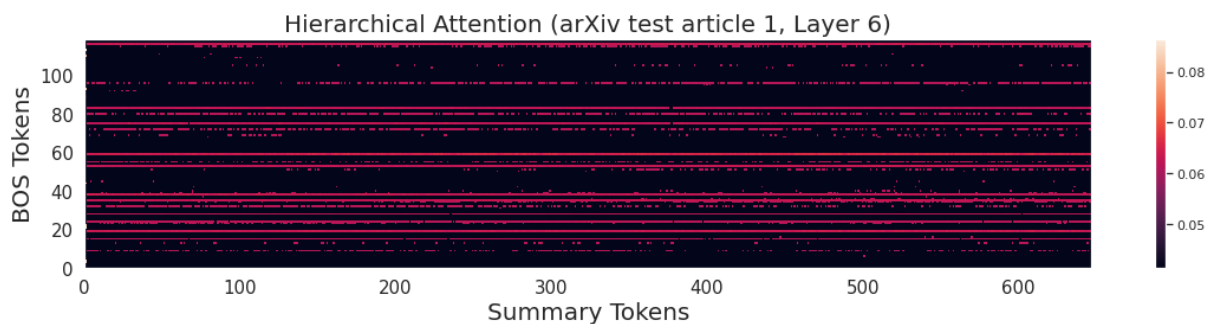
Table 19: English source text of sample document from WMT19 EN-DE test data (3).

HAT	<p>Air Traffic Controller Dies Um Hunderte Auf Plane Can Escape Erdbeben sicher zu machen Ein Air Traffic Controller in Indonesien wird als Held gefeiert, nachdem er starb, um sicherzustellen, dass ein Flugzeug mit Hunderten von Menschen es sicher vor dem Boden gemacht hat. Mehr als 800 Menschen sind gestorben und viele sind vermisst, nachdem ein großes Erdbeben die Insel Sulawesi am Freitag getroffen hat, was einen Tsunami ausgelöst hat. Starke Nachbeben plagen weiterhin das Gebiet und viele sind in Schutt in der Stadt Palu gefangen. Aber trotz seiner Kollegen, die für ihr Leben fliehen, weigerte sich der 21-jährige Anthonius Gunawan Agung, seinen Posten im wilden schwenkenden Kontrollturm auf dem Flughafen Mutiara Sis Al Jufri Airport Palu zu verlassen. Er blieb, um sicherzustellen, dass die Batik Air Flight 6321, die damals auf der Startbahn war, in der Lage war, sicher zu starten. Er sprang dann aus dem Kontrollturm, als er dachte, es stürzte. Er starb später im Krankenhaus. Spokesman für Air Navigation Indonesien, Yohannes Sirait, sagte, die Entscheidung könnte Hunderte von Leben gerettet haben, Australiens ABC News berichtet. Wir bereiten einen Hubschrauber von Balikpapan in Kalimantan, um ihn zu einem größeren Krankenhaus in einer anderen Stadt zu nehmen. Leider haben wir ihn heute Morgen verloren, bevor der Hubschrauber Palu erreichte. "Unser Herz bricht, um darüber zu hören," fügte er hinzu. Unterdessen fürchten die Behörden, dass die Zahl der Todesopfer die Tausenden erreichen könnte, mit der Behörde für Katastrophenabwehr des Landes, dass der Zugang zu den Städten Donggala, Sigi und Boutong begrenzt ist. "Die Zahl der Todesopfer wird immer noch steigen, da viele Körper noch unter dem Trümmerstand waren, während viele nicht in der Lage sind, erreicht zu werden," sagte der Sprecher der Agentur Sutopo Purwo Nugroho. Wellen, die bis zu sechs Meter erreicht haben, haben Palu verwüstet, die am Sonntag eine Massenattacke halten wird. Militärische und kommerzielle Flugzeuge bringen Hilfe und Lieferungen. Risa Kusuma, eine 35-jährige Mutter, sagte Sky News: "Jede Minute ein Krankenwagen bringt Körper. Sauberes Wasser ist knapp. Die Mini-Märkte werden überall geplündert Jan Gelfand, Leiter des Internationalen Roten Kreuzes in Indonesien, sagte CNN: "Das indonesische Rote Kreuz ist Rennen, um Überlebenden zu helfen, aber wir wissen nicht, was sie dort finden werden. Dies ist bereits eine Tragödie, aber es könnte viel schlimmer werden. "Indonesiens Präsident Joko Widodo kam am Sonntag in Palu an und erzählte dem Militär des Landes: "Ich bitte Sie alle, Tag und Nacht zu arbeiten, um alle Aufgaben im Zusammenhang mit der Evakuierung abzuschließen. Sind Sie bereit?" CNN berichtete. Indonesien wurde Anfang dieses Jahres von Erdbeben in Lombok getroffen, in dem mehr als 550 Menschen starben.</p>
Target	<p>Fluglotse stirbt, um sicherzustellen, dass Hunderte von Flugzeugen dem Erdbeben entkommen können. Ein Fluglotse in Indonesien wird nach seinem Tod als Held gefeiert, weil er dafür gesorgt hat, dass ein Flugzeug mit Hunderten von Menschen sicher auf dem Boden landen konnte. Mehr als 800 Menschen sind gestorben und viele werden vermisst, nachdem am Freitag ein schweres Erdbeben die Insel Sulawesi heimgesucht und einen Tsunami ausgelöst hat. Starke Nachbeben plagen das Gebiet weiterhin und viele sind in der Stadt Palu in Trümmern gefangen. Seine Kollegen liefen um ihr Leben, aber der 21-jährige Anthonius Gunawan Agung weigerte sich, seinen Posten im extrem schwankenden Kontrollturm am Flughafen Mutiara Sis Al Jufri Airport Palu zu verlassen. Er blieb an Ort und Stelle, um sich zu vergewissern, dass der Batik-Air-Flug 6321, der sich zum Zeitpunkt auf der Startbahn befand, sicher starten konnte. Er sprang dann vom Verkehrskontrollturm, als er dachte, dieser würde einstürzen. Er erlag später im Krankenhaus seinen Verletzungen. Der Sprecher der Air Navigation Indonesia, Yohannes Sirait, sagte, die Entscheidung habe wohlmöglich Hunderte von Leben gerettet, berichtete ABC-News in Australien. Wir haben einen Hubschrauber von Balikpapan in Kalimantan vorbereitet, um ihn in ein größeres Krankenhaus in einer anderen Stadt zu bringen. Leider haben wir ihn heute Morgen verloren, bevor der Hubschrauber Palu erreichte. "Wir waren zutiefst erschüttert, als wir davon hörten", fügte er hinzu. In der Zwischenzeit befürchten die Behörden, dass die Zahl der Todesopfer in die tausende gehen könnte, wobei die Katastrophenschutzbehörde des Landes sagt, dass der Zugang zu den Städten Donggala, Sigi und Boutong begrenzt ist. "Es wird angenommen, dass die Zahl noch weiter steigen wird, da viele Leichen noch unter dem Trümmern lagen, und viele nicht erreicht werden konnten," sagte Agentursprecher Sutopo Purwo Nugroho. Bis zu sechs Meter hohe Wellen haben Palu verwüstet, wo am Sonntag eine Massenbestattung stattfindet. Militärische und kommerzielle Flugzeuge liefern Hilfe und Vorräte. Risa Kusuma, eine 35-jährige Mutter, erzählte Sky News: "Jede Minute bringt ein Krankenwagen Leichen herein. Sauberes Wasser ist knapp. Die kleinen Geschäfte werden überall geplündert." Jan Gelfand, Leiter des Internationalen Roten Kreuzes in Indonesien, sagte CNN: "Das Indonesische Rote Kreuz arbeitet unermüdlich, um Überlebenden zu helfen, aber wir wissen nicht, was sie dort finden werden. Es ist bereits eine Tragödie, aber es könnte noch viel schlimmer werden." Indonesiens Präsident Joko Widodo ist am Sonntag in Palu eingetroffen und sagte dem Militär: "Ich bitte Sie alle, Tag und Nacht zu arbeiten, um alle Aufgaben im Zusammenhang mit der Evakuierung zu erfüllen. Sind Sie bereit?" berichtete CNN. Indonesien wurde in diesem Jahr von Erdbeben in Lombok heimgesucht, bei dem mehr als 550 Menschen ums Leben kamen.</p>

Table 20: BLEU(HAT): 48.0

C Appendix





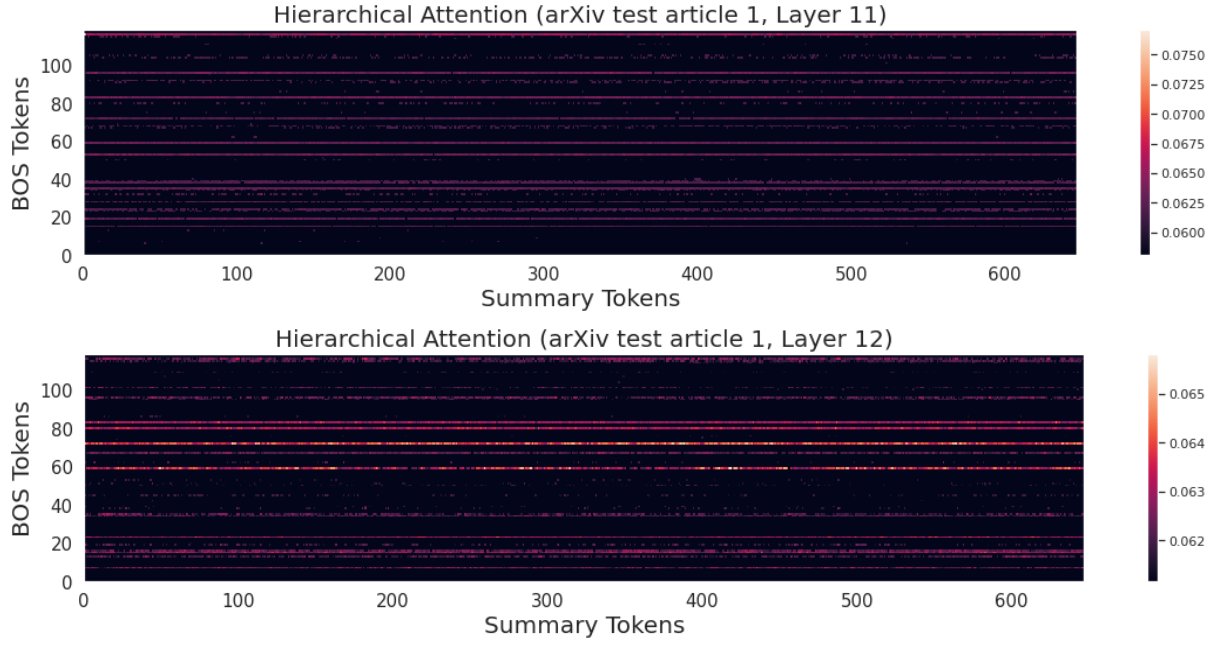
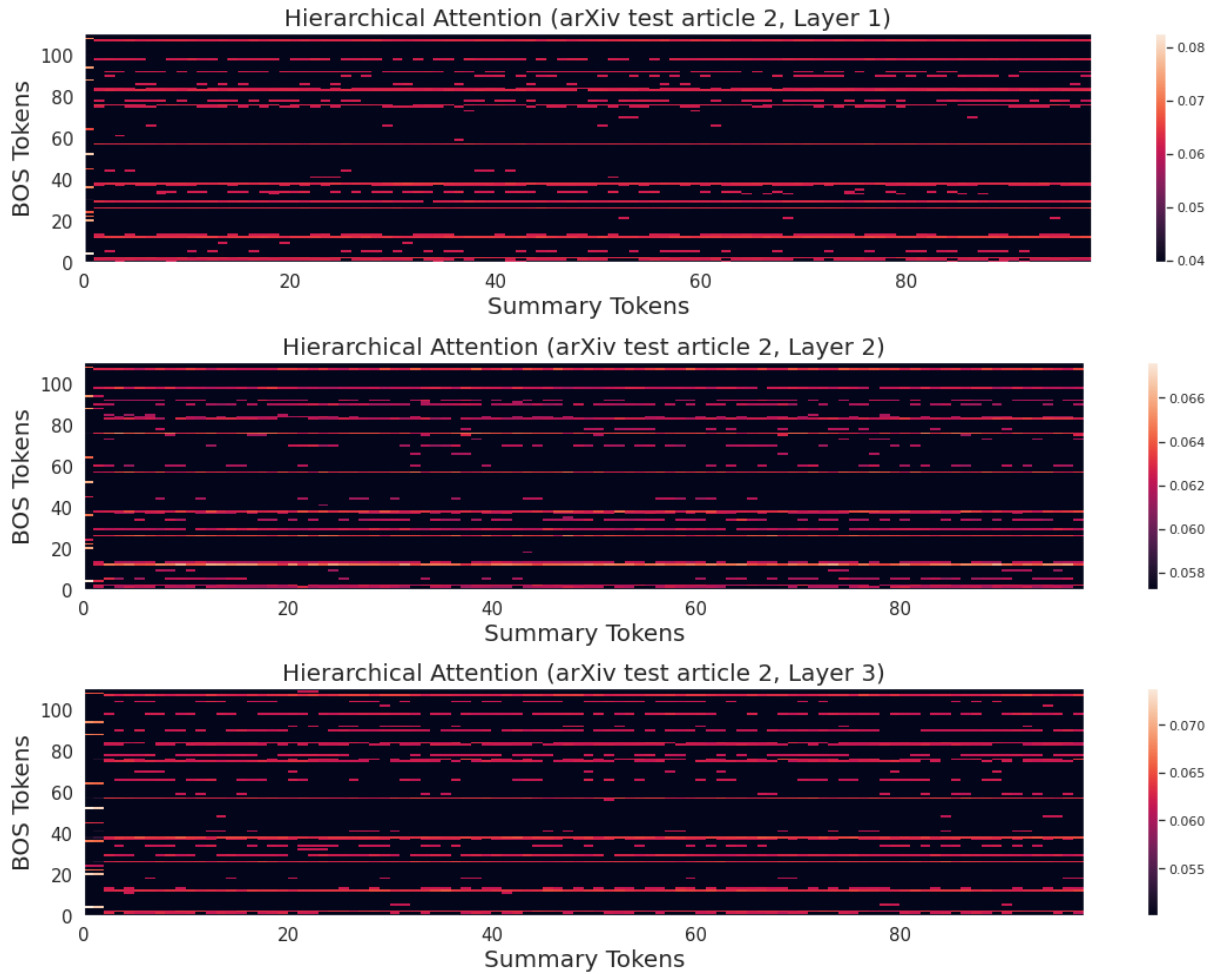
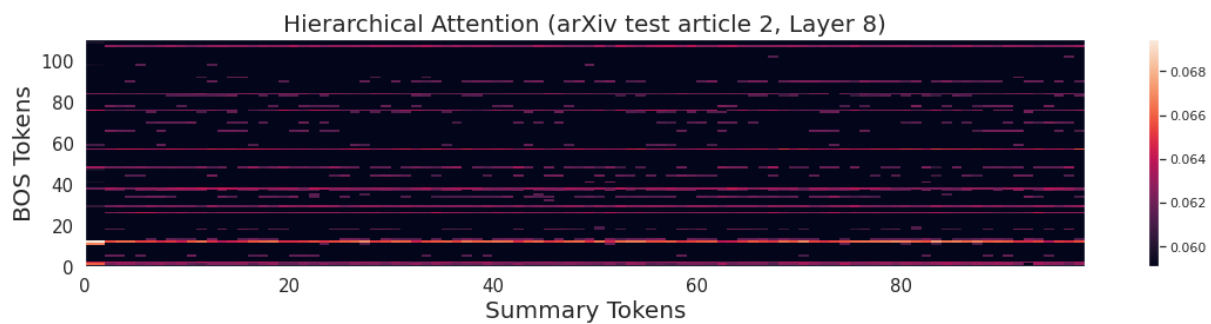
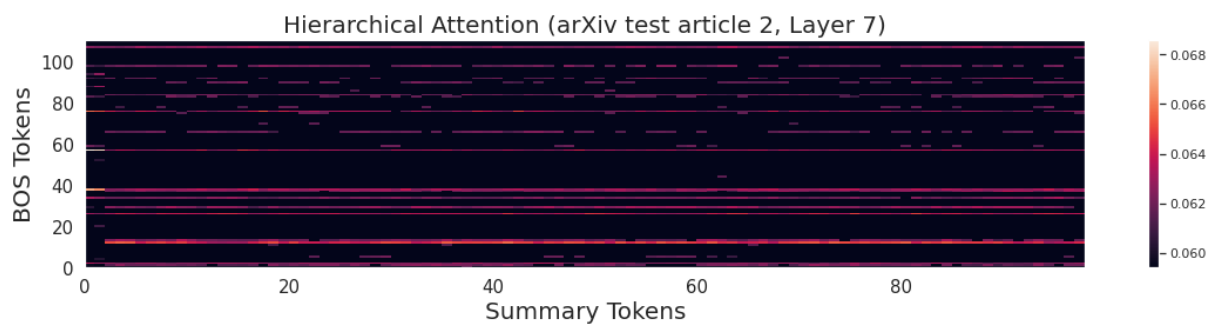
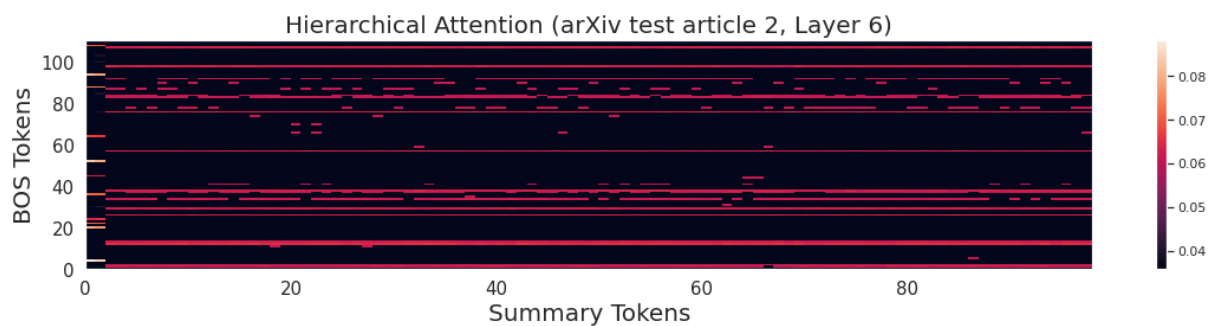
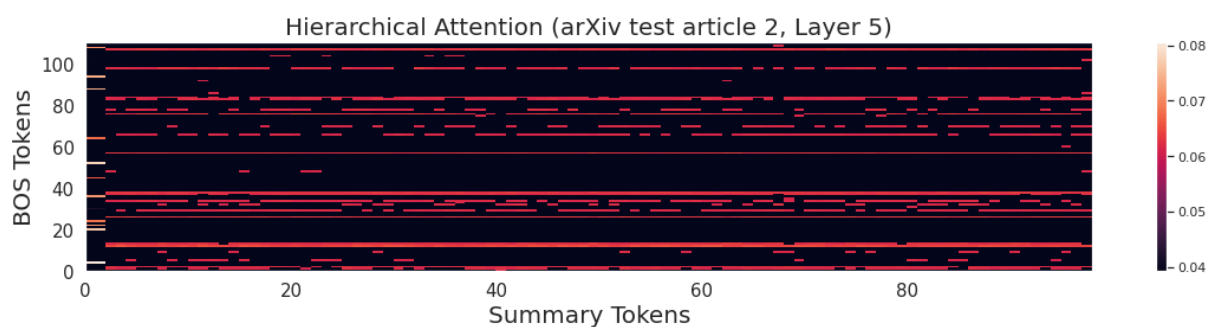
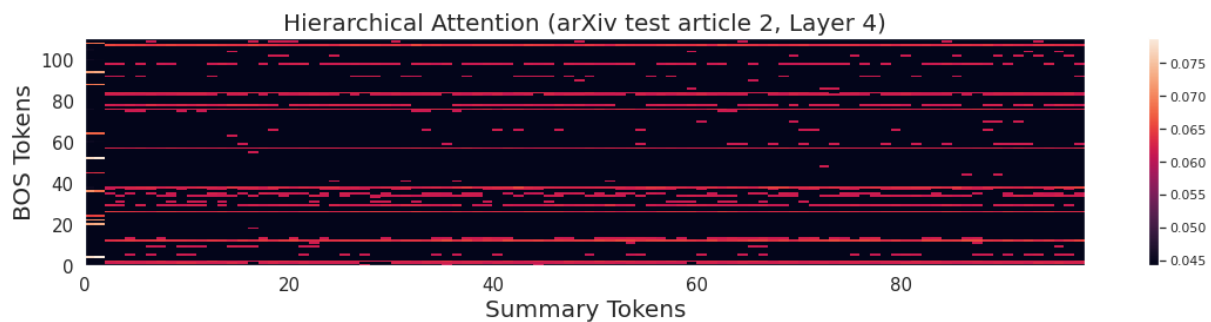


Figure 2: Hierarchical Attention heatmaps for arXiv test article 1 (Table 9 and 10). For each summary token, we only show the top 16 BOS tokens across each head.





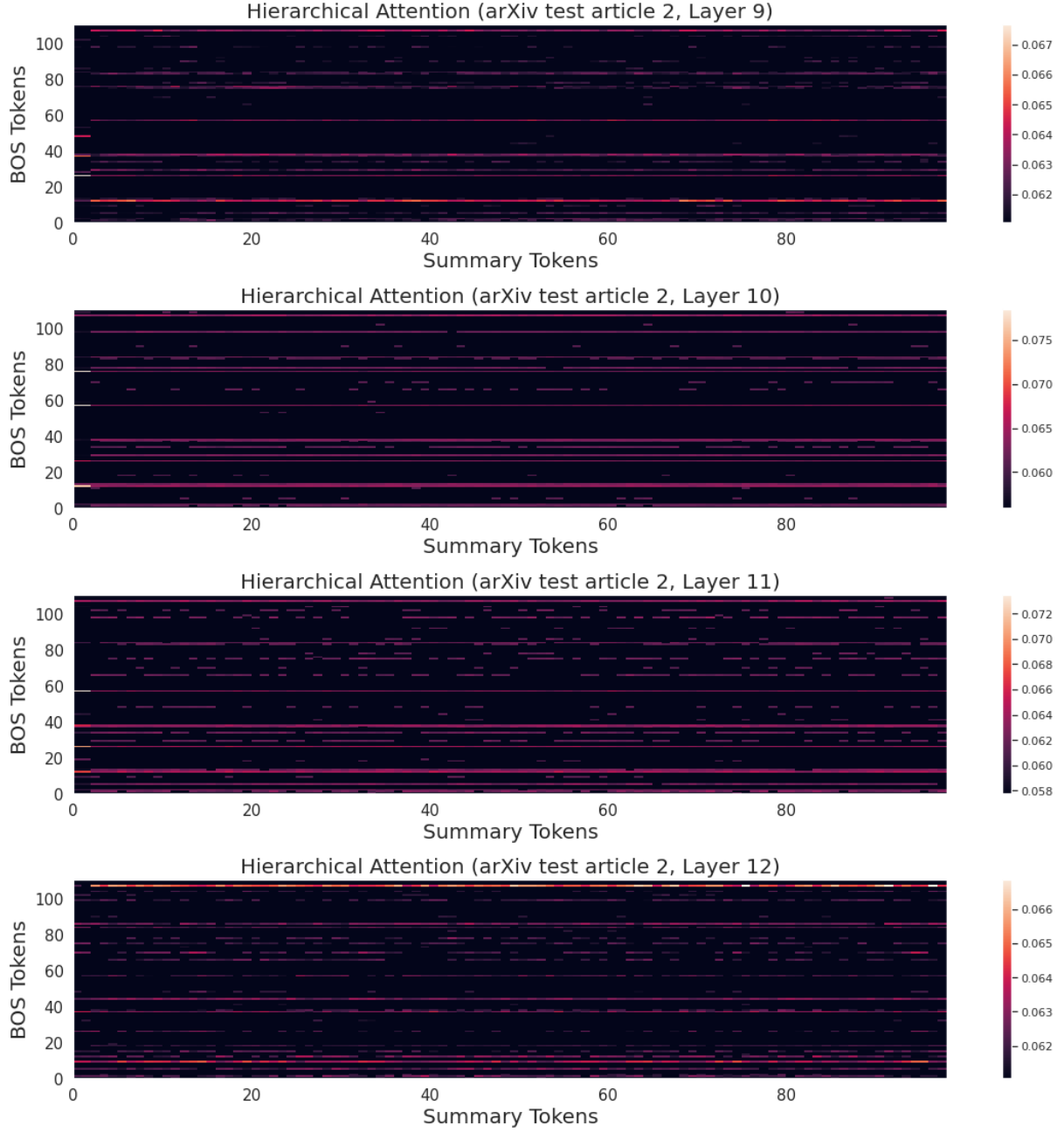
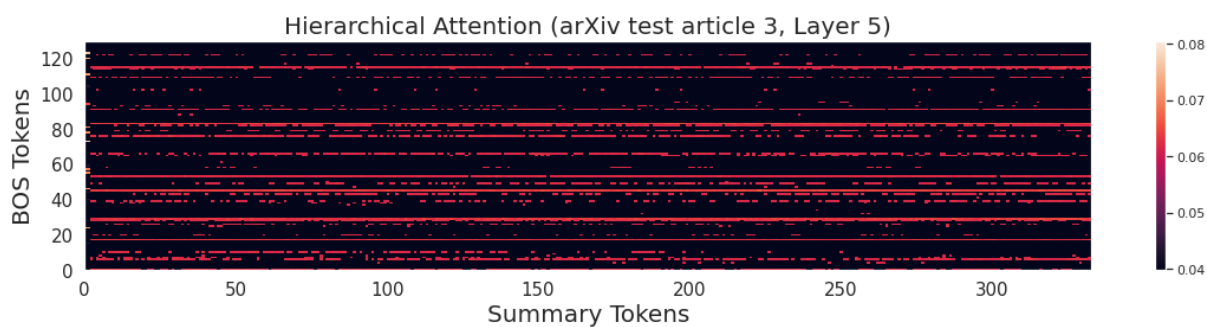
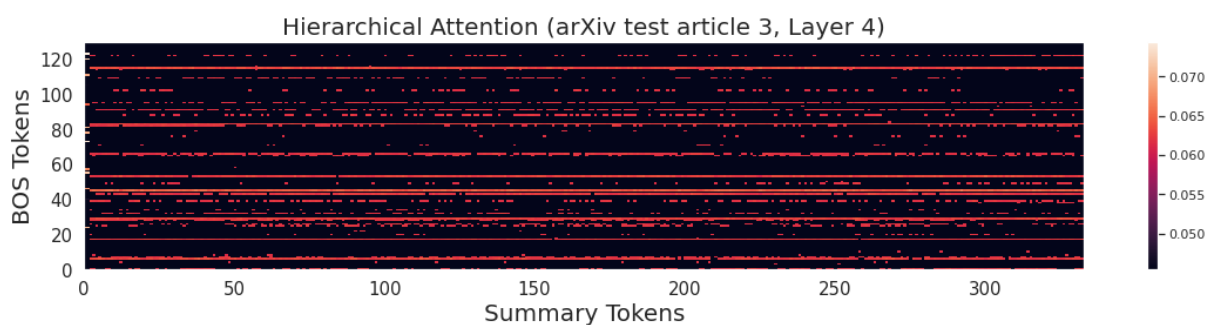
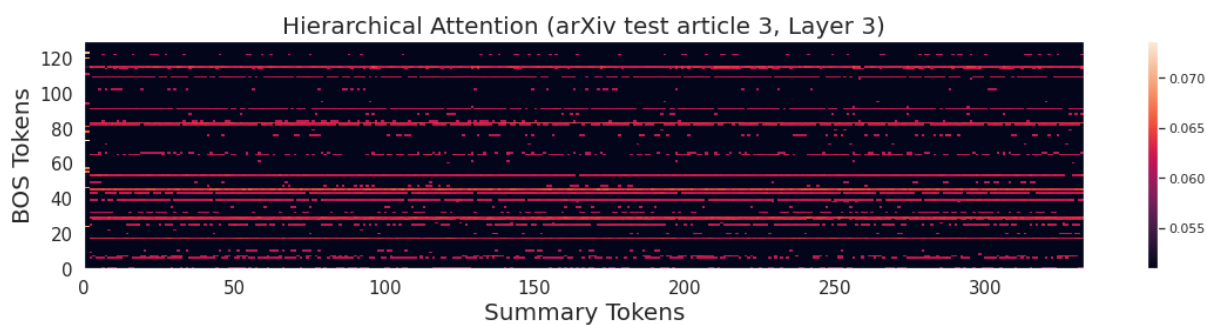
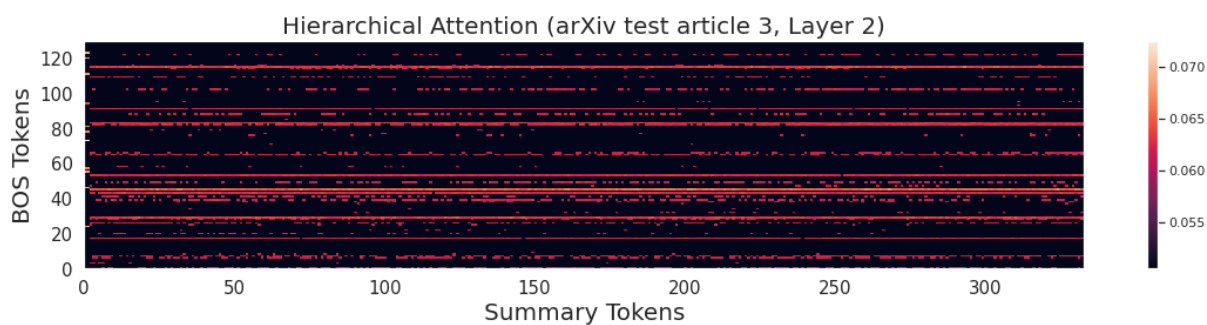
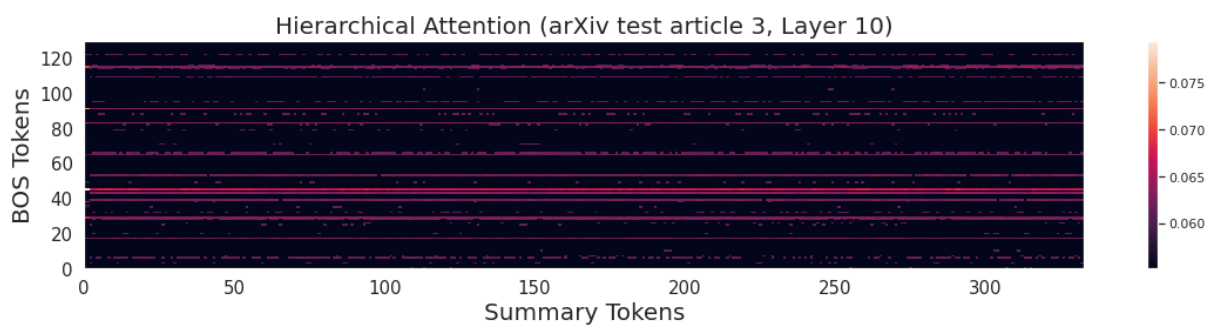
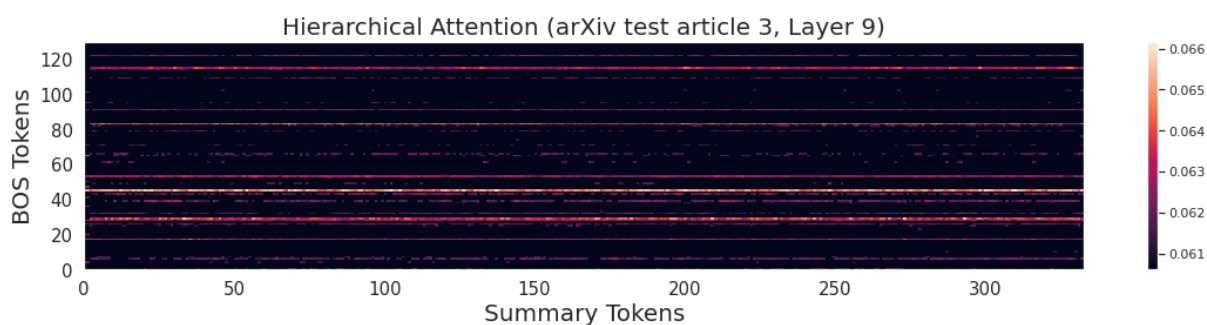
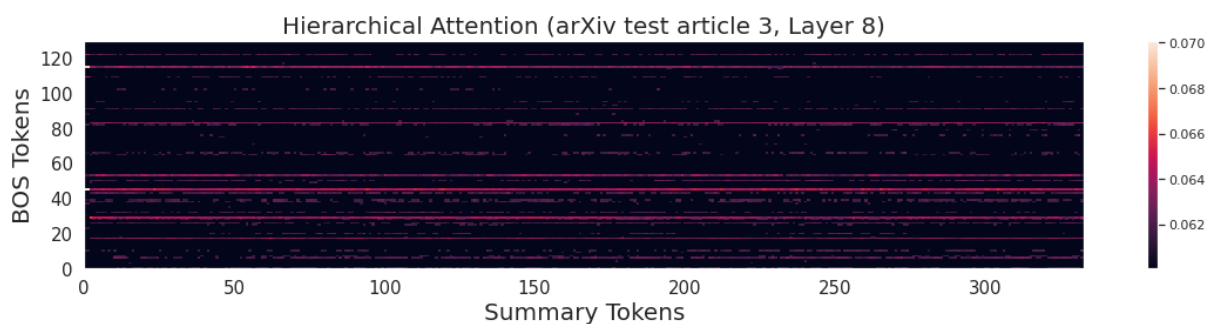
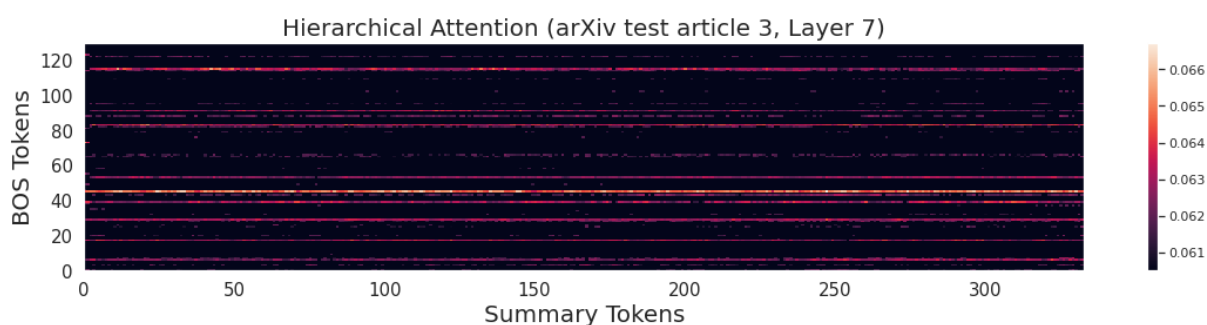
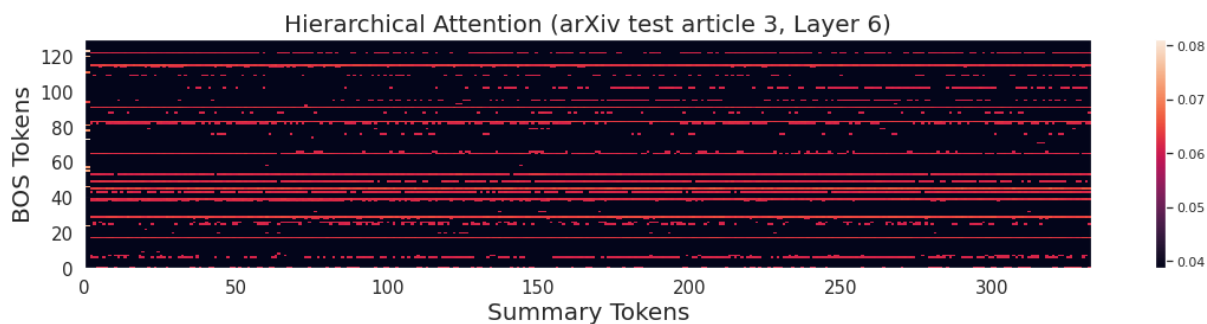


Figure 3: Hierarchical Attention heatmaps for arXiv test article 2 (Table 11 and 12). For each summary token, we only show the top 16 BOS tokens across each head.





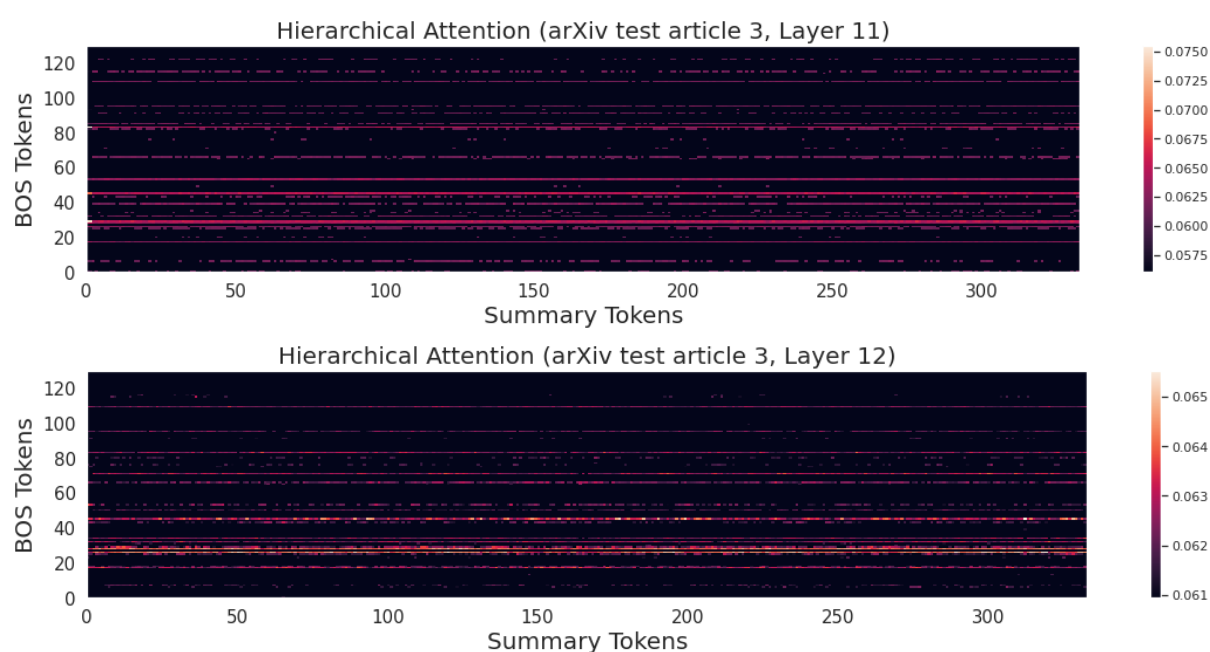


Figure 4: Hierarchical Attention heatmaps for arXiv test article 3 (Table 13 and 14). For each summary token, we only show the top 16 BOS tokens across each head.