# Beyond Paragraphs: NLP for Long Sequences

**Iz Beltagy**[†] **Arman Cohan**[†] **Hannaneh Hajishirzi**[‡] **Sewon Min**[‡] **Matthew E. Peters**[†]

[‡] Paul G. Allen School, University of Washington, Seattle, WA
[†] Allen Institute for AI, Seattle, WA

## 1 Introduction

A significant subset of natural language data includes documents that span thousands of tokens. The ability to process such long sequences is critical for many NLP tasks including document classification, summarization, multi-hop, and open-domain question answering, and document-level or multi-document relationship extraction and coreference resolution. These tasks have important practical applications in domains such as scientific document understanding and the digital humanities (Ammar et al., 2018; Cohan et al., 2018; Kociský et al., 2018; Lo et al., 2020; Wang et al., 2020a). Yet, scaling state-of-the-art models to long sequences is challenging as many models are designed and tested for shorter sequences. One notable example is transformer models (Vaswani et al., 2017) that have $O(N^2)$ computational cost in the sequence length $N$, making them prohibitively expensive to run for many long sequence tasks. This is reflected in many widely-used models such as RoBERTa and BERT where the sequence length is limited to only 512 tokens.

In this tutorial, we aim at bringing interested NLP researchers up to speed about the recent and ongoing techniques for document-level representation learning. Additionally, our goal is to reveal new research opportunities to the audience, which will hopefully bring us closer to address existing challenges in this domain.

We will first provide an overview of established long sequence NLP techniques, including hierarchical, graph-based, and retrieval-based methods. We will then focus on the recent long-sequence transformer methods, how they compare to each other, and how they can be applied to NLP tasks (see Tay et al. (2020) for a recent survey). We will also discuss various memory-saving methods that are key to processing long sequences. Throughout the tutorial, we will use classification, question answering, and information extraction as motivating tasks. In the end, we will have a hands-on coding exercise focused on summarization.[1]

## 2 Description

**Tutorial Content** This tutorial covers methods for long-sequence processing and their application to NLP tasks. We will start by explaining why processing long sequences is difficult. Many popular models scale poorly with the sequence length, either in computational or memory requirements, making them too expensive or impossible to run on current hardware. Another reason is that we want models that can capture long-distance information while ignoring large amounts of irrelevant text. The introduction also covers the tasks that we will use throughout the tutorial, namely information extraction (relation extraction (Jia et al., 2019) and coreference resolution (Pradhan et al., 2012; Bamman et al., 2020)), question answering (especially the multi-hop setting as in HotpotQA (Yang et al., 2018) and Wikihop (Welbl et al., 2018)), and document classification, and summarization.

The next section will review well-established methods for dealing with long sequences, namely chunking and graph based methods. Chunking refers to splitting the sequence into smaller chunks, processing each one independently, then aggregating them in a task-specific way (Joshi et al., 2019). Hierarchical models are a special case of chunking where the chunks are linguistic constructs (usually sentences) that are aggregated following the document hierarchy (Yang et al., 2016). Finally, retrieval-based methods use a recall-optimized simple model to retrieve short text snippets relevant for the task, then follow up with a stronger, more

[1] Slides and Code `https://github.com/allenai/naacl2021-longdoc-tutorial`

*Proceedings of NAACL-HLT 2021: Tutorials*, pages 20–24
June 6–11, 2021. ©2021 Association for Computational Linguistics

expensive model. Retrieval methods have been discussed in detail in the Open Domain QA tutorial (Chen and Yih, 2020) so we will cover it here very briefly. Graph-based methods will also be discussed, with a focus on question answering. These methods usually use local context to identify potentially relevant information across the document, heuristically connect the identified information in a graph, then apply a graph neural network (Kipf and Welling, 2017) to propagate information across the document between the snippets. This is particularly effective for the multi-hop reasoning setting (Fang et al., 2019).

Next, we will focus on the recent transformer-based methods for efficient processing of long sequences. The key question these models are addressing is how to perform the expensive $O(N^2)$ self-attention computation efficiently. All models make this computation faster by approximating the full self-attention leading to different models with different behaviors and applications. We will survey a few of the key papers summarized in Tay et al. (2020). In particular, we will talk about Transformer-XL (Dai et al., 2019), Longformer (Beltagy et al., 2020), Reformer (Kitaev et al., 2020) and Linformer (Wang et al., 2020b). We will also discuss how they apply to NLP tasks; Transformer-XL is mainly suitable for autoregressive tasks while the other three are equally suitable for autoregressive and bidirectional tasks. We will compare the performance of the other three models on various NLP tasks.

The next section discusses pretraining and finetuning of the transformer models. For pretraining, we will discuss different approaches to warm start the model weights from existing pretrained models for short sequences (Gupta and Berant, 2020; Beltagy et al., 2020). These approaches are versatile and make it possible to adapt most existing pretrained transformer models for short sequences into models that can process long sequences with a tiny pretraining cost. We will also demonstrate how to finetune such models for tasks such as question answering and classification.

The following section is a practical use case on summarization. We will show how to start from the BART (Lewis et al., 2020) checkpoint, convert it into a model that can work with a long input that's tens of thousands of tokens long, then finetune it on a long-input summarization task. It will also discuss practical techniques necessary to run the model on current hardware, including memory optimization techniques such as gradient checkpointing (Chen et al., 2016) and gradient accumulation. These are generic memory saving methods applicable to all neural models, and especially applicable in the long sequence setting.

Finally, the future work section will discuss open questions and future research directions like pretraining objectives that are better suited for long documents, encoder-decoder models with long output sequence, the balance between two-stage retrieval methods and single stage methods with long input, and how we think about long-sequence scaling for large models where the self-attention compute overhead reduces relative to feed-forward layers.

**Relevance to ACL** The models we cover are generic machine learning tools, but we discuss them from the NLP perspective, and study their application to core NLP tasks like IE, QA, and text generation. These methods have the potential to improve tasks that are currently challenging like multi-document summarization, story generation, and long dialogues. It can also enable new applications that have not yet been considered.

## 3 Type of the tutorial

This is a **cutting-edge** tutorial. The methods we discuss, especially the transformer-based and the graph-based methods, are active areas of research.

## 4 Outline

This tutorial will be 3 hours long.

1. **Introduction** (15 minutes long): This section will introduce the theme of the tutorial: why processing long sequence is important and why it is difficult. It will also introduce the NLP end-tasks that we will use throughout the tutorial.

2. **Chunking, hierarchical, and graph based methods** (35 minutes long): This section discusses graph-based methods and their application to information extraction and question answering, especially in the multi-hop reasoning setting. It also covers chunking and hierarchical methods as applied to coreference resolution, classification, and question answering.

3. **Transformer-based methods** (45 minutes long): This section reviews recently introduced long-sequence transformer models, compares the pros and cons of their designs, and discuss their applicability to NLP applications.

4. **Pretraining and finetuning** (25 minutes long): This section discusses how the long-sequence transformer methods are pretrained and how they are finetuned for downstream tasks including classification and question answering.

5. **Use Case: Summarization** (40 minutes long): This section is a practical exercise where we demonstrate in code how to build and train a long-document summarization model. It will also cover the technical details of multiple memory-saving methods that are key for training models on long sequences including gradient accumulation, and gradient checkpointing.

6. **Open problems and directions** (20 minutes long): In this final section, we will provide an outlook into the future. We will highlight both open problems and point to future research directions.

## 5 Breadth

We estimate 75% of the work covered will not be by the tutorial presenters.

## 6 Prerequisites

- Machine Learning: Basic knowledge of common recent neural network architectures like RNNs, and Transformers.

- Computational linguistics: Familiarity with standard NLP tasks such as text classification, natural language generation, and question answering.

## 7 Reading List

Reading the following papers is nice to have but not required for attendance.

- Hierarchical attention for classification (Yang et al., 2016)

- Graph network for question answering (Fang et al., 2019)

- Survey of long sequence transformers (Tay et al., 2020)

- Extractive/Abstractive summarization (Subramanian et al., 2019)

## 8 Instructors

In alphabetical order,

**Iz Beltagy** Iz Beltagy is a Research Scientist at AI2 focusing on language modeling, domain adaptation, and document-level understanding. His research has been recognized with a best paper honorary mention at ACL 2020. He worked as a Teaching Assistant at the University of Texas at Austin teaching computer science courses.
Email: beltagy@allenai.org
Homepage: beltagy.net

**Arman Cohan** Arman Cohan is a Research Scientist at AI2 focusing on representation learning and transfer learning methods, as well as NLP applications in scientific and health-related domains. His research has been recognized with a best paper award at EMNLP 2017, an honorable mention at COLING 2018, and Harold N. Glassman Distinguished Doctoral Dissertation award in 2019.
Email: armanc@allenai.org
Homepage: armancohan.com

**Hannaneh Hajishirzi** Hannaneh Hajishirzi is an Assistant Professor in the Paul G. Allen School of Computer Science & Engineering at the University of Washington and a Research Fellow at the Allen Institute for AI. Her research spans different areas in NLP, focusing on developing machine learning algorithms that represent, comprehend, and reason about textual data at large scale. Honors include the Sloan Fellowship, Allen Distinguished Investigator Award, Intel rising star award, multiple best paper and honorable mention awards, and several industry research faculty awards. She has given previous tutorials at top NLP conferences.
Email: hannaneh@cs.washington.edu
Homepage: homes.cs.washington.edu/~hannaneh/

**Sewon Min** Sewon Min is a Ph.D. student in the Paul G. Allen School of Computer Science & Engineering at the University of Washington, advised by Hannaneh Hajishirzi and Luke Zettlemoyer. Her research focuses on natural language understanding, question answering, and knowledge representation.

She is a co-organizer of the 3rd Workshop on Machine Reading for Question Answering at EMNLP 2021, Competition on Efficient Open-domain Question Answering at NeurIPS 2020, and Workshop on Structured and Unstructured KBs at AKBC 2020.
Email: sewon@cs.washington.edu
Homepage: shmsw25.github.io

**Matthew Peters** Matthew Peters is a Research Scientist at AI2 focusing on representation learning for NLP, transfer methods, and model interpretability. His research was awarded a best paper at NAACL-HLT 2018, and he gave a previous tutorial at NAACL-HLT 2019.
Email: matthewp@allenai.org
Homepage: scholar.google.com/citations?user=K5nCPZwAAAAJ

## 9 Estimated Attendance

Due to the broad appeal, we expect the tutorial to be well attended with around 150 people. This is especially the case for the long-sequence transformer methods because they open up pretrained models to applications that haven't been considered before. They are also easy to use, something that appeals to researchers and practitioners alike.

This tutorial has not been previously offered, but some of the methods have been covered before. In particular, retrieval-based methods have been covered in the Open-Domain QA tutorial at ACL 2020 (Chen and Yih, 2020), so we won't cover this topic and will refer the attendees to the previous tutorial.

## 10 Venue

The tutorial will be held at NAACL-HLT 2021.

## 11 Open Access

All the slides, video recordings, and software used for the tutorial will be publicly available.

## References

Waleed Ammar, Dirk Groeneveld, Chandra Bhagavatula, Iz Beltagy, Miles Crawford, Doug Downey, Jason Dunkelberger, Ahmed Elgohary, Sergey Feldman, Vu Ha, Rodney Kinney, Sebastian Kohlmeier, Kyle Lo, Tyler Murray, Hsu-Han Ooi, Matthew Peters, Joanna Power, Sam Skjonsberg, Lucy Wang, Chris Wilhelm, Zheng Yuan, Madeleine van Zuylen, and Oren Etzioni. 2018. Construction of the literature graph in semantic scholar. In *NAACL-HLT*.

David Bamman, Olivia Lewke, and Anya Mansoor. 2020. An annotated dataset of coreference in english literature. In *LREC*.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *ArXiv*, abs/2004.05150.

Danqi Chen and Wen-tau Yih. 2020. Open-domain question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 34–37, Online. Association for Computational Linguistics.

Tianqi Chen, Bing Xu, Chiyuan Zhang, and Carlos Guestrin. 2016. Training deep nets with sublinear memory cost. *arXiv preprint*, abs/1604.06174.

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 *NAACL-HLT*.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *ACL*.

Yuwei Fang, Siqi Sun, Zhe Gan, Rohit Pillai, Shuohang Wang, and Jing jing Liu. 2019. Hierarchical graph network for multi-hop question answering. *arXiv preprint*, abs/1911.03631.

Ankit Gupta and Jonathan Berant. 2020. Gmat: Global memory augmentation for transformers. *ArXiv*, abs/2006.03274.

Robin Jia, Cliff Wong, and Hoifung Poon. 2019. Document-level n-ary relation extraction with multiscale representation learning. 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)*.

Mandar Joshi, Omer Levy, Luke Zettlemoyer, and Daniel Weld. 2019. BERT for coreference resolution: Baselines and analysis. In *EMNLP-IJCNLP*.

Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. *ICLR*.

Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. 2020. Reformer: The efficient transformer. In *ICLR*.

Tomás Kociský, Jonathan Schwarz, P. Blunsom, Chris Dyer, K. Hermann, Gábor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pretraining for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. 2020. S2ORC: The semantic scholar open research corpus. In *ACL*.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Joint Conference on EMNLP and CoNLL - Shared Task*, pages 1–40, Jeju Island, Korea. Association for Computational Linguistics.

Sandeep Subramanian, Raymond Li, Jonathan Pilault, and C. Pal. 2019. On extractive and abstractive neural document summarization with transformer language models. *ArXiv*, abs/1909.03186.

Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. 2020. Efficient transformers: A survey. *ArXiv*, abs/2009.06732.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

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

Sinong Wang, Belinda Li, Madian Khabsa, Han Fang, and Hao Ma. 2020b. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*.

Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. 2018. Constructing datasets for multi-hop reading comprehension across documents. *TACL*, 6:287–302.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *EMNLP*.

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, San Diego, California. Association for Computational Linguistics.