Thank you very much for your careful, insightful and valuable comments, we will explain your concerns point by point.

Common questions: 1. Qualitative Examples. (a) A hard unsupervised training case. (b) Updated Figure 5.

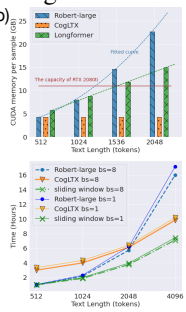
Score: Highest scoring blocks by judge; Score: Marked as irrelevant; Score: Marked as relevant

(a) An unsupervised example in 20news class "soc.religion.christian"

		Ep1	Ep2	Ep3
(1)	Harrassed at work, could use some prayers =CSE Dept., U.C. San…		0.16	0.19
(2)	Yesterday I counted and realized that on seven different occasions…		0.16	0.16
(3)	If he/she does not seem to take any action, keep going up higher ..		0.12	0.14
(4)	If you feel you can not discuss this with your boss, perhaps your …			0.13
(5)	It is unclear from your letter if you have done this or not. It is not …	0.01		0.13
(6)	If the company indeed does seem to want to ignore the entire…	0.01		
(7)	That is, someone that is supportive, comforting, etc. … healing…	0.01		0.09
(8)	No one could be bothered to call me at the other building, even …	0.01	0.13	
(9)	People in offices tend to be more insensitive while working than …	0.01	0.12	0.08
(10)	Moderator allows me this latest indulgence. Well, if you can't turn …	0.01		
(11)	Then they will come back and wonder why I didn't want to go …	0.01	0.14	
(12)	They are doing it because they are still the playground bully …			
(13)	In MY day, we had to make do with 5 bytes of swap...		0.13	

All blocks are initialized as "irrelevant" by BM25 (no common words with the label "soc.religion.christian"). In the 1st epoch, the judge is nearly untrained and selects some blocks at random. Among them, (7) contributes most to the correct classification, thus is marked "relevant". In the 2nd epoch, trained judge finds (1) with strong evidence "prayers" and (1) is marked as "relevant" at once. Then in the next epoch, (7) becomes not essential for classification and is marked as "irrelevant".

(b)



2. Detailed discussion about related works. **#1**: Reformer uses LSH for content-based grouping attention, but it is not friendly to GPU, weakens position relevance and may tend to be finetuned to a local-minimum grouping. It still needs verification for BERTs. Longformer mixes global and window attention. It (O($L \log L$) space) is contemporaneous with CogLTX (O(1) space) but ArXived in advance. It performs similar to CogLTX on HotpotQA (69.5 vs 69.2). Its window size is 512 and most HotpotQA samples < 2,048, so whether faraway sentences in longer texts can fully interact really via global attention is still unknown, but CogLTX **can seamlessly combine it ("Orthogonal")** to handle longer texts. **#2**: The structure-based contexts (ECML'19) is indeed relevant, but it relies on Metadata and 3 manual cases, and CogLTX is more general. We will discuss it in the camera-ready version. **#3**: Yes, ORQA did partly inspire this work, but it focuses on retrieval via BERT embeddings (fast), while CogLTX is for long texts in reader period (fine-grained).

To Reviewer#1: 1. About the assumption. We agree there are some tasks violating the assumption, but to the best of our knowledge, the assumption can be applied to **most** of the common NLP tasks (paper Figure 2) for long texts, including summarization. One of the most popular summarization setting is "extractive summarization"[1], aiming to select important sentences to form a summarization. "Abstractive summarization" also mainly used key sentences[2]. [1] Text summarization with pretrained encoders. [2] Bottom-up abstractive summarization. EMNLP'18.

2. Time consumption for batch size > 1. See Figure (b). The time of batch size = 8 shows a similar trend as = 1 (the same total number of samples). We also compare the space of Longformer, which is still much heavier than CogLTX.

3. No enough details to fully reproduce. The main concerns are about details for unsupervised mode. We will definitely add details and polish the writing in the camera-ready version. The codes will be open-sourced too.

To Reviewer#2: 1. Expensive trial-and-error search without labels. Not so much. The trials only need "model inference" and are gradient-free, which is much faster than training with data-flow graph (Algo 1 Line 19). In experiments, it only cost $\sim 2\times$ time of training with labels, instead of N$\times$ time (N is the number of blocks).

2. Explain sufficient condition in Eq(6). This means some key sentences $\mathbf{z}$ are *enough* for the task, more sentences are useless(won't reduce the loss). See Figure (a) for case study.

3. Memory concerns during initally judging z^+. This is in the "model inference" period, when memory issue is not so serious(in training, sentences are separated with their relevance labels as different samples). We can also split them into different batches in the *retrieval competition* step to keep fixed memory overhead.

To Reviewer#3: 1. Explain details about relevant scores. Yes, they are binary and updated by *intervention*, a.k.a. removing it from $\mathbf{z}$ and to see the change of loss (Line 139 and Algo1 Line 18-21), which is fast (Reviewer#2 **1.**).

To Reviewer#4: CogLTX is a **general** framework to apply BERTs to arbitrarily long texts **without memory concerns** and retain the long-distance attention.

1. Comparison with SAE. (1) CogLTX is more **general** than SAE, which is specific to HotpotQA. It **cannot** be used for other tasks, e.g. classification, let alone **unsupervised** cases. SAE selects top 2 paragraphs because the HotpotQA is constructed by 2 of 10 paragraphs. It uses an extra GCN on graphs built with 3 kinds of co-occurrence of entities from spaCy NER package, with a module for Yes/No over the GCN. These designs are hard to be applied to other datasets. Different from SAE, CogLTX concatenates the paragraphs as a normal document into a universal pipeline without extra tricks. **(2)** SAE **does not** completely solve the memory problem. Actually, SAE-large needs V100 or better GPUs to train HotpotQA, and could raise OOM for longer or more paragraphs. See answer **3.** for further discussion.

2. Weak BERT backbone for baselines on 3 tasks. This might be a misunderstanding. We did use RoBERTa for all baselines (described in Line 184, Line 253), except the baseline in Task 3, whose results are from the original paper.

3. Discussion on Model over BERT. Some concerns might origin from the comparison with "Model over BERT" baselines, cutting documents into segments and aggregating BERT results by another model. They **don't** really solve the memory problem, but **sacrificing early interactions** (Line 69)(7 such methods worse than CogLTX in HotpotQA). End2end training needs O($512^2 \cdot L/512$) = O($512L$) space. It usually only improves the max length $2\times \sim 4\times$(depend on device and model size) for batch size = 1 and less for batch size > 1. As an advantage, CogLTX and sliding window only need **constant space**, especially fit for real-world data. Besides, "Model over BERT" mainly optimizes classification. Other tasks, like *span extraction*, has L BERT outputs, still need O(L^2) space for self-attention aggregation.