We would like to thank the reviewers for their valuable comments. Below, we address the concerns raised by the reviewers. We will make sure to reflect the *comments, suggested references, and our answers* to the final version.

Response to Reviewer 1. ***Q1-1. Number of layers required?*** In § D of the supplementary material, Lemmas 6–8 show that we need $\frac{dn}{\delta}$ sparse Transformer blocks (eq (2)) for quantization, $\frac{p(n-1)}{\delta^d}+s$ for contextual mapping, and $\frac{n}{\delta^{dn}}$ for value mapping. Recall that p and s are from Assumption 1, and δ is from Step 1 of § 4.1. In comparison, § C of [28] shows that the dense counterpart requires $\frac{dn}{\delta}$, $\frac{n}{\delta^d}+1$, and $\frac{n}{\delta^{dn}}$ Transformer blocks (eq (1)) for the three corresponding lemmas. Note two observations: ***1)*** *The value mapping* ***dominates*** *the depth, and its depth requirements are* ***identical*** *for the two cases*; and ***2)*** *For contextual mappings (where the attention layers are used), we need roughly* ***p times more*** *layers for sparse models.* Since p is usually a small constant, these observations mean that sparse Transformers can achieve universal approximation using depth of the **same order** in d and n as the dense Transformers.

Q1-2. Re: comments on clarity: Thank you for the suggestions. In our revision, we plan to add more details of the assumptions/proofs/references, especially Step 2 of § 4.1. We believe that this will also help remedy Weakness 2 and better motivate Assumption 1.2. Please also see Q2-2 on the necessity of assumptions. In Line 297, we use 1–4 sparse attention layers and one feed-forward layer (please see § G.1). In Line 265, we meant "all the other tokens."

Response to Reviewer 2. ***Q2-1. The result is not surprising, given arbitrary depth:*** Our result is *not* a straightforward extension of [28], as we illustrate in § 4.2; we overcome nontrivial challenges posed by sparsity, which is also appreciated by Rev1. Although a "connection" between any two positions can happen through multiple sparse layers, this is only an intuition, and turning it into a rigorous analysis is not easy. Moreover, there are results showing that *limited width* can render universal approximation *impossible*, **even with arbitrary depth**: see e.g., "Deep, skinny neural networks are not universal approximators." We would like to emphasize that our paper is the first to provide a concrete theory justifying sparse attention; our careful analysis reduces the connections per layer from n^2 to $O(n)$, with only p times more attention layers (see Q1-1). Our analysis also gives insights into the design of the sparsity patterns.

Q2-2. Necessity of assumptions? We believe that the assumptions are quite reasonable, as also mentioned by Rev1 and Rev4. Our assumptions are weak enough to be satisfied by many existing sparsity patterns, as we cover in § 3.4. In fact, we can show that Assumptions 1.1 and 1.3 are ***necessary*** for universal approximation to hold. For A1.3, assume $n=2$ and consider a sparsity pattern with $p=1$: $\mathcal{A}_1^1=\{1,2\}, \mathcal{A}_2^1=\{2\}$. Note that it satisfies A1.1 and A1.2, but not A1.3. Since the second token never attends to the first token, this sparse Transformer can never approximate a function whose second output token is dependent on both input tokens; this proves the necessity of A1.3. Next, consider $n=2$ and a pattern with $p=2$: $\mathcal{A}_1^1=\{1,2\}, \mathcal{A}_2^1=\{1,2\}, \mathcal{A}_1^2=\{1\}, \mathcal{A}_2^2=\{1\}$. One can check that this pattern satisfies all assumptions but A1.1. Since both tokens in the second layer attend to the same single token ($\mathcal{A}_1^2=\mathcal{A}_2^2=\{1\}$), the two tokens in the sequence become identical afterwards, and hence cannot approximate arbitrary functions. As per Rev2's suggestions on empirical verification, we will add experiments to further validate our assumptions; please also see Q4-1.

Q2-3. Extension to the encoder-decoder attention? For now, our analysis applies to the encoder (i.e. BERT-style) part of the model. Extending this analysis to the encoder-decoder attention would be a very interesting future direction.

Response to Reviewer 3. Thanks for the questions. For the hardness of the proof, please refer to Q2-1. Below, we address the concerns raised; we hope that our answers will clarify the proof and Rev3 will reassess our paper.

Q3-1. Concern 1) Both properties are used. § E.3 uses the fact that there are $n|\mathbb{H}_\delta| = n\delta^{-dn}$ distinct real numbers that $q(\boldsymbol{H})_k := \boldsymbol{u}^T g_c(\boldsymbol{H})_k$ takes (for positions $k \in [n]$ and contexts $\boldsymbol{H} \in \mathbb{H}_\delta$), which is implied by *both* Properties 7.1 & 7.2.

Q3-2. Concern 2) In case of $d=1$, Lemma 8 uses Properties 7.1 & 7.2 to construct a function g_{v} that maps all $n\delta^{-n}$ distinct values of $y_k := g_c(\boldsymbol{H})_k \in \mathbb{R}$ to $z_k := \overline{f}(\boldsymbol{H}-\boldsymbol{E})_k \in \mathbb{R}$. The reason why y_k must be distinct for different k's is that the feed-forward layers operate in a *position-wise* manner, hence so does g_{v}; in other words, the same map $g_{\mathrm{v}}^{\mathrm{tkn}}$ is applied to each token. For example, if $y_1=y_2=c$, then the first two tokens of the output $g_{\mathrm{v}}(\boldsymbol{y})$ must be identical because $g_{\mathrm{v}}(\boldsymbol{y})_1 = g_{\mathrm{v}}^{\mathrm{tkn}}(c) = g_{\mathrm{v}}(\boldsymbol{y})_2$. As a result, having $y_1=y_2$ prevents us from representing any arbitrary $\overline{f}$.

Response to Reviewer 4. ***Q4-1. Experiment section is not connected to the theoretical part?*** We appreciate the suggestions on experiments by Rev2/Rev4, and we plan to supplement the paper with more experiments on the assumptions and the BERT architecture in the revision. As for the validity of our assumptions, we note that the poor performance of the RANDOM pattern partially accounts for the necessity of Assumption 1, because it is very unlikely to satisfy Assumption 1 with random sparse connections. For a theoretical discussion on the necessity, please see Q2-2.

Q4-2. Masking and prediction in copying task? We implemented it as a masked-LM style prediction task by masking all the tokens in the second half of the sequence. For the test examples, each masked token is predicted independently. For the reported results we used autoregressive models as in LM. We re-ran experiments with bidirectional models and observed improved performance (e.g., STAR & 4-layer: 31.19% $\rightarrow$ 83.57%); we plan to report it in the final version.

Q4-3. Other questions: In LM with STAR pattern, tokens cannot attend to the last relay token. We'll add BERT experiments to make fairer comparisons. In MT, we use the encoder-decoder architecture, and sparsity for both parts.