Which Transformer architecture fits my data?
A vocabulary bottleneck in self-attention: Supplementary Material

Noam Wies[1] Yoav Levine[1] Daniel Jannai[1] Amnon Shashua[1]

Contents

[1] The Hebrew University of Jerusalem. Correspondence to: Noam Wies <noam.wies@cs.huji.ac.il>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

1. Upper bounds on the separation rank

In the following section, we show how an upper bound on the separation rank is implied by the rank of embedding.

1.1. Preliminaries

We will use the notation of $\left(\!\!\binom{n}{k}\!\!\right)$ – the multiset coefficient, given in the binomial form by $\binom{n+k-1}{k}$. We will use the identity $|\{a_1 \dots a_n \in \mathbb{Z} \geq 0 : \sum_{r=1}^{n} a_r = k\}| = \left(\!\!\binom{n}{k}\!\!\right)$. In addition, we will use the following two lemmas from (Levine et al., 2020) regarding the composition of L self-attention layers, and inequality of arithmetic and geometric multiset coefficient means.

Lemma 1. *Defining $C(L) \coloneqq \frac{3^L-1}{2}$, any depth L composition of self-attention layers defined in eq. 5 of the main text can be written as:*

$$y^{i,L,d_x,H}\left(y^{0,1},...,y^{0,N}\right) = \sum_{j_1,...,j_{C(L)}=1}^{N} \sum_{h\in[H]^{[C(L)]}} \sum_{r_1,...,r_{C(L)+1}=1}^{d_a} B_{r_1,p}^{(0,h)} \left(\prod_{c=1}^{C(L)+1} \left\langle A_{r_c}^{(c,h)}, \mathbf{y}^{0,j_c}\right\rangle\right) \left(\prod_{c=1}^{C(L)} \left\langle B_{r_{c+1}}^{(c,h)}, \mathbf{y}^{0,j_c}\right\rangle\right) \tag{1}$$

Where $\forall h \in [H]^{[C(L)]}$ $1 \leq c \leq C(L)+1$ $\quad A^{(c,h)}, B^{(c,h)} \in \mathbb{R}^{d_a \times d_x}$ and for convenient $j_{C(L)+1} \coloneqq i$.

Lemma 2. *Let $n, k \in \mathbb{N}$ and $\phi : \mathbb{N}^k \to \mathbb{N} \coloneqq r_1, \dots r_k \vdash \prod_{j=1}^{k}\left(\binom{n}{r_j}\right)$ then:*

$$\forall r_z, \dots r_k \in \mathbb{N} \quad \phi(r_1, \dots r_k) \leq \frac{\left(\prod_{t=1}^{n-1}\left(\frac{M}{k}+t\right)\right)^k}{((n-1)!)^k}$$

where $M \coloneqq \sum_{j=1}^{k} r_j$

Finally, we will use the following lemma to upper bound the multiset coefficient:

Lemma 3. $(\binom{n}{k}) \leq \left(\frac{2e(n+k)}{n}\right)^n$

Proof. : by using the inequality $\binom{n}{k} \leq \left(\frac{en}{k}\right)^k$ we have

$$\left(\!\!\binom{n}{k}\!\!\right) = \binom{n+k-1}{n-1} \leq \left(\frac{2e(n+k)}{n}\right)^n$$

□

1.2. Vocabulary based embedding

In the following theorem, we show how an upper bound on the separation rank is implied by the rank of vocabulary matrix.

Theorem 1. *Let $y_p^{i,L,d_x,H,r}$ be the scalar function computing the p^{th} entry of an output vector at position $i \in [N]$ of the H-headed depth-L width-d_x Transformer network defined in eqs. 1 and 5 of the main text, where the embedding rank r is defined by eq. 3 of the main text. Let r_e denote the rank of the positional embedding matrix and sep $\left(y_p^{i,L,d_x,H,r}\right)$ denote its separation rank w.r.t. any partition $P \mathbin{\dot\cup} Q = [N]$. Then the following holds:*

$$sep(y_p^{i,L,d_x,H,r}) \leq \left(\!\!\binom{r+r_e}{\cdot 3^L}\!\!\right) \left(\!\!\binom{4}{3^L}\!\!\right) \left(3^L+1\right)^{r+r_e} \tag{2}$$

Proof. By the embedding low-rank assumptions, there exists $M^{\text{vocab}} \in \mathbb{R}^{r \times V}, M^{\text{pos}} \in \mathbb{R}^{r_e \times N}$ and $M^{\text{low-rank}} \in \mathbb{R}^{d_x \times r}, P^{\text{low-rank}} \in \mathbb{R}^{d_x \times r_e}$ such that

$$\mathbf{y}^{0,i} = M^{\text{low-rank}} M^{\text{vocab}}\, \hat{\mathbf{w}}^i + P^{\text{low-rank}} M_i^{\text{pos}} \tag{3}$$

So we begin by substituting $\mathbf{y}^{0,i}$ in eq 1 (for convenience, we denote $j_{C(L)+1} \coloneqq i$):

$$y^{i,L,d_x,H,r}\left(w^0,...,w^N\right) = \sum_{j_1,...,j_{C(L)}=1}^{N} \sum_{h\in[H]^{[C(L)]}} \sum_{r_1,...,r_{C(L)+1}=1}^{d_a} B^{(0,h)}_{r_1,p} \left(\prod_{c=1}^{C(L)+1} \left\langle A^{(c,h)}_{r_c}, M^{\text{low-rank}} M^{\text{vocab}}\, \hat{\mathbf{w}}^{j_c} + P^{\text{low-rank}} M^{\text{pos}}_{j_c}\right\rangle\right)$$

$$\left(\prod_{c=1}^{C(L)} \left\langle B^{(c,h)}_{r_{c+1}}, M^{\text{low-rank}} M^{\text{vocab}}\, \hat{\mathbf{w}}^{j_c} + P^{\text{low-rank}} M^{\text{pos}}_{j_c}\right\rangle\right)$$

And separating between the tokens and the positional embeddings:

$$= \underbrace{\sum_{\substack{I_A\subseteq[C(L)+1]\\ I_B\subseteq[C(L)]}}}_{\text{the indices of tokens}} \sum_{j_1,...,j_{C(L)}=1}^{N} \sum_{h\in[H]^{[C(L)]}} \sum_{r_1,...,r_{C(L)+1}=1}^{d_a} \underbrace{\left(\prod_{c\in[C(L)+1]\setminus I_A} \left\langle A^{(c,h)}_{r_c}, P^{\text{low-rank}} M^{\text{pos}}_{j_c}\right\rangle\right)\left(\prod_{c\in[C(L)]\setminus I_B} \left\langle B^{(c,h)}_{r_{c+1}}, P^{\text{low-rank}} M^{\text{pos}}_{j_c}\right\rangle\right)}_{\text{The positional embeddings}}$$

$$B^{(0,h)}_{r_1,p} \underbrace{\left(\prod_{c\in I_A} \left\langle A^{(c,h)}_{r_c}, M^{\text{low-rank}} M^{\text{vocab}}\, \hat{\mathbf{w}}^{j_c}\right\rangle\right)\left(\prod_{c\in I_B} \left\langle B^{(c,h)}_{r_{c+1}}, M^{\text{low-rank}} M^{\text{vocab}}\, \hat{\mathbf{w}}^{j_c}\right\rangle\right)}_{\text{The tokens}}$$

Now we can open the inner products, explicitly writing the indices:

$$= \sum_{\substack{I_A\subseteq[C(L)+1]\\ I_B\subseteq[C(L)]}} \sum_{\substack{\alpha_1,...,\alpha_{C(L)+1}\\ \beta_1,...,\beta_{C(L)}}=1}^{r} \sum_{\substack{\sigma_1,...,\sigma_{C(L)+1}\\ \mu_1,...,\mu_{C(L)}}=1}^{r_e} \sum_{j_1,...,j_{C(L)}=1}^{N} \sum_{h\in[H]^{[C(L)]}} \sum_{r_1,...,r_{C(L)+1}=1}^{d_a} B^{(0,h)}_{r_1,p}$$

$$\left(\sum_{\substack{\gamma_1,...,\gamma_{C(L)+1}\\ \delta_1,...,\delta_{C(L)}}=1}^{d_x} \left(\prod_{c\in[C(L)+1]\setminus I_A} A^{(c,h)}_{r_c,\gamma_c} P^{\text{low-rank}}_{\gamma_c,\sigma_c} M^{\text{pos}}_{\sigma_c,j_c}\right)\left(\prod_{c\in[C(L)]\setminus I_B} B^{(c,h)}_{r_{c+1},\delta_c} P^{\text{low-rank}}_{\delta_c,\mu_c} M^{\text{pos}}_{\mu_c,j_c}\right)\right)$$

$$\left(\sum_{\substack{\gamma_1,...,\gamma_{C(L)+1}\\ \delta_1,...,\delta_{C(L)}}=1}^{d_x} \left(\prod_{c\in I_A} A^{(c,h)}_{r_c,\gamma_c} M^{\text{low-rank}}_{\gamma_c,\alpha_c} M^{\text{vocab}}_{\alpha_c,w^{j_c}}\right)\left(\prod_{c\in I_B} B^{(c,h)}_{r_{c+1},\delta_c} M^{\text{low-rank}}_{\delta_c,\beta_c} M^{\text{vocab}}_{\beta_c,w^{j_c}}\right)\right)$$

And separating between coefficients and w's:

$$= \sum_{\substack{I_A\subseteq[C(L)+1]\\ I_B\subseteq[C(L)]}} \sum_{\substack{\alpha_1,...,\alpha_{C(L)+1}\\ \beta_1,...,\beta_{C(L)}}=1}^{r} \sum_{\substack{\sigma_1,...,\sigma_{C(L)+1}\\ \mu_1,...,\mu_{C(L)}}=1}^{r_e} \sum_{j_1,...,j_{C(L)}=1}^{N} \tau_{I_A,I_B,\alpha_1,...,\mu_{C(L)}} \left(\prod_{c\in[C(L)+1]\setminus I_A} M^{\text{pos}}_{\sigma_c,j_c}\right)\left(\prod_{c\in[C(L)]\setminus I_B} M^{\text{pos}}_{\mu_c,j_c}\right)$$

$$\left(\prod_{c\in I_A} M^{\text{vocab}}_{\alpha_c,w^{j_c}}\right)\left(\prod_{c\in I_B} M^{\text{vocab}}_{\beta_c,w^{j_c}}\right)$$

Where the coefficients are equals to:

$$\tau_{I_A,I_B,\alpha_1,...,\mu_{C(L)}} \coloneqq \sum_{h\in[H]^{[C(L)]}} \sum_{r_1,...,r_{C(L)+1}=1}^{d_a} B^{(0,h)}_{r_1,p} \left[\sum_{\substack{\gamma_1,...,\gamma_{C(L)+1}\\ \delta_1,...,\delta_{C(L)}}=1}^{d_x} \left(\prod_{c\in I_A} A^{(c,h)}_{r_c,\gamma_c} M^{\text{low-rank}}_{\gamma_c,\alpha_c}\right)\left(\prod_{c\in I_B} B^{(c,h)}_{r_{c+1},\delta_c} M^{\text{low-rank}}_{\delta_c,\beta_c}\right)\right]$$

$$\left[\sum_{\substack{\gamma_1,...,\gamma_{C(L)+1}\\ \delta_1,...,\delta_{C(L)}}=1}^{d_x} \left(\prod_{c\in[C(L)+1]\setminus I_A} A^{(c,h)}_{r_c,\gamma_c} P^{\text{low-rank}}_{\gamma_c,\sigma_c}\right)\left(\prod_{c\in[C(L)]\setminus I_B} B^{(c,h)}_{r_{c+1},\delta_c} P^{\text{low-rank}}_{\delta_c,\mu_c}\right)\right]$$

Now we can group monomials by the powers $n_1, \dots, n_r, p_1, \dots, p_{r_e}$ of each coordinate:

$$= \underbrace{\sum_{N_{A \triangle B}=0}^{C(L)+1} \sum_{N_{A\cap B}=0}^{C(L)}}_{\substack{\text{How many } j_c \text{ indices} \\ \text{are token indices}}} \underbrace{\sum_{\substack{n_1+\dots+n_r=N_{A \triangle B}+2N_{A\cap B} \\ p_1+\dots+p_{r_e}=2C(L)+1-N_{A \triangle B}-2N_{A\cap B}}}}_{\text{The powers}} \underbrace{\sum_{\substack{m_1+\dots+m_N=N_{A \triangle B}+2N_{A\cap B} \\ z_1+\dots+z_N=2C(L)+1-N_{A \triangle B}-2N_{A\cap B} \\ \forall j\in[N]\; m_j+z_j\equiv\begin{cases}1 \bmod 2 & j=i \\ 0 \bmod 2 & j\neq i\end{cases}}}}_{\substack{\text{How many indices} \\ \text{are equal to each } j\in[N]}} \underbrace{\sum_{\substack{0\leq n_{1,1},\dots,n_{r,N}\leq N_{A \triangle B}+2N_{A\cap B} \\ \forall\alpha\in[r]\; \sum_{j=1}^{N} n_{\alpha,j}=n_\alpha \\ \forall j\in[N]\; \sum_{\alpha=1}^{r} n_{\alpha,j}=m_j}}}_{\text{How to distribute the token powers between } [N]}$$

$$\underbrace{\sum_{\substack{0\leq p_{1,1},\dots,p_{r_e,N}\leq 2C(L)+1-N_{A \triangle B}-2N_{A\cap B} \\ \forall\sigma\in[r_e]\; \sum_{j=1}^{N} p_{\sigma,j}=p_\sigma \\ \forall j\in[N]\; \sum_{\sigma=1}^{r_e} p_{\sigma,j}=z_j}}}_{\text{How to distribute the pos powers between } [N]} \lambda_{N_{A \triangle B}N_{A\cap B},n_1,\dots,n_r,p_1,\dots,p_{r_e}} \left(\prod_{j=1}^{N}\prod_{\sigma=1}^{r_e}\left(M^{\text{pos}}_{\sigma,j}\right)^{p_{\sigma,j}}\right)\left(\prod_{j=1}^{N}\prod_{\alpha=1}^{r}\left(M^{\text{vocab}}_{\alpha,w^j}\right)^{n_{\alpha,j}}\right)$$

Where

$$\lambda_{N_{A \triangle B}N_{A\cap B},n_1,\dots,n_r,p_1,\dots,p_{r_e}} \coloneqq \sum_{\substack{I_A\subseteq[C(L)+1] \\ I_B\subseteq[C(L)] \\ |I_A \triangle I_B|=N_{A \triangle B} \\ |I_A\cap I_B|=N_{A\cap B}}} \sum_{\substack{\substack{\alpha_1,\dots,\alpha_{C(L)+1} \\ \beta_1,\dots,\beta_{C(L)}}=1 \\ \forall\delta\in[r]\; |\{c\in I_A|\alpha_c=\delta\}|+|\{c\in I_B|\beta_c=\delta\}|=n_\delta}}^{r}$$

$$\sum_{\substack{\substack{\sigma_1,\dots,\sigma_{C(L)+1} \\ \mu_1,\dots,\mu_{C(L)}}=1 \\ \forall\delta\in[r_e]\; |\{c\in[C(L)+1]\setminus I_A|\sigma_c=\delta\}|+|\{c\in[C(L)]\setminus I_B|\mu_c=\delta\}|=p_\delta}}^{r_e} \tau_{I_A,I_B,\alpha_1,\dots,\mu_{C(L)}}$$

Now we can divide the powers between P, Q in the following way:

$$= \sum_{N_{A \triangle B}=0}^{C(L)+1} \sum_{N_{A\cap B}=0}^{C(L)} \sum_{\substack{n_1+\dots+n_r=N_{A \triangle B}+2N_{A\cap B} \\ p_1+\dots+p_{r_e}=2C(L)+1-N_{A \triangle B}-2N_{A\cap B}}} \underbrace{\sum_{\substack{m_P+m_Q=N_{A \triangle B}+2N_{A\cap B} \\ z_P+z_Q=2C(L)+1-N_{A \triangle B}-2N_{A\cap B} \\ \forall j\in\{P,Q\}\quad m_j+z_j\equiv\begin{cases}1 \bmod 2 & i\in j \\ 0 \bmod 2 & i\notin j\end{cases}}}}_{\substack{\text{How many indices} \\ \text{are in } P \text{ and in } Q}} \underbrace{\sum_{\substack{0\leq n_{1,P},\dots,n_{r,Q},p_{1,P},\dots,p_{r,Q}\leq 2C(L)+1 \\ \forall\alpha\in[r]\; n_{\alpha,P}+n_{\alpha,Q}=n_\alpha \\ \forall j\in\{P,Q\}\; \sum_{\alpha=1}^{r} n_{\alpha,j}=m_j\wedge\sum_{\sigma=1}^{r_e} p_{\sigma,j}=z_j \\ \forall\sigma\in[r_e]\; p_{\sigma,P}+p_{\sigma,Q}=p_\sigma}}}_{\text{How to distribute the powers between } P \text{ and in } Q}$$

$$\lambda_{N_{A \triangle B}N_{A\cap B},n_1,\dots,n_r,p_1,\dots,p_{r_e}} \chi_P \chi_Q$$

Where χ_P, χ_Q are functions of P, Q that defined as:

$$\chi_T \coloneqq \sum_{\substack{(m_j)_{j\in T}\in[m_T]\cup\{0\} \\ (z_j)_{j\in T}\in[z_T]\cup\{0\} \\ \sum_{j\in T} m_j=m_T\wedge\sum_{j\in T} z_j=z_T \\ \forall j\in T\; m_j+z_j\equiv\begin{cases}1 \bmod 2 & i=j \\ 0 \bmod 2 & i\neq j\end{cases}}} \sum_{\substack{(n_{\alpha,j})_{\alpha,j\in T\times[r]},(p_{\alpha,j})_{\alpha,j\in T\times[r_e]}\in[2C(L)+1]\cup\{0\} \\ \forall\alpha\in[r]\; \sum_{j\in T} n_{\alpha,j}=n_{\alpha,T} \\ \forall\alpha\in[r_e]\; \sum_{j\in T} p_{\alpha,j}=p_{\alpha,T} \\ \forall j\in T\; \sum_{\alpha=1}^{r} n_{\alpha,j}=m_{j,T}\wedge\sum_{\alpha=1}^{r_e} p_{\alpha,j}=z_{j,T}}} \prod_{j\in T}\left(\prod_{\alpha=1}^{r}\left(M^{\text{vocab}}_{\alpha,w^j}\right)^{n_{m,j}}\right)\left(\prod_{\sigma=1}^{r_e}\left(M^{\text{pos}}_{\sigma,j}\right)^{p_{\sigma,j}}\right)$$

Thus, since each summand is of separation rank 1 , the separation rank of $y_p^{i,L,d_x,H,r}$ is bounded by the number of summands:

$$\sum_{N_{A\,\triangle\,B}=0}^{C(L)+1}\sum_{N_{A\cap B}=0}^{C(L)}\sum_{\substack{n_1+\cdots+n_r=N_{A\,\triangle\,B}+2N_{A\cap B}\\ p_1+\cdots+p_{r_e}=2C(L)+1-N_{A\,\triangle\,B}-2N_{A\cap B}}}\underbrace{\sum_{\substack{m_P+m_Q=N_{A\,\triangle\,B}+2N_{A\cap B}\\ z_P+z_Q=2C(L)+1-N_{A\,\triangle\,B}-2N_{A\cap B}\\ \forall j\in\{P,Q\}\quad m_j+z_j\equiv\begin{cases}1 \bmod 2 & i\in j\\ 0 \bmod 2 & i\notin j\end{cases}}}}_{\substack{\text{How many indices}\\ \text{are in } P \text{ and in } Q}}\underbrace{\left(\prod_{\alpha=1}^{r}\left(\!\!\binom{2}{n_\alpha}\!\!\right)\right)\left(\prod_{\sigma=1}^{r_e}\left(\!\!\binom{2}{p_\sigma}\!\!\right)\right)}_{\text{How to distribute the powers between } P \text{ and in } Q}$$

$$\leq\underbrace{\left(\!\!\binom{r+r_e}{2C\left(L\right)+1}\!\!\right)}_{\substack{\text{ways to divide the powers}\\ \text{between the coordinates}}}\underbrace{\left(\!\!\binom{4}{2C\left(L\right)+1}\!\!\right)}_{\substack{\text{ways to divide the indices}\\ \text{between } P \text{ and in } Q}}\underbrace{\left(\frac{2C\left(L\right)+1}{r}+1\right)^{r}\left(\frac{2C\left(L\right)+1}{r_e}+1\right)^{r_e}}_{\text{How to distribute the powers between } P \text{ and in } Q}$$

where the inequality followed from lemma 2. □

From here, the upper bound in theorem 1 of the main text for vocabulary based embedding follows by lemma 3 with an additional assumption that $r_e = 1$. This assumption is reasonable since successful models such as T5 (Raffel et al., 2020) use rank 1 positional embeddings. Moreover, in order to verify the validly of this assumption for our setting in practice, in subsection 3.4 we show that the degradation in performance of models with a low rank positional embedding matrix is much smaller than the degradation caused by the analyzed bottleneck effects.

1.3. Convolution based embedding

In the following theorem, we show how an upper bound on the separation rank is implied by the rank of convolution based embedding. The proof uses similar techniques to the ones used in the previous subsection with some modifications due to the first convolutional layer.

Theorem 2. *Let $y_p^{i,L,d_x,H,r}$ be the scalar function computing the p^{th} entry of an output vector at position $i \in [N]$ of the H-headed depth-L width-d_x Transformer network defined in eq. 2 and 5 of the main text, where the embedding rank r is defined by eq. 4 of the main text. Let r_e denote the rank of the positional embedding matrix and sep $\left(y_p^{i,L,d_x,H,r}\right)$ denote its separation rank w.r.t. any partition $P \cupdot Q = [M]$ that does not split any patch. Then the following holds:*

$$sep(y_p^{i,L,d_x,H,r}) \leq \left(\!\!\binom{r+r_e}{\cdot 3^L}\!\!\right)\left(\!\!\binom{4}{3^L}\!\!\right)\left(3^L+1\right)^{r+r_e} \tag{4}$$

Proof. By the embedding low-rank assumptions, there exists $M^{\text{conv}} \in \mathbb{R}^{\frac{M}{N}\times r\times d_{\text{input}}}$, $M^{\text{low-rank}} \in \mathbb{R}^{d_x\times r}$, $M^{\text{pos}} \in \mathbb{R}^{N\times r_e}$ and $P^{\text{low-rank}} \in \mathbb{R}^{d_x\times r_e}$ such that:

$$\mathbf{y}^{0,i} = \sum_{k=1}^{\frac{M}{N}} M^{\text{low-rank}}\, M_k^{\text{conv}}\, \mathbf{x}^{\frac{M}{N}\cdot(i-1)+k} + P^{\text{low-rank}} M_i^{\text{pos}} \tag{5}$$

We can begin by substituting $\mathbf{y}^{0,i}$ in eq 1 (for convenience, we denote $j_{C(L)+1} \coloneqq i$):

$$\begin{aligned}y_p^{i,L,d_x,H,\Theta} = \sum_{j_1,\ldots,j_{C(L)}=1}^{N}\sum_{h\in[H]^{[C(L)]}}\sum_{r_1,\ldots,r_{C(L)+1}=1}^{d_a} B_{r_1,p}^{(0,h)} &\left(\prod_{c=1}^{C(L)+1}\left\langle A_{r_c}^{(c,h)}, \sum_{k=1}^{\frac{M}{N}} M^{\text{low-rank}}\, M_k^{\text{conv}}\, \mathbf{x}^{\frac{M}{N}\cdot(j_c-1)+k} + P^{\text{low-rank}} M_{j_c}^{\text{pos}}\right\rangle\right)\\ &\left(\prod_{c=1}^{C(L)}\left\langle B_{r_{c+1}}^{(c,h)}, \sum_{k=1}^{\frac{M}{N}} M^{\text{low-rank}}\, M_k^{\text{conv}}\, \mathbf{x}^{\frac{M}{N}\cdot(j_c-1)+k} + P^{\text{low-rank}} M_{j_c}^{\text{pos}}\right\rangle\right)\end{aligned}$$

And separating between the tokens and the positional embeddings:

$$= \underbrace{\sum_{\substack{I_A \subseteq [C(L)+1] \\ I_B \subseteq [C(L)]}}}_{\text{the indices of tokens}} \sum_{j_1,\ldots,j_{C(L)}=1}^{N} \sum_{h \in [H]^{[C(L)]}} \sum_{r_1,\ldots,r_{C(L)+1}=1}^{d_a} \underbrace{\left(\prod_{c \in [C(L)+1] \setminus I_A} \left\langle A_{r_c}^{(c,h)}, P^{\text{low-rank}} M_{j_c}^{\text{pos}} \right\rangle \right) \left(\prod_{c \in [C(L)] \setminus I_B} \left\langle B_{r_{c+1}}^{(c,h)}, P^{\text{low-rank}} M_{j_c}^{\text{pos}} \right\rangle \right)}_{\text{The positional embeddings}}$$

$$B_{r_1,p}^{(0,h)} \underbrace{\left(\prod_{c \in I_A} \left\langle A_{r_c}^{(c,h)}, \sum_{k=1}^{\frac{M}{N}} M^{\text{low-rank}} \, M_k^{\text{conv}} \, \mathbf{x}^{\frac{M}{N} \cdot (j_c - 1) + k} \right\rangle \right) \left(\prod_{c \in I_B} \left\langle B_{r_{c+1}}^{(c,h)}, \sum_{k=1}^{\frac{M}{N}} M^{\text{low-rank}} \, M_k^{\text{conv}} \, \mathbf{x}^{\frac{M}{N} \cdot (j_c - 1) + k} \right\rangle \right)}_{\text{The tokens}}$$

Now we can open the inner products, explicitly writing the indices:

$$= \sum_{\substack{I_A \subseteq [C(L)+1] \\ I_B \subseteq [C(L)]}} \sum_{\substack{\alpha_1,\ldots,\alpha_{C(L)+1} \\ \beta_1,\ldots,\beta_{C(L)}}=1}^{r} \sum_{\substack{\kappa_1,\ldots,\kappa_{C(L)+1} \\ \eta_1,\ldots,\eta_{C(L)}}=1}^{\frac{M}{N}} \sum_{\substack{\sigma_1,\ldots,\sigma_{C(L)+1} \\ \mu_1,\ldots,\mu_{C(L)}}=1}^{r_e} \sum_{j_1,\ldots,j_{C(L)}=1}^{N} \sum_{h \in [H]^{[C(L)]}} \sum_{r_1,\ldots,r_{C(L)+1}=1}^{d_a} B_{r_1,p}^{(0,h)}$$

$$\left(\sum_{\substack{\gamma_1,\ldots,\gamma_{C(L)+1} \\ \delta_1,\ldots,\delta_{C(L)}}=1}^{d_x} \left(\prod_{c \in [C(L)+1] \setminus I_A} A_{r_c,\gamma_c}^{(c,h)} P_{\gamma_c,\sigma_c}^{\text{low-rank}} M_{\sigma_c,j_c}^{\text{pos}} \right) \left(\prod_{c \in [C(L)] \setminus I_B} B_{r_{c+1},\delta_c}^{(c,h)} P_{\delta_c,\mu_c}^{\text{low-rank}} M_{\mu_c,j_c}^{\text{pos}} \right) \right)$$

$$\left(\sum_{\substack{\gamma_1,\ldots,\gamma_{C(L)+1} \\ \delta_1,\ldots,\delta_{C(L)}}=1}^{d_x} \left(\prod_{c \in I_A} A_{r_c,\gamma_c}^{(c,h)} M_{\gamma_c,\alpha_c}^{\text{low-rank}} \left(M_{\kappa_c}^{\text{conv}} \, \mathbf{x}^{\frac{M}{N} \cdot (j_c - 1) + \kappa_c} \right)_{\alpha_c} \right) \left(\prod_{c \in I_B} B_{r_{c+1},\delta_c}^{(c,h)} M_{\delta_c,\beta_c}^{\text{low-rank}} \left(M_{\eta_c}^{\text{conv}} \, \mathbf{x}^{\frac{M}{N} \cdot (j_c - 1) + \eta_c} \right)_{\beta_c} \right) \right)$$

And separating between coefficients and embeddings:

$$= \sum_{\substack{I_A \subseteq [C(L)+1] \\ I_B \subseteq [C(L)]}} \sum_{\substack{\alpha_1,\ldots,\alpha_{C(L)+1} \\ \beta_1,\ldots,\beta_{C(L)}}=1}^{r} \sum_{\substack{\kappa_1,\ldots,\kappa_{C(L)+1} \\ \eta_1,\ldots,\eta_{C(L)}}=1}^{\frac{M}{N}} \sum_{\substack{\sigma_1,\ldots,\sigma_{C(L)+1} \\ \mu_1,\ldots,\mu_{C(L)}}=1}^{r_e} \sum_{j_1,\ldots,j_{C(L)}=1}^{N} \tau_{I_A,I_B,\alpha_1,\ldots,\mu_{C(L)}}$$

$$\left(\prod_{c \in [C(L)+1] \setminus I_A} M_{\sigma_c,j_c}^{\text{pos}} \right) \left(\prod_{c \in [C(L)] \setminus I_B} M_{\mu_c,j_c}^{\text{pos}} \right) \left(\prod_{c \in I_A} \left(M_{\kappa_c}^{\text{conv}} \, \mathbf{x}^{\frac{M}{N} \cdot (j_c - 1) + \kappa_c} \right)_{\alpha_c} \right) \left(\prod_{c \in I_B} \left(M_{\eta_c}^{\text{conv}} \, \mathbf{x}^{\frac{M}{N} \cdot (j_c - 1) + \eta_c} \right)_{\beta_c} \right)$$

Where the coefficients are equal to:

$$\tau_{I_A,I_B,\alpha_1,\ldots,\mu_{C(L)}} := \sum_{h \in [H]^{[C(L)]}} \sum_{r_1,\ldots,r_{C(L)+1}=1}^{d_a} B_{r_1,p}^{(0,h)} \left[\sum_{\substack{\gamma_1,\ldots,\gamma_{C(L)+1} \\ \delta_1,\ldots,\delta_{C(L)}}=1}^{d_x} \left(\prod_{c \in I_A} A_{r_c,\gamma_c}^{(c,h)} M_{\gamma_c,\alpha_c}^{\text{low-rank}} \right) \left(\prod_{c \in I_B} B_{r_{c+1},\delta_c}^{(c,h)} M_{\delta_c,\beta_c}^{\text{low-rank}} \right) \right]$$

$$\left[\sum_{\substack{\gamma_1,\ldots,\gamma_{C(L+1)} \\ \delta_1,\ldots,\delta_{C(L)}}=1}^{d_x} \left(\prod_{c \in [C(L)+1] \setminus I_A} A_{r_c,\gamma_c}^{(c,h)} P_{\gamma_c,\sigma_c}^{\text{low-rank}} \right) \left(\prod_{c \in [C(L)] \setminus I_B} B_{r_{c+1},\delta_c}^{(c,h)} P_{\delta_c,\mu_c}^{\text{low-rank}} \right) \right]$$

Now we can group monomials by the powers $n_1, \ldots, n_r, p_1, \ldots, p_{r_e}$ of each coordinate:

$$= \underbrace{\sum_{N_{A \triangle B}=0}^{C(L)+1} \sum_{N_{A\cap B}=0}^{C(L)}}_{\substack{\text{How many } j_c \text{ indices}\\ \text{are token indices}}} \underbrace{\sum_{\substack{n_1+\cdots+n_r=N_{A \triangle B}+2N_{A\cap B}\\ p_1+\cdots+p_{r_e}=2C(L)+1-N_{A \triangle B}-2N_{A\cap B}}}}_{\text{The powers}} \underbrace{\sum_{\substack{m_{1,1}+\cdots+m_{N,\frac{M}{N}}=N_{A \triangle B}+2N_{A\cap B}\\ z_{1,1}+\cdots+z_{N,\frac{M}{N}}=2C(L)+1-N_{A \triangle B}-2N_{A\cap B}\\ \forall j\in[N]\ \sum_{k=1}^{\frac{M}{N}}(m_{j,k}+z_{j,k})\equiv\begin{cases}1 \text{ mod } 2 & i=j\\ 0 \text{ mod } 2 & i\neq j\end{cases}}}}_{\substack{\text{How many indices}\\ \text{are equal to each } (j,k)\in[N]\times\left[\frac{M}{N}\right]}}$$

$$\underbrace{\sum_{\substack{0\leq n_{1,1,1},\ldots,n_{r,N,\frac{M}{N}}\leq N_{A \triangle B}+2N_{A\cap B}\\ \forall\alpha\in[r]\ \sum_{j=1}^{N}\sum_{k=1}^{\frac{M}{N}} n_{\alpha,j,k}=n_\alpha\\ \forall(j,k)\in[N]\times\left[\frac{M}{N}\right]\ \sum_{\alpha=1}^{r} n_{\alpha,j,k}=m_{j,k}}}}_{\text{How to distribute the pixel powers between } [N]\times\left[\frac{M}{N}\right]} \underbrace{\sum_{\substack{0\leq p_{1,1,1},\ldots,p_{r_e,N,\frac{M}{N}}\leq 2C(L)+1-N_{A \triangle B}-2N_{A\cap B}\\ \forall\sigma\in[r_e]\ \sum_{j=1}^{N}\sum_{k=1}^{\frac{M}{N}} p_{\sigma,j,k}=p_\sigma\\ \forall(j,k)\in[N]\times\left[\frac{M}{N}\right]\ \sum_{\sigma=1}^{r_e} p_{\sigma,j,k}=z_{j,k}}}}_{\text{How to distribute the pos powers between } [N]\times\left[\frac{M}{N}\right]}$$

$$\Gamma_{z_{1,1},\ldots,z_{N,\frac{M}{N}},p_{1,1,1},\ldots,p_{r,N,\frac{M}{N}}} \lambda_{N_{A \triangle B}N_{A\cap B},n_1,\ldots,n_r,p_1,\ldots,p_{r_e}} \left(\prod_{j=1}^{N}\prod_{k=1}^{\frac{M}{N}}\prod_{\sigma=1}^{r_e}\left(M^{\text{pos}}_{\sigma,j}\right)^{p_{\sigma,j,k}}\right)\left(\prod_{j=1}^{N}\prod_{k=1}^{\frac{M}{N}}\prod_{\alpha=1}^{r}\left(\left(M^{\text{conv}}_k \mathbf{x}^{\frac{M}{N}\cdot(j-1)+k}\right)_\alpha\right)^{n_{\alpha,j,k}}\right)$$

where

$$\lambda_{N_{A \triangle B}N_{A\cap B},n_1,\ldots,n_r,p_1,\ldots,p_{r_e}} := \sum_{\substack{I_A\subseteq[C(L)+1]\\ I_B\subseteq[C(L)]\\ |I_A \triangle I_B|=N_{A \triangle B}\\ |I_A\cap I_B|=N_{A\cap B}}} \sum_{\substack{\alpha_1,\ldots,\alpha_{C(L)+1}\\ \beta_1,\ldots,\beta_{C(L)}\\ \forall\delta\in[r]\ |\{c\in I_A|\alpha_c=\delta\}|+|\{c\in I_B|\beta_c=\delta\}|=n_\delta}=1}^{r} \sum_{\substack{\kappa_1,\ldots,\kappa_{C(L)+1}\\ \eta_1,\ldots,\eta_{C(L)}\\ \forall\delta\in\left[\frac{M}{N}\right]\ |\{c\in I_A|\kappa_c=\delta\}|+|\{c\in I_B|\eta_c=\delta\}|=n_\delta}=1}^{\frac{M}{N}}$$

$$\sum_{\substack{\sigma_1,\ldots,\sigma_{C(L)+1}\\ \mu_1,\ldots,\mu_{C(L)}\\ \forall\delta\in[r_e]\ |\{c\in[C(L)+1]\setminus I_A|\sigma_c=\delta\}|+|\{c\in[C(L)]\setminus I_B|\mu_c=\delta\}|=p_\delta}=1}^{r_e} \tau_{I_A,I_B,\alpha_1,\ldots,\mu_{C(L)}}$$

and

$$\Gamma_{z_{1,1},\ldots,z_{N,\frac{M}{N}},p_{1,1,1},\ldots,p_{r,N,\frac{M}{N}}} := \left(\prod_{j=1}^{N}\left[\left(\!\!\left(\genfrac{}{}{0pt}{}{\frac{M}{N}}{z_{j,1}+\ldots+z_{j,\frac{M}{N}}}\right)\!\!\right)\cdot\prod_{\sigma=1}^{r_e}\left(\!\!\left(\genfrac{}{}{0pt}{}{\frac{M}{N}}{p_{\sigma,j,1}+\ldots+p_{\sigma,j,\frac{M}{N}}}\right)\!\!\right)\right]\right)^{-1}$$

Note that the positional powers are actually independent of the indices in $\left[\frac{M}{N}\right]$, so $\Gamma_{z_{1,1},\ldots,z_{N,\frac{M}{N}},p_{1,1,1},\ldots,p_{r,N,\frac{M}{N}}}$ is a multiplicative factor that is used in order to cancel out double counting.

For convenience, we will treat (P,Q) as a partition of the Cartesian product $[N]\times\left[\frac{M}{N}\right]$ (as there is a one-to-one correspondence between $[N]\times\left[\frac{M}{N}\right]$ and $[M]$). Now we can divide the powers between P, Q in the following way:

$$= \sum_{N_{A\triangle B}=0}^{C(L)+1}\sum_{N_{A\cap B}=0}^{C(L)} \sum_{\substack{n_1+\cdots+n_r=N_{A\triangle B}+2N_{A\cap B}\\ p_1+\cdots+p_{r_e}=2C(L)+1-N_{A\triangle B}-2N_{A\cap B}}} \underbrace{\sum_{\substack{m_P+m_Q=N_{A\triangle B}+2N_{A\cap B}\\ z_P+z_Q=2C(L)+1-N_{A\triangle B}-2N_{A\cap B}\\ \forall j\in\{P,Q\}\; m_j+z_j=\begin{cases}1 \bmod 2 & \exists k\in\left[\frac{M}{n}\right]\ (i,k)\in j\\ 0 \bmod 2 & \text{else}\end{cases}}}}_{\substack{\text{How many indices}\\ \text{are in } P \text{ and in } Q}}$$

$$\underbrace{\sum_{\substack{0\leq n_{1,P},\dots,n_{r,Q},p_{1,P},\dots,p_{r,Q}\leq 2C(L)+1\\ \forall\alpha\in[r]\; n_{\alpha,P}+n_{\alpha,Q}=n_\alpha\\ \forall T\in\{P,Q\}\; \sum_{\alpha=1}^{r} n_{\alpha,T}=m_T\wedge\sum_{\sigma=1}^{r_e} p_{\sigma,T}=z_T\\ \forall\sigma\in[r_e]\; p_{\sigma,P}+p_{\sigma,Q}=p_\sigma}}}_{\text{How to distribute the powers between } P \text{ and in } Q} \Gamma_{z_{1,1},\dots,z_{N,\frac{M}{N}},p_{1,1,1},\dots,p_{r,N,\frac{M}{N}}}\lambda_{N_{A\triangle B}N_{A\cap B},n_1,\dots,n_r,p_1,\dots,p_{r_e}}\chi_P\chi_Q$$

Where χ_P, χ_Q are functions of P, Q that defined as:

$$\chi_T := \sum_{\substack{(m_{j,k})_{(j,k)\in T}\in[m_T]\cup\{0\}\\ (z_{j,k})_{(j,k)\in T}\in[z_T]\cup\{0\}\\ \sum_{(j,k)\in T} m_{j,k}=m_T\wedge\sum_{(j,k)\in T} z_{j,k}=z_T\\ \forall j\in\left\{j\in[N]:\exists k\in\left[\frac{M}{n}\right]\,(j,k)\in T\right\}\; \sum_{k=1}^{\frac{M}{N}}(m_{j,k}+z_{j,k})\equiv\begin{cases}1 \bmod 2 & i=j\\ 0 \bmod 2 & i\neq j\end{cases}}} \sum_{\substack{(n_{\alpha,j,k})_{(j,k),\alpha\in T\times[r]},(p_{\alpha,j,k})_{(j,k),\alpha\in T\times[r_e]}\in[2C(L)+1]\cup\{0\}\\ \forall\alpha\in[r]\;\sum_{(j,k)\in T} n_{\alpha,j,k}=n_{\alpha,T}\\ \forall\alpha\in[r_e]\;\sum_{(j,k)\in T} p_{\alpha,j,k}=p_{\alpha,T}\\ \forall(j,k)\in T\;\sum_{\alpha=1}^{r} n_{\alpha,j,k}=m_{j,k}\wedge\sum_{\alpha=1}^{r_e} p_{\alpha,j,k}=z_{j,k}}}$$

$$\prod_{(j,k)\in T}\left(\prod_{\alpha=1}^{r}\left(\left(M_k^{\text{conv}}\,\mathbf{x}^{\frac{M}{N}\cdot(j-1)+k}\right)_\alpha\right)^{n_{\alpha,j,k}}\right)\left(\prod_{\sigma=1}^{r_e}\left(M_{\sigma,j}^{\text{pos}}\right)^{p_{\sigma,j,k}}\right)$$

Thus, since each summand is of separation rank 1 , the separation rank of $y_p^{i,L,d_x,H,r}$ is bounded by the number of summands:

$$\sum_{N_{A\triangle B}=0}^{C(L)+1}\sum_{N_{A\cap B}=0}^{C(L)} \sum_{\substack{n_1+\cdots+n_r=N_{A\triangle B}+2N_{A\cap B}\\ p_1+\cdots+p_{r_e}=2C(L)+1-N_{A\triangle B}-2N_{A\cap B}}} \underbrace{\sum_{\substack{m_P+m_Q=N_{A\triangle B}+2N_{A\cap B}\\ z_P+z_Q=2C(L)+1-N_{A\triangle B}-2N_{A\cap B}\\ \forall j\in\{P,Q\}\; m_j+z_j=\begin{cases}1 \bmod 2 & \exists k\in\left[\frac{M}{n}\right]\ (i,k)\in j\\ 0 \bmod 2 & \text{else}\end{cases}}}}_{\substack{\text{How many indices}\\ \text{are in } P \text{ and in } Q}} \underbrace{\left(\prod_{\alpha=1}^{r}\left(\!\!\binom{2}{n_\alpha}\!\!\right)\right)\left(\prod_{\sigma=1}^{r_e}\left(\!\!\binom{2}{p_\sigma}\!\!\right)\right)}_{\text{How to distribute the powers between } P \text{ and in } Q}$$

$$\leq \underbrace{\left(\!\!\binom{r+r_e}{2C\left(L\right)+1}\!\!\right)}_{\substack{\text{ways to divide the powers}\\ \text{between the coordinates}}} \underbrace{\left(\!\!\binom{4}{2C\left(L\right)+1}\!\!\right)}_{\substack{\text{ways to divide the indices}\\ \text{between } P \text{ and in } Q}} \underbrace{\left(\frac{2C\left(L\right)+1}{r}+1\right)^{r}\left(\frac{2C\left(L\right)+1}{r_e}+1\right)^{r_e}}_{\text{How to distribute the powers between } P \text{ and in } Q}$$

□

Similarly to the vocabulary embedding case from here, the upper bound in theorem 1 of the main text for convolution based embedding follows by lemma 3 with an additional assumption that $r_e = 1$.

2. Lower bounds on the separation rank

2.1. Preliminaries

2.1.1. Tensors and their matricization

We begin by laying out basic concepts in tensor theory required for the upcoming analysis. The core concept of a *tensor* may be thought of as a multi-dimensional array. The *order* of a tensor is defined to be the number of indexing entries in the array, referred to as *modes*. The *dimension* of a tensor in a particular mode is defined as the number of values taken by the index in that mode. If $\mathcal{A}$ is a tensor of order N and dimension M_i in each mode $i \in [N]$, its entries are denoted $\mathcal{A}_{d_1 \ldots d_N}$, where the index in each mode takes values $d_i \in [M_i]$.

We will make use of the concept of the *matricization of $\mathcal{A}$ w.r.t. the balanced partition* (P, Q), denoted $[\![\mathcal{A}]\!]_{P,Q} \in \mathbb{R}^{M^{N/2} \times M^{N/2}}$, which is essentially the arrangement of the tensor elements as a matrix whose rows correspond to P and columns to Q. Suppose $\mathcal{A} \in \mathbb{R}^{M \times \cdots \times M}$ is a tensor of order N, and let (P, Q) be a balanced partition of $[N]$, *i.e.* P and Q are disjoint size ${}^{N}\!/_{2}$ subsets of $[N]$ whose union gives $[N]$. The *matricization of $\mathcal{A}$ w.r.t. the partition* (P, Q), denoted $[\![\mathcal{A}]\!]_{P,Q}$, is the $M^{N/2}$-by-$M^{N/2}$ matrix holding the entries of $\mathcal{A}$ such that $\mathcal{A}_{d_1 \ldots d_N}$ is placed in row index $1 + \sum_{t=1}^{N/2} (d_{p_t} - 1) M^{N/2 - t}$ and column index $1 + \sum_{t=1}^{N/2} (d_{q_t} - 1) M^{N/2 - t}$.

2.1.2. Grid tensors provide lower bounds for the separation rank

We now present the concept of grid tensors, which are a form of function discretization (Hackbusch, 2012). Essentially, the function is evaluated for a set of points on an exponentially large grid in the input space and the outcomes are stored in a tensor. Formally, fixing a set of *template* vectors $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(Z)}$, either in $\mathbb{R}^{d_{\text{input}}}$ for the convolutional embedding method or in $[V]$ for the vocabulary embedding method, the points on the grid are the set $\{(\mathbf{x}^{(d_1)}, \ldots, \mathbf{x}^{(d_N)})\}_{d_1, \ldots, d_N = 1}^{Z}$. Given a function $y(\mathbf{x}^1, \ldots, \mathbf{x}^N)$, the set of its values on the grid arranged in the form of a tensor are called the grid tensor induced by y, denoted $\mathcal{A}(y)_{d_1, \ldots, d_N} \equiv y(\mathbf{x}^1 = \mathbf{x}^{(d_1)}, \ldots, \mathbf{x}^N = \mathbf{x}^{(d_N)})$.

Let T denote the number of raw inputs to the network. In the notation of section 2.1 of the main text, T is either equal to N in the case of the vocabulary input embedding or M in the case of the convolution input embedding. The following claim from (Levine et al., 2020) establishes a fundamental relation between a function's separation rank (see section 3 of the main text) and the rank of the matrix obtained by the corresponding grid tensor matricization. This relation, which holds for all functions, is formulated below for functions realized by the analyzed Transformer network:

Claim 1. *Let $y_p^{i,L,d_x,H,r}$ be the scalar function computing the p^{th} entry of an output vector at position $i \in [N]$ of the H-headed depth-L width-d_x Transformer network defined in eq. 5 and either eq 1 or eq 2 of the main text. Let sep $\left(y_p^{i,L,d_x,H,r}\right)$ denote its separation rank w.r.t. any partition $P \cupdot Q = [T]$. Then, for any integer Z and any set of template vectors $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(Z)} \in \mathbb{R}^{d_x}$ it holds that:*

$$\mathrm{sep}_{(P,Q)}\left(y_p^{i,L,d_x,H,r}\right) \geq \mathrm{rank}\left([\![\mathcal{A}(y_p^{i,L,d_x,H,r})]\!]_{P,Q}\right), \tag{6}$$

where $\mathcal{A}(y_p^{i,L,d_x,H,r})$ is the grid tensor of $y_p^{i,L,d_x,H,r}$ with respect to the above template vectors.

In the next subsection we will show a corollary from (Levine et al., 2020) that uses this claim to prove the lower bound in theorem 2 of the main text.

2.2. Proof of the lower bounds

In this subsection we prove the lower bound in theorem 2 of the main text. We will use a direct corollary of the proof in Levine et al. (2020) regarding composition of the self-attention separation rank. Essentially, though the required form of $\mathbf{y}^{0,j}$ in corollary below looks complex, Levine et al. (2020) prove that for this form of inputs to the self-attention block, the rank of the grid tensor is with probability 1 lower bounded by the multiset term in eq 7 below. The corollary below simply states that if the input embedding is able to produce vectors that do not change the analysis in (Levine et al., 2020), their bound on the grid tensor rank can be used, and together with claim 1 this implies a lower bound on the separation rank.

Denote by $y_p^{i,L,d_x,H,r}$ the scalar function computing the p^{th} entry of an output vector at position $i \in [N]$ of the H-headed depth-L width-d_x Transformer network defined in eq. 5 and either eq 1 or eq 2 of the main text, then:

Corollary 1. *Assume that for any matrix $A \in \mathbb{R}^{\left(\!\!\binom{(r-H)/2}{3^{L-2}}\!\!\right) \times (r-H)/2}$ with rows that are l^2 normalized, there exists a choice of template vectors $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(Z)}$, as well as an assignment to the embedding layer weights, such that for any sequence*

$(i_j)_{j=1}^N \in \left[2\cdot\left(\!\!\binom{(r-H)/2}{3^{L-2}}\!\!\right)+1\right]$ *there exists a sequence of T $\mathbf{x}$'s for which the output of the embedding layer is:*

$$\forall j\in[N]\quad \mathbf{y}_\alpha^{(0,j)}=\begin{cases}A_{i_j,\phi(\alpha)} & i_j\leq {}^{V}\!/2\wedge(\alpha-1)\bmod d_a<\frac{d_a-1}{2}\wedge\phi(\alpha)\leq{}^{(r-H)}\!/2\\ A_{i_j-V/2,\phi\left(\alpha-\frac{d_a-1}{2}\right)} & {}^{V}\!/2<i_j\leq V\wedge\frac{d_a-1}{2}\leq(\alpha-1)\bmod d_a<d_a-1\wedge\phi(\alpha-\frac{d_a-1}{2})\leq{}^{(r-H)}\!/2\\ 1 & (\alpha-1)\bmod d_a=d_a-1\\ 0 & \textit{Otherwise}\end{cases}$$

where $\phi(j)\equiv\lfloor{}^{j-1}\!/_{d_a}\rfloor\cdot(d_a-1)+(j-1\bmod d_a)+1$ *and* $V:=2\left(\!\binom{(r-H)/2}{3^{L-2}}\!\right)$.

Further in the convolutional embedding case assume the partition $P\,\dot\cup\,Q=[T]$ does not split any patch. Then for all values of the network weights but a set of Lebesgue measure zero, the following holds:

$$sep(y_p^{i,L,d_x,H,r})\geq\left(\!\!\binom{(r-H)/2}{3^{L-2}}\!\!\right)\tag{7}$$

Now we will prove that both for the convolutional embedding method and the vocabulary embedding method the assumption of corollary 1 holds. We will thus prove the lower bound in theorem 2, since the following lemma 4 shows that $\log\left(\!\!\binom{(r-H)/2}{3^{L-2}}\!\!\right)=\tilde{\Omega}\left(L\cdot(\min\{r,d_x\}-H)\right)$.

Lemma 4. $\left(\!\binom{n}{k}\!\right)\geq\left(\frac{2e(n+k)}{n}\right)^n$

Proof. : by using the inequality $\binom{n}{k}\geq\left(\frac{n}{k}\right)^k$ we have $\left(\!\binom{n}{k}\!\right)=\binom{n+k-1}{n-1}\geq\left(\frac{(n+k-1)}{n-1}\right)^{n-1}$ □

2.2.1. Convolution based embedding

We start with the convolutional embedding method. The lemma below shows that the assumption of corollary 1 holds, by dividing the desired vector coordinates into chunks of size d_{input}, and using a convolutional kernel to unify these chunks.

Lemma 5. *Let* $A\in\mathbb{R}^{\left(\!\binom{(r-H)/2}{3^{L-2}}\!\right)\times(r-H)/2}$ *be a matrix with rows that are l^2 normalized, then there exists a choice of template vectors $\mathbf{x}^{(1)},\dots,\mathbf{x}^{(Z)}$, as well as an assignment to the convolutional embedding layer weights, such that for any sequence* $(i_j)_{j=1}^N\in\left[2\cdot\left(\!\!\binom{(r-H)/2}{3^{L-2}}\!\!\right)+1\right]$ *there exists a sequence of M $\mathbf{x}$'s for which the output of the embedding layer is:*

$$\forall j\in[N]\quad \mathbf{y}_\alpha^{(0,j)}=\begin{cases}A_{i_j,\phi(\alpha)} & i_j\leq {}^{V}\!/2\wedge(\alpha-1)\bmod d_a<\frac{d_a-1}{2}\wedge\phi(\alpha)\leq{}^{(r-H)}\!/2\\ A_{i_j-V/2,\phi\left(\alpha-\frac{d_a-1}{2}\right)} & {}^{V}\!/2<i_j\leq V\wedge\frac{d_a-1}{2}\leq(\alpha-1)\bmod d_a<d_a-1\wedge\phi(\alpha-\frac{d_a-1}{2})\leq{}^{(r-H)}\!/2\\ 1 & (\alpha-1)\bmod d_a=d_a-1\\ 0 & \textit{Otherwise}\end{cases}\tag{8}$$

where $\phi(j)\equiv\lfloor{}^{j-1}\!/_{d_a}\rfloor\cdot(d_a-1)+(j-1\bmod d_a)+1$ *and* $V:=2\left(\!\binom{(r-H)/2}{3^{L-2}}\!\right)$.

Proof. Denote the convolutional kernel's width by $k:=\frac{M}{N}$. We will define a convolutional kernel, $W^{\text{conv}}\in\mathbb{R}^{k\times d_x\times d_{\text{input}}}$, a positional embedding matrix $P\in\mathbb{R}^{N\times d_x}$, and a set k template vectors, $\mathbf{x}^{(j,1)},\dots,\mathbf{x}^{(j,k)}$ for each $j\in[N]$, such that:

$$\mathbf{y}^{(0,j)}=\left(\sum_{l=1}^{k}W_l^{\text{conv}}\mathbf{x}^{(j,l)}+P_i\right)_\alpha$$

We will assign weights for the convolutional kernel that will "read" only the non zeros coordinates of eq 8:

$$W^{\text{conv}}_{l,\alpha,\lambda}=\begin{cases}1 & k\cdot(\lambda-1)+l=\alpha\wedge(\alpha-1)\bmod d_a<\frac{d_a-1}{2}\wedge\phi\left(\alpha\right)\leq{}^{(r-H)}/_2\\ 1 & k\cdot(\lambda-1)+l=\alpha\wedge\frac{d_a-1}{2}\leq(\alpha-1)\bmod d_a<d_a-1\wedge\phi\left(\alpha-\frac{d_a-1}{2}\right)\leq{}^{(r-H)}/_2\\ 1 & k\cdot(\lambda-1)+l=\alpha\wedge(\alpha-1)\bmod d_a=d_a-1\\ 0 & \text{Otherwise}\end{cases}$$

where $\psi\left(l,\lambda\right)\equiv\phi\left(k\cdot(\lambda-1)+l\right)$. Clearly, the rank of the chosen convolutional kernel is (at most) r and thus satisfy the assumption regarding the embedding rank in theorem 2 of the main text. We will set the positional embedding matrix to be: $P\equiv 0$, and the template vectors will be defined as follows:

$$\forall j\in\left[N\right],l\in\left[k\right]\quad\mathbf{x}^{(j,l)}_{\lambda}=\begin{cases}A_{i_j,\psi(l,\lambda)} & i_j\leq{}^{V}/_2\wedge\left(k\cdot(\lambda-1)+l-1\right)\bmod d_a<\frac{d_a-1}{2}\wedge\psi\left(l,\lambda\right)\leq{}^{(r-H)}/_2\\ A_{i_j-{}^{V}/_2,\psi\left(l-\left\lfloor\frac{d_a-1}{2}\right\rfloor,\lambda\right)} & {}^{V}/_2<i_j\leq V\wedge\frac{d_a-1}{2}\leq(k\cdot(\lambda-1)+l-1)\bmod d_a<d_a-1\wedge\psi\left(l-\left\lfloor\frac{d_a-1}{2}\right\rfloor,\lambda\right)\leq{}^{(r-H)}/_2\\ 1 & \left(k\cdot(\lambda-1)+l-1\right)\bmod d_a=d_a-1\\ 0 & \text{Otherwise}\end{cases}$$

Now, for each $j\in\left[N\right],\alpha\in\left[d_x\right]$ we have:

$$\begin{aligned}\mathbf{y}^{(0,j)}_{\alpha}&=\left(\sum_{l=1}^{k}W^{\text{conv}}_{l}\mathbf{x}^{(j,l)}\right)_{\alpha}=\sum_{l=1}^{k}\sum_{\lambda=1}^{d_{\text{input}}}W^{\text{conv}}_{l,\alpha,\lambda}\mathbf{x}^{(j,l)}_{\lambda}\\ &\overset{(1)}{=}\begin{cases}\mathbf{x}^{(j,(\alpha-1\mod k)+1)}_{\left\lfloor\frac{\alpha-1}{k}\right\rfloor+1} & \begin{matrix}\left((\alpha-1)\bmod d_a<\frac{d_a-1}{2}\wedge\phi(\alpha)\leq{}^{(r-H)}/_2\right)\vee\\ \left(\frac{d_a-1}{2}\leq(\alpha-1)\bmod d_a<d_a-1\wedge\phi\left(\alpha-\frac{d_a-1}{2}\right)\leq{}^{(r-H)}/_2\right)\vee\\ \left((\alpha-1)\bmod d_a=d_a-1\right)\end{matrix}\\ 0 & \text{Otherwise}\end{cases}\\ &\overset{(2)}{=}\begin{cases}A_{i_j,\phi(\alpha)} & \alpha\leq r\wedge i_j\leq{}^{V}/_2\wedge(\alpha-1)\bmod d_a<\frac{d_a-1}{2}\wedge\phi\left(\alpha\right)\leq{}^{(r-H)}/_2\\ A_{i_j-{}^{M}/_2,\phi(\alpha)} & \alpha\leq r\wedge{}^{V}/_2<i_j\leq V\wedge\frac{d_a-1}{2}\leq(\alpha-1)\bmod d_a<d_a-1\wedge\phi\left(\alpha-\frac{d_a-1}{2}\right)\leq{}^{(r-H)}/_2\\ 1 & \alpha\leq r\wedge(\alpha-1)\bmod d_a=d_a-1\\ 0 & \text{Otherwise}\end{cases}\end{aligned}$$

Where $^{(1)}$ due to the fact that there there's a single combination of $l\in\left[k\right],\lambda\in\left[d_{\text{input}}\right]$ that satisfies $\alpha=k\cdot(\lambda-1)+l$ (for all other value of l and λ, $W^{\text{conv}}_{l,\alpha,\lambda}=0$), and $^{(2)}$ is since:

$$\alpha=k\cdot\left\lfloor\frac{\alpha-1}{k}\right\rfloor+(\alpha-1\mod k)+1$$

□

2.2.2. Vocabulary based embedding

Now we move to the vocabulary embedding method, in this case we will use A to create an assignment for $M_{\text{V}}\in\mathbb{R}^{d_x\times V}$, since the input is no longer continuous the number of unique inputs is limited to only V^N . To overcome this issue we will add an additional assumption that either $V\geq 2\cdot\left(\!\!\binom{{}^{(r-H)}/_2}{3^{L-2}}\!\!\right)+1$ or N is very large. Importantly, the upper bound for the small vocabulary size holds with small V and N, so the bottleneck phenomenon is theoretically established for the vocabulary embedding method also in cases that neither of the additional assumptions hold.

We start with the $V\geq 2\cdot\left(\!\!\binom{{}^{(r-H)}/_2}{3^{L-2}}\!\!\right)+1$ assumption that overcomes the unique inputs issue by enlarging the number of unique tokens (while keeping the rank r constraint of M_{V}).

Lemma 6. *Assume $V\geq 2\cdot\left(\!\!\binom{{}^{(r-H)}/_2}{3^{L-2}}\!\!\right)+1$ and let $A\in\mathbb{R}^{\left(\!\binom{{}^{(r-H)}/_2}{3^{L-2}}\!\right)\times{}^{(r-H)}/_2}$ be a matrix with rows that are l^2 normalized, then there exists a choice of template vectors $\hat{\mathbf{w}}^{(1)},\ldots,\hat{\mathbf{w}}^{(Z)}$, as well as an assignment to the vocabulary embedding layer weights, such that for any sequence $\left(i_j\right)_{j=1}^{N}\in\left[2\cdot\left(\!\!\binom{{}^{(r-H)}/_2}{3^{L-2}}\!\!\right)+1\right]$ there exists a sequence of T $\hat{\mathbf{w}}$'s for which the output* *of the embedding layer is:*

$$\forall j\in[N]\quad \mathbf{y}^{(0,j)}_{\alpha}=\begin{cases}A_{i_j,\phi(\alpha)} & i_j\le {}^{E}\!/2\wedge(\alpha-1)\bmod d_a<\frac{d_a-1}{2}\wedge\phi(\alpha)\le {}^{(r-H)}\!/2\\ A_{i_j-E/2,\phi\left(\alpha-\frac{d_a-1}{2}\right)} & {}^{E}\!/2<i_j\le E\wedge\frac{d_a-1}{2}\le(\alpha-1)\bmod d_a<d_a-1\wedge\phi(\alpha-\frac{d_a-1}{2})\le {}^{(r-H)}\!/2\\ 1 & (\alpha-1)\bmod d_a=d_a-1\\ 0 & \textit{Otherwise}\end{cases}$$

where $\phi(j)\equiv\lfloor {}^{j-1}\!/d_a\rfloor\cdot(d_a-1)+(j-1\bmod d_a)+1$ *and* $E:=2\left(\!\binom{(r-H)/2}{3^{L-2}}\!\right)$.

Proof. Our templates vectors will be: $\forall i\in[E+1]\quad w^i:=i$. We will ignore the positional embedding by choosing $\mathbf{p}^i:=0$ (by the terms of corollary 1 it suffices to find any assignment of the learned weights). Now we can use A to create an assignment for $M_{\mathrm{V}}\in\mathbb{R}^{d_x\times V}$:

$$(M_{\mathrm{V}})_{\alpha,i}:=\begin{cases}A_{i_j,\phi(\alpha)} & i_j\le\left(\!\binom{(r-H)/2}{3^{L-2}}\!\right)\wedge(\alpha-1)\bmod d_a<\frac{d_a-1}{2}\\ A_{i_j-\left(\!\binom{(r-H)/2}{3^{L-2}}\!\right),\phi\left(\alpha-\frac{d_a-1}{2}\right)} & \left(\!\binom{(r-H)/2}{3^{L-2}}\!\right)<i_j\le 2\left(\!\binom{(r-H)/2}{3^{L-2}}\!\right)\wedge\frac{d_a-1}{2}\le(\alpha-1)\bmod d_a<d_a-1\\ 1 & (\alpha-1)\bmod d_a=d_a-1\\ 0 & \text{Otherwise}\end{cases}$$

Clearly, the rank of $M_{\mathrm{V}}\in\mathbb{R}^{d_x\times V}$ is (at most) r, since it has at most r non zero rows, and thus satisfy the assumption regarding the embedding rank in theorem 2 of the main text. Now by eq 1 of the main text, for any given sequence $(i_j)_{j=1}^N\in\left[2\cdot\left(\!\binom{(r-H)/2}{3^{L-2}}\!\right)+1\right]$ we get:

$$\mathbf{y}^{(0,j)}_{\alpha}=\left(M_{\mathrm{V}}\,\hat{\mathbf{w}}^{i_j}\right)_{\alpha}+\mathbf{p}^{i_j}_{\alpha}=(M_{\mathrm{V}})_{\alpha,i_j}$$

□

Now, we prove a lower bound with $V=r$ for the infinite N limit. Note that while our proof technique requires unpractical N values, its usage of N is clearly wasteful, and we conjecture (and empirically demonstrate in section 5) that the upper bound in theorem 1 of the main text is tight for $N=\Omega\left({}^{r\cdot L}\!/\log_3 r\right)$[1].

In this case, the input embedding is unable to produce vectors that do not change the analysis in (Levine et al., 2020), and therefore the assumption of corollary 1 does not holds. Instead we will use the first self-attention layer of the network to take advantage of the larger N, and apply corollary 2 below to prove a lower bound on the separation rank. This corollary which is direct results of the proof in (Levine et al., 2020) and lemma 8, simply states that if the output of the first self-attention layer is able to produce vectors that do not change the analysis in (Levine et al., 2020), their bound on the grid tensor rank can be used, and together with claim 1 this implies a lower bound on the separation rank.

Corollary 2. *Let* $d>0$*, assume that for any balanced partition of* $[T]$*, denoted* (P,Q)*, for any matrix* $A\in\mathbb{N}^{\left(\!\binom{d}{3^{L-2}}\!\right)\times d}$ *with rows that have equal* l^2 *norm, there exists a choice of template vectors* $\mathbf{x}^{(1)},\ldots,\mathbf{x}^{(Z)}$*, an assignment to the embedding layer and the first self-attention layer key and query weights, as well as a mapping* $\pi_J:\left[\left(\!\binom{d}{3^{L-2}}\!\right)\right]\to(i_j)_{j\in J}$*, such that for any* $j_1,j_2\in\left[\left(\!\binom{d}{3^{L-2}}\!\right)\right]$ *the output of the first self-attention layer on the sequence defined by* $\pi_P(j_1),\pi_Q(j_2)$ *is:*

$$\mathbf{y}^{(1,j)}=\left(\sum_{h=1}^{H}W^{O,1,h}W^{V,1,h}\right)\mathbf{u}$$

for

$$\forall\alpha\in[d_x]\quad\mathbf{u}_{\alpha}=\begin{cases}A_{j_1,\phi(\alpha)} & (\alpha-1)\bmod d_a<\frac{d_a-1}{2}\wedge\phi(\alpha)\le d\\ A_{j_2,\phi\left(\alpha-\frac{d_a-1}{2}\right)} & \frac{d_a-1}{2}\le(\alpha-1)\bmod d_a<d_a-1\wedge\phi(\alpha-\frac{d_a-1}{2})\le d\\ N & (\alpha-1)\bmod d_a=d_a-1\\ 0 & \textit{Otherwise}\end{cases}$$

[1] Since for N that is larger than this bound, the limitation of V^N unique vectors does not constitutes a bottleneck anymore.

where $\phi(j) \equiv \lfloor {}^{j-1}/_{d_a} \rfloor \cdot (d_a - 1) + (j - 1 \bmod d_a) + 1$.

Then for all values of the network weights but a set of Lebesgue measure zero, the following holds:

$$sep(y_p^{i,L,d_x,H,r}) \geq \left(\!\!\binom{d}{3^{L-2}}\!\!\right) \tag{9}$$

The lemma below shows that the assumption of corollary 2 holds for $d \coloneqq {}^{(r-1-H)}/_{2}$[2], by choosing an assignment to the first self-attention layer that utilize the large N for summing it's inputs embedding, and use π to construct sequences that repeat the one-hot embedding vectors amount of times that depends on A.

Lemma 7. *Assume $V \geq r$ and let $A \in \mathbb{N}^{\left(\!\binom{(r-1-H)/2}{3^{L-2}}\!\right) \times (r-1-H)/2}$ be a matrix with rows that have equal l^2 norm, then there exists a choice of template vectors $\hat{\mathbf{w}}^{(1)}, \ldots, \hat{\mathbf{w}}^{(Z)}$, large enough N, an assignment to the vocabulary embedding layer and the first self-attention layer key and query weights, as well as a mapping $\pi_J : \left[\left(\!\!\binom{(r-1-H)/2}{3^{L-2}}\!\!\right)\right] \to (i_j)_{j \in J}$, such that for any $j_1, j_2 \in \left[\left(\!\!\binom{(r-1-H)/2}{3^{L-2}}\!\!\right)\right]$ the output of the first self-attention layer on the sequence defined by $\pi_P(j_1), \pi_Q(j_2)$ is:*

$$\mathbf{y}^{(1,j)} = \left(\sum_{h=1}^{H} W^{O,1,h} W^{V,1,h}\right) \mathbf{u}$$

for

$$\forall \alpha \in [d_x] \quad \mathbf{u}_\alpha = \begin{cases} A_{j_1,\phi(\alpha)} & (\alpha - 1) \bmod d_a < \frac{d_a-1}{2} \wedge \phi(\alpha) \leq {}^{(r-1-H)}/_{2} \\ A_{j_2,\phi\left(\alpha - \frac{d_a-1}{2}\right)} & \frac{d_a-1}{2} \leq (\alpha - 1) \bmod d_a < d_a - 1 \wedge \phi(\alpha - \frac{d_a-1}{2}) \leq {}^{(r-1-H)}/_{2} \\ N & (\alpha - 1) \bmod d_a = d_a - 1 \\ 0 & \textit{Otherwise} \end{cases}$$

where $\phi(j) \equiv \lfloor {}^{j-1}/_{d_a} \rfloor \cdot (d_a - 1) + (j - 1 \bmod d_a) + 1$.

Proof. Our templates vectors will be: $\forall i \in [r] \quad w^i \coloneqq i$. We will ignore the positional embedding by choosing $\mathbf{p}^i \coloneqq 0$ (by the terms of corollary 2 it suffices to find any assignment of the learned weights).

To implement summation of the inputs embedding in the first self-attention layer we will follow (Levine et al., 2020) and set the inputs embedding matrix and the first layer self-attention key and query weights to:

$$(M_{\mathrm{V}})_{\alpha,i} = \begin{cases} 1 & (\alpha - 1) \bmod d_a = d_a - 1 \\ 1 & 1 < i \leq r - H \wedge \phi(\alpha) = i - 1 \\ 0 & \text{Otherwise} \end{cases}$$

$$W^{K,1,h}_{i,j} = W^{Q,1,h}_{i,j} = 1_{i=1 \wedge j=d_a}$$

Clearly, the rank of $M_{\mathrm{V}} \in \mathbb{R}^{d_x \times V}$ is (at most) r, since it has less than r non zero rows, and thus satisfy the assumption regarding the embedding rank in theorem 2 of the main text. This assignment implements summation of the inputs embedding in the first self-attention layer since:

$$\mathbf{y}^{(1,i)}(\hat{\mathbf{w}}^{(d_1)}, \ldots, \hat{\mathbf{w}}^{(d_N)})_\alpha = \sum_{j=1}^{N} \sum_{h=1}^{H} \left\langle W^{Q,1,h} M_{\mathrm{V}} \hat{\mathbf{w}}^{(d_i)}, W^{K,1,h} M_{\mathrm{V}} \hat{\mathbf{w}}^{(d_j)} \right\rangle W^{O,1,h} W^{V,1,h} M_{\mathrm{V}} \hat{\mathbf{w}}^{(d_j)} \tag{10}$$

$$\overset{1}{=} \sum_{j=1}^{N} \sum_{h=1}^{H} \overbrace{(M_{\mathrm{V}})_{d_a,d_i}}^{=1} \cdot \overbrace{(M_{\mathrm{V}})_{d_a,d_j}}^{=1} W^{O,1,h} W^{V,1,h} M_{\mathrm{V}} \hat{\mathbf{w}}^{(d_j)} \tag{11}$$

$$\overset{2}{=} \left(\sum_{h=1}^{H} W^{O,1,h} W^{V,1,h}\right) \left(\sum_{j=1}^{N} M_{\mathrm{V}} \hat{\mathbf{w}}^{(d_j)}\right) \tag{12}$$

[2] For simplicity we assume that $r - H$ is odd *i.e.* $d \in \mathbb{N}$, otherwise we can use $\lfloor d \rfloor$.

where (1) is because $W^{Q,1,h} = W^{K,1,h}$ are matrices that are zero everywhere except for entry $(1, d_a)$, and (2) because of linearity. Therefore, for any $j_1, j_2 \in \left[\left(\!\!\binom{(r-1-H)/2}{3^{L-2}}\!\!\right)\right]$ the output of the first self-attention layer on the sequence defined by $\pi_P\left(j_1\right), \pi_Q\left(j_2\right)$ is:

$$\mathbf{y}^{(1,j)} = \left(\sum_{h=1}^{H} W^{O,1,h} W^{V,1,h}\right) \underbrace{\sum_{t=1}^{N/2} \left(M_{\mathrm{V}} \hat{\mathbf{w}}^{\left(\pi_P(j_1)_t\right)} + M_{\mathrm{V}} \hat{\mathbf{w}}^{\left(\pi_Q(j_2)_t\right)}\right)}_{=:\mathbf{u}} \tag{13}$$

where $(p_t)_{t=1}^{N/2} \in P$,$(q_t)_{t=1}^{N/2} \in Q$ is some ordering of P and Q.

Denote by $E \coloneqq \max_{j,\alpha}\left(A_{j,\alpha}\right)$ the maximum entry of A and let $N \geq E \cdot (r - 1 - H)$. Conceptually the mappings π_P, π_Q will divide P, Q into ${(r-1-H)}/{2}$ length E non-overlapping segments, where the α‘th segment will repeat $\hat{\mathbf{w}}^{(\alpha+1)} \quad A_{j,\alpha}$ times and fill the rest with the ”zero” template vector $\hat{\mathbf{w}}^{(1)}$. Thus after the first self-attention layer summation we will get the relevant A’s rows.

Formally, we define the mappings π_P, π_Q as:

$$\forall j \in \left[\left(\!\!\binom{(r-1-H)/2}{3^{L-2}}\!\!\right)\right], t \in [N/2] \quad \left(\pi_P\left(j\right)\right)_{p_t} = \begin{cases} \lfloor t/E \rfloor + 2 & (t-1) \bmod E < A_{j,\lfloor t/E \rfloor + 1} \\ 1 & \text{Otherwise} \end{cases} \tag{14}$$

$$\forall j \in \left[\left(\!\!\binom{(r-1-H)/2}{3^{L-2}}\!\!\right)\right], t \in [N/2] \quad \left(\pi_Q\left(j\right)\right)_{q_t} = \begin{cases} \lfloor t/E \rfloor + 2 + (r-1-H)/2 & (t-1) \bmod E < A_{j,\lfloor t/E \rfloor + 1} \\ 1 & \text{Otherwise} \end{cases} \tag{15}$$

Finally, substituting π_P, π_Q and M_{V} in eq 13 give the desired $\mathbf{u}$:

$$\forall \alpha \in [d_x] \quad \mathbf{u}_\alpha = \begin{cases} A_{j_1,\phi(\alpha)} & (\alpha-1) \bmod d_a < \frac{d_a-1}{2} \wedge \phi(\alpha) \leq (r-1-H)/2 \\ A_{j_2,\phi\left(\alpha - \frac{d_a-1}{2}\right)} & \frac{d_a-1}{2} \leq (\alpha-1) \bmod d_a < d_a - 1 \wedge \phi(\alpha - \frac{d_a-1}{2}) \leq (r-1-H)/2 \\ N & (\alpha-1) \bmod d_a = d_a - 1 \\ 0 & \text{Otherwise} \end{cases}$$

□

2.3. Technical lemmas

The lemma below prove the existence of matrix $A \in \mathbb{N}^{\left(\!\binom{d}{\lambda}\!\right) \times d}$ with constant l^2 row norms, such that the operation of taking the rank d matrix $AA^\top$ to the Hadamard power of λ would result in a fully ranked matrix. Together with corollary 2, this lemma is used in lemma 7 to prove theorem 2 of the main text for the vocabulary based embedding when assuming large N. Note that above lemma is an extension of a direct corollary of the proof in Levine et al. (2020) regarding composition of the self-attention separation rank.

Lemma 8. *For any $d, \lambda \in \mathbb{N}$ there exist $A \in \mathbb{N}^{\left(\!\binom{d}{\lambda}\!\right) \times d}$ with constant l^2 rows norm $c \in \mathbb{N}$ such that:*

$$\operatorname{rank}\left(\left(AA^\top\right)^{\odot\lambda}\right) = \left(\!\!\binom{d}{\lambda}\!\!\right) \tag{16}$$

Proof. We will use the fact that

$$\left(AA^\top\right)^{\odot\lambda} = \sum_{k=1}^{\left(\!\binom{d}{\lambda}\!\right)} \mathbf{a}^{(k)} \otimes \mathbf{b}^{(k)} \tag{17}$$

is of full rank for $\{\mathbf{a}^{(k)}\}_{k=1}^{\left(\!\binom{d}{\lambda}\!\right)}$ and $\{\mathbf{b}^{(k)}\}_{k=1}^{\left(\!\binom{d}{\lambda}\!\right)}$ which are two sets of linearly independent vectors.

For $\alpha, \beta \in [\left(\!\binom{d}{\lambda}\!\right)]$, observing an entry of $\left(AA^\top\right)^{\odot\lambda}$:

$$\left(\left(AA^\top\right)^{\odot\lambda}\right)_{\alpha\beta} = \left(AA^\top\right)^{\lambda}_{\alpha\beta} = \left(\sum_{r=1}^{d} v_r^{(\alpha)} v_r^{(\beta)}\right)^{\lambda} = \tag{18}$$

$$\sum_{k_1+\cdots+k_d=\lambda} \binom{\lambda}{k_1,\ldots,k_d} \left[\prod_{r=1}^{d} \left(v_r^{(\alpha)}\right)^{k_r}\right] \left[\prod_{r=1}^{d} \left(v_r^{(\beta)}\right)^{k_r}\right] \tag{19}$$

where the first equality follows from the definition of the Hadamard power, in the section we denoted $v_r^{(\alpha)}, v_r^{(\beta)}$ as the rth entries in rows α and β of A, and in the second line we expanded the power with the multinomial identity.

Identifying the form of eq. (19) with the schematic form of eq. (17), it remains to find a specific matrix $A \in \mathbb{N}^{\left(\!\binom{d}{\lambda}\!\right) \times d}$ with constant l^2 row norms $c \in \mathbb{N}$ for which the size $\left(\!\binom{d}{\lambda}\!\right)$ set $\left\{\mathbf{a}^{(k_1,\ldots,k_d)}\right\}_{k_1+\cdots+k_d=\lambda}$ is linearly independent, where $a_\alpha^{(k_1,\ldots,k_d)} = \prod_{r=1}^{d} \left(v_r^{(\alpha)}\right)^{k_r}$.

Levine et al. (2020) proved there exists such $B \in \mathbb{R}^{\left(\!\binom{d}{\lambda}\!\right) \times d}$ with $\forall \alpha, \beta \; [B]_{\alpha,\beta} > 0$. Therefore, it is enough to prove that we can approximate B with non-negative rational[3] matrix with normalized rows, while keeping the set $\left\{\mathbf{a}^{(k_1,\ldots,k_d)}\right\}_{k_1+\cdots+k_d=\lambda}$ linearly independent.

To prove this we will arrange the set as the columns of the matrix C, than $\left\{\mathbf{a}^{(k_1,\ldots,k_d)}\right\}_{k_1+\cdots+k_d=\lambda}$ is linearly independent if and only if C's determinant is not zero. Now, C's determinant is polynomial in B entries and therefore from continuity arguments non-zero at neighborhood of B. Finally, this neighborhood contains row normalized rational matrix, since the unit sphere has a dense set of points with rational coordinates (Schmutz, 2008).

□

3. Experimental details

We conducted the network training described in section 5 of the main text with AdamW optimizer with $\beta_1 = 0.9, \beta_2 = 0.999$ and weight decay of 0.01 for $1M$ steps and a batch size of 512 sequences of 128 tokens. All experiments used a learning rate schedule with a 12000 step linear warm-up into $1.6 \cdot 10^{-3}$ followed by a cosine decay to zero and dropout rate of 0.1. In order to increase width without changing other architectural parameters, for all the experiments except of section 5.3 of the main text we kept the number of heads per layer constant at 2 (experimental evidence indicates that many heads per layer are not crucial (Michel et al., 2019; Kaplan et al., 2020), as does (Levine et al., 2020) theoretical analysis which shows that the number of heads per layer affects the separation rank logarithmically).

To verify that our training recipe works, and the model differences are mainly due to expressiveness rather than optimization issues, we constructed a held out test-set of 100 documents from OpenWebText and compare ourselves to GPT-2 (Radford et al., 2019) published models. Table 1 shows our models are on par with the GPT-2 models. Note that our models use shorter context of 128 and that GPT-2 might have trained on our test set, which might explain it's superior perplexity. Nevertheless, this results shows that our training recipe is competitive, and that the model comparisons in the paper are indeed meaningful.

MODEL SIZE	GPT-2 PERPLEXITY	OUR PERPLEXITY
$117M$	21.11	22.78
$345M$	16.03	-
$378M$	-	17.95

Table 1. OpenWebText test set perplexity where total number of tokens (98538 tokens) is according to gpt-2 standard vocabulary, and evaluation done with stride 1 *i.e.*for each token predication, the model use full context of the previous N tokens.

[3] Given such non-negative rational matrix with normalized rows, we can multiply it by the common denominator and get the required $A \in \mathbb{N}^{\left(\!\binom{d}{\lambda}\!\right) \times d}$ with constant l^2 rows norm.

3.1. Rank bottleneck degrades performance

We conducted the network training described in subsection 5.1 of the main text width depth $L = 12$ models. The baselines widths are: $576, 592, 640, 668, 670, 672, 674, 676, 678, 680, 688$. For the low-rank models we factorize the tokens and positional embedding into two matrices of dimensions $d_x \times r$, $r \times V$, as described in the main text. The width in all of this model was set to 680 and the r's were: $16, 32, 64, 96, 128, 256, 344, 400, 456, 512, 600, 680, 880, 1080$. Table 2 show the estimated the standard deviation of the test loss of this experiment by repeating the training 5 times. For $r \in \{16, 32, 64\}$ we also trained variant with full-rank positional embedding and achieve losses that are within 2-std of the factorized positional embedding ones.

Table 2. The standard deviation of the test loss for several experiments of subsection 5.1 in the main text, when repeating the training and evaluation experiment 5 times per point.

d_x	r	STD
680	680	$8 \cdot 10^{-4}$
576	576	$1.5 \cdot 10^{-3}$
680	128	$2.1 \cdot 10^{-3}$

3.2. Vocabulary affects the depth-to-width interplay

We tokenized the training and test corpus the with the GPT-2 (Radford et al., 2019) vocabulary, and additional 3 BPE vocabularies of sizes $V = 257, 500, 2000$ trained on our training corpus using to huggingface tokenizers[4] library.

We conducted the network training described in section 5.2 of the main text width depth $L \in \{24, 48\}$ models with width detailed in table 3.

Table 3. The widths d_x of the different trained networks.

V	L	r	WIDTHS
257	24		$144, 160, 168, 184, 192, 200, 224, 248, 264, 280, 336, 408, 480$
257	48		$104, 112, 120, 128, 136, 142, 144, 160, 176, 184, 200, 240, 288, 336$
500	24		$280, 336, 360, 384, 408, 424, 440, 480, 504, 528, 544, 576, 600$
500	48		$200, 240, 272, 288, 296, 312, 336, 352, 376, 384, 408, 424$
2000	24	500	$280, 336, 408, 448, 480, 504, 528, 544, 576, 600, 628$
2000	48	500	$200, 240, 288, 320, 336, 352, 376, 384, 408, 424, 440$
2000	24		$200, 224, 248, 280, 336, 408, 480, 544, 704, 744, 792, 848, 1064$
2000	48		$144, 160, 176, 200, 240, 288, 336, 384, 496, 528, 560, 600, 752$
50257	24		$408, 480, 544, 592, 656, 704, 744, 792, 848, 1064$
50257	48		$336, 384, 432, 480, 512, 544, 576, 616, 768$

Table 4. The standard deviation of the test loss for several experiments of subsection 5.2 in the main text, when repeating the training and evaluation experiment 5 times per point.

V	L	d_x	r	STD
257	24	264		$5.1 \cdot 10^{-4}$
257	48	184		$6.8 \cdot 10^{-4}$
500	24	504		$1.7 \cdot 10^{-3}$
2000	24	528	500	$1.5 \cdot 10^{-3}$
50257	48	480		$3.1 \cdot 10^{-3}$

Beyond the experiments described in subsection 5.2 of the main text, we conduct additional experiment to verify that when the vocabulary size exceed the network width and does not constitutes a bottleneck, it has negligible effect on the "depth-efficiency" point.

[4] https://huggingface.co/docs/tokenizers/python/latest/

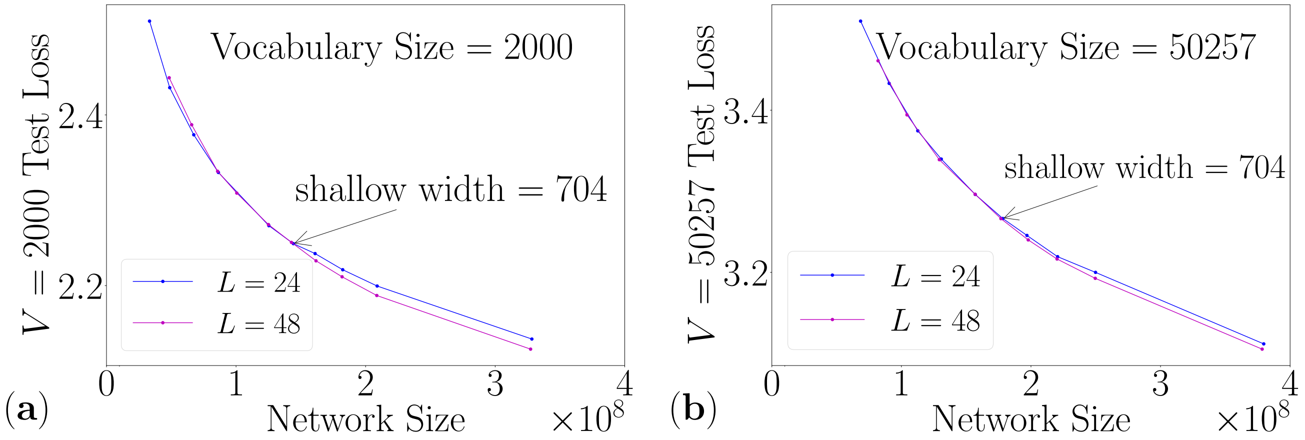

Figure 1. Unlike figure 1 of the main text, when the vocabulary size exceed the network width and does not constitutes a bottleneck, the vocabulary size has negligible effect on the "depth-efficiency" point . Note that the "depth-efficiency" point of $V = 50257$ occur at larger network size since when V grows the embedding matrix size become non negligible and correspond to 20% of the network parameters. Nevertheless, the shallow network width at the "depth-efficiency" point is very similar to the $V = 2000$ case.

Figure 1 shows that when repeating subsection 5.2 experiment with GPT-2 (Radford et al., 2019) vocabulary that is ~ 25 times larger, the "depth-efficiency" point are very similar to the $V = 2000$ case. This is directly in line with the vocabulary bottleneck prediction since the largest network width in this experiment is 1064, and clearly the $V = 2000$ vocabulary does not constitutes a width bottleneck.

3.3. Width bottlenecks the attention dimension

Unlike the rest of the experiments in this paper, the experiments described in subsection 5.3 of the main text done with larger amount of heads, mostly $H = 12$ to avoid low-rank Key, Query, Value and Output matrices when increasing the $^{H \cdot d_a}/_{d_x}$ ratio. Since the and the optimal depth per network might vary when changing $^{H \cdot d_a}/_{d_x}$ ratio we choose for each ratio the best depth in $\{12, 18, 24\}$. Table 5 give the exact details of the networks that appear in figure 4 of the main text. Figure 2 shows that the performance difference between values of the bottleneck ratio is larger than the variation between different depths per bottleneck ratio.

Table 5. Details of all architecture for each $^{H \cdot d_a}/_{d_x}$ ratio in figure 2, the bold depth are the ones showed in figure 4 of the main text.

$^{H \cdot d_a}/_{d_x}$	L	H	WIDTHS
1	**18**	12	$360, 384, 420, 456$
1	**24**	12	$420, 456, 480$
2	**12**	12	$408, 480$
2	18	18	396
2	24	24	336
4	12	12	$288, 336$
4	18	12	$240, 276$
4	**24**	12	$204, 240$
8	**12**	12	$204, 240$
8	18	12	$168, 192$
8	24	12	$144, 168$
16	**12**	24	$144, 168$

3.4. Low-rank positional embedding

In this subsection, we show that a low rank positional embedding matrix has a negligible effect on the model loss, when compared with the effect of decreasing the vocabulary rank. This justifies both the practical use of rank-1 positional

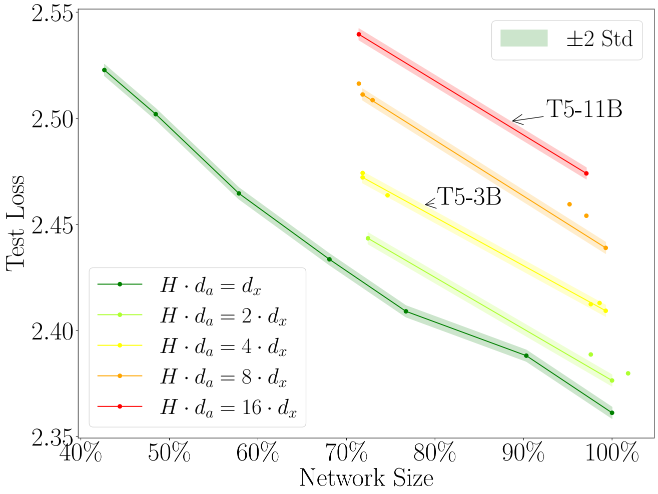

Figure 2. The performance difference between values of the bottleneck ratio is larger than the variation between different depths per bottleneck ratio. The lines are the points of figure 4 in the main text, and the circles are another depths detailed in table 5.

embedding matrices in leading models such as T5 (Raffel et al., 2020), and the assumption in theorems 1 and 2.

We trained depth $L = 12$ width $d_x = 512$ networks with vocabulary size $V = 2000$, sequences of 512 tokens and positional embedding ranks of $\{1, 2, 4, 8, 16, 32, 64, 128, 256, 512\}$, In addition, we compared to baseline networks of vocabulary ranks[5] $\{2, 4, 8, 16, 32, 64, 128, 256, 512\}$ with full-rank positional embedding.

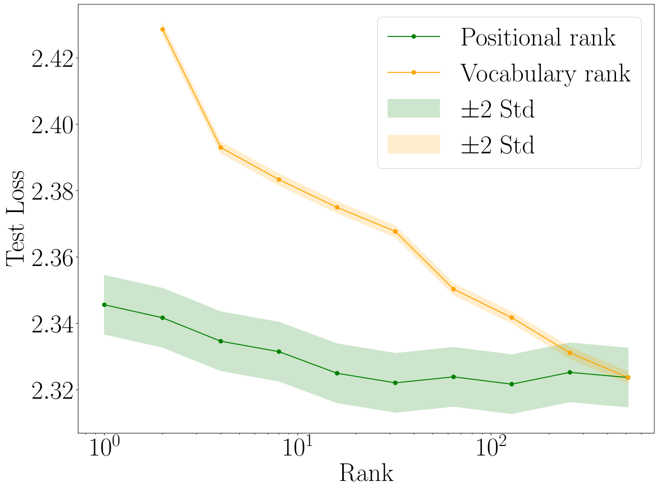

Figure 3. Low-rank positional embedding as assumed in theorem 1 when compared to low-rank due to vocabulary size has negligible effect on the model. For example even positional rank of 16 perform on par with full positional embedding.

Figure 3 shows that when decreasing the positional embedding rank down to 16, performance is on par with full-rank positional embedding (within 2 std). Moreover, even the extreme low-rank positional embedding of rank-1 reaches a loss of much higher vocabulary rank (between 64 and 128), thus justifying the assumption in theorems 1 and 2. Note that we have shown in section 5.1 of the main text that the practical effect of the predicted vocabulary bottlenecking phenomenon is comparable to a substantial reduction in model size.

[5] We did not compare to rank 1 vocabulary network, since we observed dramatic performance degradation in this case that are probably due to unrelated bottleneck.

References

Hackbusch, W. *Tensor spaces and numerical tensor calculus*, volume 42. Springer Science & Business Media, 2012.

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Levine, Y., Wies, N., Sharir, O., Bata, H., and Shashua, A. Limits to depth efficiencies of self-attention. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 22640–22651. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/ff4dfdf5904e920ce52b48c1cef97829-Paper.pdf.

Michel, P., Levy, O., and Neubig, G. Are sixteen heads really better than one? In *Advances in Neural Information Processing Systems*, pp. 14014–14024, 2019.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. Language models are unsupervised multitask learners. 2019.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

Schmutz, E. Rational points on the unit sphere. *Central European Journal of Mathematics*, 6(3):482–487, 2008.