Sequence Length Independent Norm-Based Generalization Bounds for Transformers

Jacob Trauger
University of Michigan

Ambuj Tewari
University of Michigan

Abstract

This paper provides norm-based generalization bounds for the Transformer architecture that do not depend on the input sequence length. We employ a covering number based approach to prove our bounds. We use three novel covering number bounds for the function class of bounded linear mappings to upper bound the Rademacher complexity of the Transformer. Furthermore, we show this generalization bound applies to the common Transformer training technique of masking and then predicting the masked word. We also run a simulated study on a sparse majority data set that empirically validates our theoretical findings.

1 INTRODUCTION

Since Vaswani et al. (2017) debuted the Transformer, it has become one of the most preeminent architectures of its time. It has achieved state of the art prediction capabilities in various fields (Dosovitskiy et al., 2020; Wu et al., 2022; Vaswani et al., 2017; Pettersson and Falkman, 2023) and an implementation of it has even passed the BAR exam (Katz et al., 2023). With such widespread use, the theoretical underpinnings of this architecture are of great interest.

Specifically, this paper is concerned with bounding the generalization gap when using the Transformer in supervised learning. These upper bounds can be used to help understand how sample size needs to scale with different architecture parameters and they are a very common theoretical tool to understand machine learning algorithms (Kakade et al., 2008; Garg et al., 2020; Truong, 2022; Lin and Zhang, 2019).

One such architecture parameter is the sequence length of the input. Since the input of Transformers can be thought of as a sequence of tokens (e.g. a sequence of word embeddings), the maximum allowable length of an input sequence is called the sequence length.

The main contribution of this paper is providing norm-based generalization bounds for the Transformer architecture that have no explicit dependence on the input sequence length. We also contribute 3 novel vector valued linear mapping covering number bounds that are the key to obtain our generalization bounds. Furthermore, we give an example of a regime where our bounds apply and we give empirical evidence that our theory holds.

Previously, the best known norm-based generalization bound scaled with the logarithm of the sequence length (Edelman et al., 2022). Removing the dependence on sequence length leads to more intuitively appealing bounds since the total number of parameters in the Transformer is independent of input sequence length. Since our bounds are norm-based they have potential to provide meaningful guarantees even in overparameterized regimes where parameter counting bounds might be less meaningful.

We are able to show this by going through the Rademacher complexity of the Transformer and then using three novel linear covering number bounds to bound the Rademacher complexity. Therefore, Section 1 goes over the necessary background needed. Section 2 shows the novel covering number bounds for a linear mapping function class with bounded matrices and inputs. In Section 3 we start dealing specifically with Transformers. Here we show a new Rademacher complexity bound for a single

Proceedings of the 27$^{\text{th}}$ International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238. Copyright 2024 by the author(s).

layer Transformer. Section 4 provides details on how our covering number bounds can be used in the multi-layer analysis of Edelman et al. (2022) to get a sequence length independent norm-based generalization bound. Section 5 shows how a method of training used in BERT (Devlin et al., 2018) can be reduced to what is studied in this paper. Finally, in Section 7 we show an experiment on a simulated sparse majority data set to empirically validate our theoretical findings.

1.1 Related Works

This paper is most closely related to the work of Edelman et al. (2022) who prove a norm-based generalization bound that grows logarithmically with sequence length. Due to this, they state Transformers have an inductive bias to represent a sparse function of the inputs. We bolster this claim further by removing the dependence on sequence length altogether.

Another result similar to our is given by Zhang et al. (2022). They are able to remove the dependence on sequence length. However, the bound they get, which we shall call a parameter counting-based bound, has several drawbacks which we discuss in Section 2.5. Wei et al. (2022) also show generalization bounds for Transformers, but specifically study binary classification setting with 0-1 loss and use a margin approach. Fu et al. (2023) freeze some of the weight matrices at initialization and bound the excess risk in this setting as a function of the amount of heads in the attention layer. This paper's bound also do not depend on sequence length.

Outside of Transformers, using Rademacher complexity in deep learning to bound the generalization gap has a rich history. Golowich et al. (2018) was able to use Rademacher complexities to get a generalization bound independent of the depth and width of a neural network. Truong (2022) is able to use Rademacher complexity to get nearly tight bounds on neural networks under some assumptions on the data. Also, Bartlett et al. (2017) use covering numbers and Rademacher complexity to get generalization bounds on multiclass neural networks using a margin based approach.

2 BACKGROUND

2.1 Matrix Definitions

For our matrix notation we will, in general, use capital letters for matrices and lowercase letters for vectors. For a matrix W, we will use $W_{:,i}$ to denote the i^{th} column of W and W_i to denote the i^{th} row unless otherwise noted.

Now, we will define a few well-used matrix norms. Let $p, q, r, d \in \mathbb{N}$ and let $W \in \mathbb{R}^{r \times d}$. The first norm, denoted as $\|W\|_{q,p}$, will be defined as $\|W\|_{q,p} = \left\| \left[\|W_{:,1}\|_q, \dots \|W_{:,d}\|_q \right]^\top \right\|_p$.

Another matrix norm, also known as the operator norm, we will denote as $\|W\|_{q\to p}$. This one is defined as:

$$\|W\|_{q\to p} = \sup_{x \in \mathbb{R}^d \setminus 0} \frac{\|Wx\|_p}{\|x\|_q}$$

One final matrix norm we will review is the Frobenius norm, denoted as $\|W\|_F$, which is defined as:

$$\|W\|_F = \sqrt{\sum_{i=1}^{r} \sum_{j=1}^{d} W_{ij}^2}$$

We will also denote $\|W\|_{2\to 2}$ as $\|W\|_2$. A well known property of the $\|\cdot\|_{2\to 2}$ operator is that it is equal to the largest singular value of the input matrix. The Frobenius norm is also well known to be equal to the the square root of the squared sum of the singular values of a matrix. Therefore, we have $\|W\|_{2\to 2} \leq \|W\|_F$ for any matrix W.

2.2 Generalization Bounds

When training machine learning algorithms, we can only use a finite amount of data to learn from, however, we want our resulting function to generalize well outside of our training sample. Thus, having guarantees with high probability on the difference between the loss on our training sample and the loss on our testing population is extremely important. Generalization bounds try to upper bound this loss gap.

Mathematically, if we have a hypothesis class $\mathcal{H}$, sample space $\mathcal{X}$, label space $\mathcal{Y}$, loss function ℓ, and distribution over the sample and label space $\mathcal{D}$, then our generalization gap for a set of samples and labels $S = \{(x_i, y_i)\}_{i=1}^n$, $x_i \in \mathcal{X}$, $y_i \in \mathcal{Y}$, on the hypothesis $h \in \mathcal{H}$ is defined to be

$$\left| \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\ell(h(x), y) \right] - \frac{1}{n} \sum_{i=1}^{n} \ell(h(x_i), y_i) \right|$$

Notice how if we can have this value go to 0 with high probability over all sets of samples and for all $h \in \mathcal{H}$, then we can be confident that minimizing the sample loss will not impact our generalization.

2.3 Rademacher Complexity

One such tool that can be used to upper bound the generalization gap is the Rademacher complexity. Let us have the same set up as in the previous section. Then the Rademacher complexity of a hypothesis class $\mathcal{H}$ is defined to be

$$Rad_n(\mathcal{H}, S) = \frac{1}{n}\mathbb{E}_\sigma\left[\sup_{h\in\mathcal{H}}\sum_{i=1}^{n}\sigma_i h(x_i)\right]$$

where each σ_i are i.i.d. and take values ± 1 each with half probability and $\sigma = (\sigma_1, \ldots, \sigma_n)$. It is well known that (Shalev-Shwartz and Ben-David, 2014), if the magnitude of our loss function is bounded above by c, with probability greater than $1-\delta$ for all $h \in \mathcal{H}$, we have

$$\left|\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\ell(h(x), y)\right] - \frac{1}{n}\sum_{i=1}^{n}\ell(h(x_i), y_i)\right| \leq$$
$$2Rad_n(\ell\circ\mathcal{H}, S) + 4c\sqrt{\frac{2\log(4/\delta)}{m}}$$

where $\ell\circ\mathcal{H} = \{(x, y) \mapsto \ell(h(x), y) \mid h \in \mathcal{H}\}$. Therefore, if we have an upper bound on the Rademacher complexity, we can have an upper bound on the generalization gap.

2.4 Covering Numbers

The use of covering numbers is one such way we can bound the Rademacher complexity of a hypothesis class. Let $q \in \mathbb{R}_{>0}$ and let us have a function class $\mathcal{F}$, $\forall f \in \mathcal{F}\ f : \mathbb{R}^d \to \mathbb{R}^k$. We say a subset $\hat{\mathcal{F}} \subset \mathcal{F}$ covers a set of inputs $\{x_i\}_{i=1}^n$ if for every $f \in \mathcal{F}$, $\exists \hat{f} \in \hat{\mathcal{F}}$ such that $\sup_{x_i}\left\|f(x_i) - \hat{f}(x_i)\right\|_q < \epsilon$. We will use the notation $N_\infty(\mathcal{F}, \epsilon, \{x_i\}_{i=1}^n, \|\cdot\|_q)$ for this. We will define the covering number, $N_\infty(\mathcal{F}, \epsilon, n, \|\cdot\|_q)$ as

$$\sup_{\{x_i\}_{i=1}^n} N_\infty(\mathcal{F}, \epsilon, \{x_i\}_{i=1}^n, \|\cdot\|_q)$$

It has been shown that for scalar valued hypothesis classes, the Rademacher complexity can be upper bounded using the covering number of the hypothesis class (Dudley, 1967). Using a slightly modified version, we have for a constant c:

$$Rad_n(f, S) \leq c\inf_{\delta\geq 0}\left(\delta + \int_\delta^\infty\sqrt{\frac{\log N_\infty(\mathcal{F}, \epsilon, n)}{n}}d\epsilon\right)$$

Notice that if we have a bounded function class, then the ∞ in the integral limit can become the upper bound of the function class.

2.5 Two Types of Bounds

Suppose our inputs have dimension d and the inputs have an upper norm bound of B_x. Suppose our matrices have dimension $k \times d$ and have an upper norm bound of B_w. Here we will note two different types of generalization bounds that can one can arrive at depending on the perspective. One is where you have the norm bounds inside a log term and the dimensions on the outside (e.g., $O(\sqrt{dk\log(B_xB_w)})/n)$. The other is where you have the bounds outside the log term and parameters inside (e.g., $O(\sqrt{B_xB_w\log(dk)})/n)$. We will refer to the former as parameter-counting type bounds and the latter as the norm-based type bounds. We note that these are not the only types of bounds, just these are the ones relevant to this paper.

Notice how, for parameter counting bounds, over-parameterized architectures can lead to vacuous bounds. Also notice that parameter counting bounds do not take full advantage of SGD's implicit regularization since the norms are within the logarithm. In contrast, as long as the norm bounds are reasonable, norm-based generalization bounds work well with over-parameterized architectures and work well with implicit regularization.

For Transformers, there are norm-based bounds that scale logarithmically with sequence length (Edelman et al., 2022) and parameter counting bounds that do not scale with sequence length (Zhang et al., 2022). This is a gap in the literature which this paper intends to fill.

2.6 Self-Attention and Transformers

We will follow the definition of self-attention and Transformers set forth by Edelman et al. (2022) and keep with their notation.

Let $W_c \in \mathbb{R}^{k\times d}$, $W_v \in \mathbb{R}^{d\times k}$, and $W_Q, W_K \in \mathbb{R}^{d\times T}$ be trainable weight matrices. Let $X \in \mathbb{R}^{T\times d}$ be the input, which can be thought of as sequence of T d-dimensional tokens. Also, let σ be a L_σ-Lipshitz activation function that is applied elementwise and has the property $\sigma(0) = 0$. Finally, let RowSoftmax donote applying softmax on each row of its input. Then, they define a Transformer head as

$$\sigma\left(\text{RowSoftmax}\left(XW_QW_K^\top X^\top\right)XW_v\right)W_c$$

Since W_Q and W_K are only ever multiplied with each other, we will combine $W_QW_K^\top$ into a single matrix $W_{QK} \in \mathbb{R}^{d\times d}$ for convenience of analysis. Once we do this, note that the total dimensionality (of W_{QK}, W_c, W_v) is independent of T, the sequence

length which counts how many tokens are in each sample. The embedding dimension is d since it is the dimension that the values in the sequence are embedded into and k is the hidden dimension.

For multi-head Transformers, we assume each head is summed up at the end of each layer. That is, the output for a layer of a multi-head Transformer is:

$$\sum_{h=1}^{H} \sigma\left(\text{RowSoftmax}\left(XW_{h,Q}W_{h,K}^{\top}X^{\top}\right)XW_{h,v}\right)W_{h,c}$$

Note how the ouput of a layer can be used as the input to another layer. This is how multi-layer Transformer networks are created. Standard practice is to add layer normalization in between each layer as this has been well studied to aid in optimization and generalization (Ba et al., 2016; Wang et al., 2019; Xu et al., 2019). Thus, keeping with the definitions and notation previously set forth, we will inductively define an L-layer Transformer block:

Let $\mathcal{W}^{(i)} = \{W_v^{(i)}, W_c^{(i)}, W_{QK}^{(i)}\}$ and let $\mathcal{W}^{1:i} = \{\mathcal{W}^1, \dots \mathcal{W}^{(i-1)}\}$. Also, let

$$f(X; W^{(i)}) = \text{RowSoftmax}\left(XW_Q^{(i)}W_K^{(i)\top}X^{\top}\right)XW_v^{(i)}$$

$$g_{block}^1\left(X; W^{1:1}\right) = X$$

Then, the output of the i^{th} layer is defined to be

$$g_{block}^{(i+1)}\left(X; W^{1:i+1}\right) =$$
$$\Pi_{norm}\left(\sigma\left(\Pi_{norm}\left(f\left(g_{block}^{(i)}\left(X; W^{1:i}\right); W^{(i)}\right)\right)\right)W_c^{(i)}\right)$$

where Π_{norm} projects each row onto the unit ball.

In our analysis we will focus on the scalar output setting for Transformers. Specifically, we will follow how BERT trains for scalar output (Devlin et al., 2018). In order to get a scalar output, we add an extra input in the sequence that can be constant or trained. Let this index be the $[CLS]$ index. Let us also have a trainable vector $w \in \mathbb{R}^d$. Then, at the last layer, take the output at the $[CLS]$ index, $Y_{[CLS]} \in \mathbb{R}^d$ and multiply it with w to get our output $w^{\top}Y_{[CLS]} \in \mathbb{R}$.

3 LINEAR COVERING NUMBER BOUNDS

In this section we will show three different covering number bounds for linear function classes with different restrictions on input and matrix norms. To show the first one, we need the following lemmas, the first one is attributed to Maurey (Pisier, 1981) and first used in this context by Zhang (2002):

Lemma 3.1. *(Maurey's Sparsification Lemma) Let $\mathcal{H}$ be a Hilbert space and let each $f \in \mathcal{H}$ have the representation $f = \sum_{i=1}^{d} \alpha_i V_i$ where $V_i \in \mathcal{H}$, $\|V_i\| \leq b$ and $\alpha_i \geq 0$ with $\gamma = \|\alpha\|_1 \leq 1$. Then, for any $k \in \mathbb{N}$, there exist $k_1, \dots, k_d$, $k_i \in \mathbb{Z}_{\geq 0}$, $\sum_{i=1}^{d} k_i \leq k$, such that*

$$\left\| f - \frac{1}{k}\sum_{i=1}^{d} k_i V_i \right\|_2^2 \leq \frac{\gamma^2 b^2 - \|f\|^2}{k}$$

We note that the total amount of $(k_1, \dots k_d)$'s that fit the criteria above is less than or equal to d^k. This has been used to upper bound the covering number for linear functions (Zhang, 2002; Bartlett et al., 2017) and we will use it similarly in our proofs.

For covering a class of scalar valued linear functions $\{x \to w^{\top}x \mid w \in \mathbb{R}^d$,w norm bounded$\}$ with inputs $\{x_i \in \mathbb{R}^d\}_{i=1}^n$, it is known that we are able to remove the dependence on n and replace it with a dependence on d. Kontorovich and Attias (2021) show this and attribute it to folklore. Given a cover such as this, it is an immediate extension to create a non-n-dependent cover of the function class $\{x \to Wx, W \in \mathcal{W}\}$ as long as each row in each $W \in \mathcal{W}$ is bounded by the norm restraint needed for the scalar valued cover (specific details are in appendix section A).

Our first contribution generalizes the above by allowing us to consider more possible norm bounds on $\mathcal{W}$. It shows that, under certain norm restrictions for our input, the linear mappings covering number for any set of size N is equivalent to the covering number on the appropriately scaled standard basis (proof in Appendix Section B).

Lemma 3.2. *Let $\mathcal{W} \subset \mathbb{R}^{k \times d}$ and let $\mathcal{F} = \{x \to Wx \mid W \in \mathcal{W}\}$ with $\|x\|_1 \leq B_x$. Then, for $N \geq d$, we have*

$$\mathcal{N}_{\infty}\left(\mathcal{F}, \epsilon, N, \|\cdot\|_q\right) = \mathcal{N}_{\infty}\left(\mathcal{F}, \epsilon, \{B_x e_1, \dots, B_x e_d\}, \|\cdot\|_q\right)$$

3.1 Log Covering Number For $\|\cdot\|_1$ Bounded Input and $\|\cdot\|_{1,\infty}$ Bounded Matrix

Now, we will show our three covering number bounds. Each of the three have different norm bounds on the inputs and the matrices, allowing for flexibility when deciding which to use.

Lemma 3.3. *Let $N \geq d$, $\mathcal{W} = \{W \in \mathbb{R}^{k \times d} \mid \|W\|_{1,\infty} \leq B_w\}$, $\mathcal{F} = \{x \to Wx \mid W \in \mathcal{W}\}$, and let our inputs $x \in \mathbb{R}^d$ have the restriction $\|x\|_1 \leq B_x$.*

Then:

$$\log \mathcal{N}_\infty\left(\mathcal{F}, \epsilon, N, \|\cdot\|_2\right) \leq \frac{dB_w^2 B_x^2}{\epsilon^2} \log(2k+1)$$

Proof. We will abuse notation slightly by referring to $\mathcal{W}$ at times instead of $\mathcal{F}$ (and similar for $\hat{\mathcal{W}}$ and $\hat{\mathcal{F}}$).

Let us have the set $V = \{v \in \mathbb{R}^k \mid \|v\|_1 \leq B_w\}$. Then notice by lemma 3.1 we have that there exists an ϵ / B_x cover for V that has log size

$$\frac{B_w^2 B_x^2}{\epsilon^2} \log(2k+1)$$

Let this cover be $\hat{V}$. We claim that

$$\hat{\mathcal{W}} = \underbrace{\hat{V} \otimes \hat{V} \cdots \otimes \hat{V}}_{d \text{ total times}}$$

is a cover for $\mathcal{W}$.

To show this, let $W \in \mathcal{W}$ and let $\hat{W} \in \hat{\mathcal{W}}$ be the one where each column is the vector that would be chosen to cover the corresponding column in W. Then, notice for all $i \in [d]$ we have

$$\left\|(W - \hat{W}) B_x e_i\right\| = B_x \left\|W_{:,i} - \hat{W}_{:,i}\right\| \leq B_x \frac{\epsilon}{B_x} = \epsilon$$

Therefore

$$\sup_{i \in [d]} \left\|(W - \hat{W}) B_x e_i\right\| \leq \epsilon$$

which, by lemma 3.2 shows that

$$\log \mathcal{N}_\infty\left(\mathcal{F}, \epsilon, N, \|\cdot\|_2\right) \leq \frac{dB_w^2 B_x^2}{\epsilon^2} \log(2k+1)$$

□

3.2 Log Covering Number For $\|\cdot\|_1$ Bounded Input and $\|\cdot\|_{2,1}$ Bounded Matrix

Lemma 3.4. *Let $N > d$, $\mathcal{W} = \{W \in \mathbb{R}^{k \times d} \mid \|W\|_{2,1} \leq B_w\}$, $\mathcal{F} = \{x \to Wx \mid W \in \mathcal{W}\}$, and let our inputs $x \in \mathbb{R}^d$ have the restriction $\|x\|_1 \leq B_x$. Then:*

$$\log \mathcal{N}_\infty\left(\mathcal{F}, \epsilon, N, \|\cdot\|_2\right) \lesssim \frac{B_w^2 B_x^2}{\epsilon^2} \log(dk)$$

where $\lesssim$ hides logarithmic dependencies except T, k, d.

Proof. This proof uses Lemma 4.6 in Edelman et al., 2022 Edelman et al. (2022), which is rewritten below for clarity.

Lemma 3.5. *(Edelman et al., 2022 Lemma 4.6) Let $\mathcal{W}$ and $\mathcal{F}$ be as above. Then for any set of points $x_1, \ldots, x_n \in R^d$ with $\|x_i\|_2 \leq B_x$ for all i, we have*

$$\log \mathcal{N}_\infty\left(\mathcal{F}, \epsilon, \{x_i\}_{i=1}^n, \|\cdot\|_2\right) \lesssim \frac{B_w^2 B_x^2}{\epsilon^2} \log(dn)$$

With this, let $\hat{\mathcal{W}}$ be an ϵ-cover for $\mathcal{W}$ over the inputs $\{B_x e_i\}_{i=1}^d$ as stated in the lemma. Then, by lemma 3.2, we have that the cardinality of $\hat{\mathcal{W}}$ is also an upper bound for $\mathcal{N}_\infty\left(\mathcal{F}, \epsilon, N, \|\cdot\|_2\right)$, which is what we needed to show. □

3.3 Log Covering Number For $\|\cdot\|_2$ Bounded Input and $\|\cdot\|_{1,1}$ Bounded Matrix

Lemma 3.6. *Let $N \geq d$, $\mathcal{W} = \{W \in \mathbb{R}^{k \times d} \mid \|W\|_{1,1} \leq B_w\}$, $\mathcal{F} = \{x \to Wx \mid W \in \mathcal{W}\}$, and let our inputs $x \in \mathbb{R}^d$ have the restriction $\|x\|_2 \leq B_x$. Then:*

$$\log \mathcal{N}_\infty\left(\mathcal{F}, \epsilon, N, \|\cdot\|_2\right) \leq \frac{B_x^2 B_w^2}{\epsilon^2} \log(2dk+1)$$

Proof. Let $\mathcal{V}$ be the set of all the flatten matrices in $\mathcal{W}$. Note how this implies $\forall v \in \mathcal{V}$, we have $\|v\|_1 \leq B_w$. Then, by Maurey's sparification lemma, we have that there exists an ϵ-cover $\hat{\mathcal{V}}$ of log size at most $\frac{B_w^2}{\epsilon^2} \log(2dk+1)$. We claim if we unflatten $\hat{\mathcal{V}}$ (call this $\hat{\mathcal{W}}$), then $\hat{\mathcal{W}}$ is a $(B_x \epsilon)$-cover for $\mathcal{W}$. Let $W \in \mathcal{W}$, let V be the flatten version of W. Then, let $\hat{V}$ be the flattened vector we would choose for V in our cover and let $\hat{W} \in \hat{\mathcal{W}}$ be the unflattened version of $\hat{V}$. Notice for any $x \in \mathbb{R}^d$, $\|x\|_2 \leq B_x$:

$$\begin{aligned}
&\left\|Wx - \hat{W}x\right\|_2 \leq \left\|W - \hat{W}\right\|_{2 \to 2} \|x\|_2 \leq \\
&\left\|W - \hat{W}\right\|_F \|x\|_2 = \left\|V - \hat{V}\right\|_2 \|x\|_2 \leq \\
&\left\|V - \hat{V}\right\|_2 B_x \leq B_x \epsilon
\end{aligned}$$

Therefore our covering number is at most $\frac{B_x^2 B_w^2}{\epsilon^2} \log(2dk+1)$ □

3.4 Observations on Results

Above we have showed a few different sharpenings of linear covering numbers with matrices instead of vectors. Specifically, these do not rely on the sample size of the input. Also, all but lemma 3.3 keeps the matrix dimensions inside the log term. We do note however, the matrix bound in lemma 3.3 is a $1, \infty$ norm bound while the others are rather $2, 1$

or $1,1$ norm bound. We know that for any matrix W, $\|W\|_{p,1} \leq d\|W\|_{p,\infty}$. Thus if we were to convert lemmas 3.4 and 3.6 into a norm bound on $2,\infty$ and $1,\infty$ bounds we would have a $d^2B_w^2$ term. This shows that lemma 3.3 is a stronger bound than it lets on.

4 TRANSFORMER RADEMACHER COMPLEXITY

4.1 Analysis for Single Layer Transformer

Let $w \in \mathbb{R}^d$, $W_c \in \mathbb{R}^{k\times d}$, $W_v \in \mathbb{R}^{d\times k}$, $W_{QK} \in \mathbb{R}^{d\times d}$ $\|w\|_1 \leq B_w$, $\left\|W_c^\top\right\|_{1,\infty} \leq B_{W_c}$, and $\left\|W_v^\top\right\|_{1,\infty} \leq B_{W_v}$. Then we have our scalar one layer Transformer as $w^\top Y_{[CLS]}$ where

$$Y_{[CLS]} = W_c^\top \sigma\left(W_v^\top X^\top \text{softmax}\left(XW_{QK}^\top x_{[CLS]}\right)\right)$$

Our Rademacher complexity is thus the following:

$$\mathbb{E}\left[\sup_{w,W_c,W_v,W_{QK}} \sum_{i=1}^m \epsilon_i w^\top W_c^\top \sigma\left(W_v^\top X_{(i)}^\top \text{softmax}\left(X_{(i)}W_{QK}x_{[CLS]}\right)\right)\right]$$

With this, we have the following theorem:

Theorem 4.1. *Suppose we have the required norm restrictions denoted above along with $\|x\|_2 < B_x$ $\forall x \in \mathcal{X}$. Also, suppose we have a covering number bound in the form of C/ϵ^2 for the class $\{x \to Wx \mid x \in \mathcal{X}, w \in \mathcal{W}\}$ where $\mathcal{X}$ and W_{QK} meets these requirements needed as well. Finally, if we have $m > d$ and $m > \ln(2d)$. Then, an upper bound on the Rademacher complexity of a single layer Transformer layer is:*

$$O\left(B_w B_{W_c} L_\sigma B_{W_v}\left(\frac{2B_x^2\sqrt{C}}{\sqrt{m}}\left(1+\ln\left(\frac{\sqrt{m}}{2B_x\sqrt{C}}\right)\right)+B_x\sqrt{\frac{\ln(2d)}{m}}\right)\right)$$

The proof is left in Appendix Section C for ease of presentation.

We can now take lemmas 3.3, 3.4, and 3.6 to get bounds on the Rademacher complexity. In the proof it can be seen the only matrix that needs to be covered in W_{QK}, thus we will only have a dependence on d and not k. We will show one corollary below and leave the rest to Appendix Section D.

Corollary 4.1.1. *Let us have the requirements needed for Theorem 4.1 along with $\|W_{QK}\|_{1,1} \leq B_{W_{QK}}$ Let*

$$B = B_w B_{W_c} L_\sigma B_{W_v}$$

and let

$$\alpha = B_{W_{QK}}\sqrt{2\log(2d^2+1)}$$

Then we have our Transformer Rademacher complexity being less than

$$O\left(B\left(\frac{B_x^3\alpha}{\sqrt{m}}\left(1+\ln\left(\frac{\sqrt{m}}{B_x^2\alpha}\right)\right)+B_x\sqrt{\frac{\ln(2d)}{m}}\right)\right)$$

4.2 Single Layer Multiple Heads

Let $H \in \mathbb{N}$ and let Y_j, $j \in [H]$ each be a Transformer head. Then notice by linearity of expectation:

$$\mathbb{E}\left[\sup_{Y_1,\ldots,Y_H} \sum_{i=1}^m \epsilon_i w^\top \sum_{j=1}^H Y_j(X_i)\right] = \sum_{j=1}^H \mathbb{E}\left[\sup_{Y_j} \sum_{i=1}^m \epsilon_i w^\top Y_j(X_i)\right]$$

The expectation above is the same as the single layer, thus multiple heads just adds a linear H term to the Rademacher complexity.

4.3 Multiple Layers

We have seen that we are able to get sequence length independent Rademacher complexities (and therefore generalization bounds) for a single layer Transformer architecture. For multiple layers, it suffices to take the proof found in Edelman et al. (2022) and slightly rework it so that it will work for an arbitrary linear covering number bound.

Theorem 4.2. *(Slight Reworking of Theorem A.17 in Edelman et al., 2022) Suppose we have a log covering numbers in the form of C_1/ϵ^2 and C_{B_x}/ϵ^2 for the function class $\{x \to Wx \mid x \in \mathcal{X}, w \in \mathcal{W}\}$ where $\|x\|_2 \leq 1$ $\forall x \in \mathcal{X}$ and $\|x\|_2 \leq B_x$ $\forall x \in \mathcal{X}$ respectively. Suppose we also have $\left\|W_c^{(i)\top}\right\|_2 \leq B_{c2}$, $\left\|W_v^{(i)\top}\right\|_2 \leq B_{v2}$, $\left\|W_{QK}^{(i)}\right\|_2 \leq B_{QK2}$, $\|w\| \leq B_w$ along with them meeting the needed covering number restrictions. Let*

$$\alpha_i = \prod_{j=i+1}^L L_\sigma B_{c2} B_{v2}(1+4B_{QK2})$$

$$\tau_i = \alpha_i^{2/3} + (2\alpha_i L_\sigma B_{c2} B_{v2})^{2/3} + (\alpha_i L_\sigma B_{v2})^{2/3}$$

$$\gamma = C_{B_x}^{1/3}(2L_\sigma B_{c2} B_{v2}\alpha_1 B_w)^{2/3} + C_1^{1/3}\left(1+(B_w L_\sigma B_{v2})^{2/3}\right)$$

$$\eta = C_1^{1/3}\left(B_w^{2/3}\sum_{i=2}^L \tau_i\right)$$

Then, the log covering number of g_{scalar}^{L+1} is

$$\frac{(\gamma+\eta)^3}{\epsilon^2}$$

The proof is left in Appendix Section E for ease of presentation.

Notice this is the covering number of the entire multi-layer Transformer. Thus, we can recover an upper bound for the Rademacher complexity of it by using Dudley's integral.

Substituting our covering number bounds into theorem 4.2 gives us the three corollaries. We state one below and leave the rest to Appendix Section F.

Corollary 4.2.1. *Suppose we have the norm bounds required in lemma 3.6 for each $W_c^{(i)}, W_v^{(i)}, W_{QK}^{(i)}, w$ and let the maximum be B. Let B_x be the input bound. Suppose we also have the bounds needed for theorem 4.2. Let*

$$\alpha_i = \prod_{j=i+1}^{L} L_\sigma B_{c2} B_{v2}(1 + 4B_{QK2})$$

$$\tau_i = \alpha_i^{2/3} + (2\alpha_i L_\sigma B_{c2} B_{v2})^{2/3} + (\alpha_i L_\sigma B_{v2})^{2/3}$$

$$\gamma = \left(B^2 B_x^2 \ln(2dk+1)\right)^{1/3} (2L_\sigma B_{c2} B_{v2} \alpha_1 B_w)^{2/3} + \left(B^2 \ln(2dk+1)\right)^{1/3} \left(1 + (B_w L_\sigma B_{v2})^{2/3}\right)$$

$$\eta = \left(B^2 \ln(2dk+1)\right)^{1/3} \left(B_w^{2/3} \sum_{i=2}^{L} \tau_i\right)$$

Then, the log covering number of g_{scalar}^{L+1} is

$$\frac{(\gamma + \eta)^3}{\epsilon^2}$$

The above covering number is precise, but unwieldy to look at. To get a better sense of it, we can see that, ignoring polylog terms and constants, we get

$$B^2 B_x^2 B_w^2 (L_\sigma B_{c2} B_{v2} B_{QK2})^{O(L)} \frac{1}{\epsilon^2}$$

We do note that the multi-layer method does also work for one layer, however, it is not quite comparable to the bound found in Section 4.1 due to the norm restrictions being different. If we were to look at just the resulting values, the given single layer method essentially trades the cross product terms in the cubed factor for a factor of B_x^2, which seems like an acceptable trade. The proof for the single layer is also much more direct and easy to digest. It also gives a linear dependence on the amount of heads when the multi-layer method extended to multiple heads gives a dependence of $H^{1.5}$.

5 THEORETICAL EXAMPLE: WORD PREDICTION IN NLP

Suppose we have a word embedding set up and a vocabulary of size K. One way to try to learn is by masking a certain percentage of words in a sentence and asking the Transformer to predict these words. Masking is done by taking the row that corresponds to the position of the masked word (let us call this row i) and giving as input a specific vector instead of the actual embedding of the word. Then the prediction is done by taking the vector in row i in the final layer of the Transformer and linearly transforming it into a size K vector. Then we can softmax this vector and use cross entropy loss to train. This is one of the ways BERT (Devlin et al., 2018) is trained. Below, we will suppose only 1 word is masked for each input for ease of presentation. Let us use the cross entropy loss with softmax:

$$\ell_i(y, x) = -\sum_{i=1}^{k} y_i \log(\text{softmax}(x)_i)$$

where $y \in \{0, 1\}^K$ is a one-hot encoded value that specifies the correct word and $x \in \mathbb{R}^K$ is the output of our Transformer at the masked index. Let us call this problem set up Transformer word masking. With this, we state the following theorem:

Theorem 5.1. *The Rademacher complexity of Transformer word masking can be found using Theorem 4.1 or Theorem 4.2, given the required norm assumptions and layer size for the theorems hold.*

Proof. First, we show this loss function is 2-Lipshitz in the ℓ_∞ norm. We prove this in Appendix G.

Therefore, if we let W be our linear map from the Transformer row to the vocabulary scores and let $Y_L^{(i)}$ be the output of our L-layer Transformer on the i^{th} sample. We then have by Foster and Rakhlin (2019):

$$\mathbb{E}\left[\sup_{W,Y} \sum_{i=1}^{m} \epsilon_i \ell_i \left(W(Y_L^{(i)})_\tau\right)\right] \leq$$

$$\tilde{\mathcal{O}}(\sqrt{K}) \max_s \mathbb{E}\left[\sup_{W,Y} \sum_{i=1}^{m} \epsilon_i \left(W(Y_L^{(i)})_\tau\right)_s\right] =$$

$$\tilde{\mathcal{O}}(\sqrt{K}) \max_s \mathbb{E}\left[\sup_{W_s,Y} \sum_{i=1}^{m} \epsilon_i W_s (Y_L^{(i)})_\tau\right]$$

Notice that $W_s^\top \in \mathbb{R}^d$ and then by definition $(Y_L^{(i)})_\tau \in \mathbb{R}^d$ is a token from our Transformer output. Thus, the resulting function class in the Rademacher complexity term is of the same form needed to use Theorem 4.1 if $L = 1$ or Theorem 4.2 for $L \geq 1$. Therefore, we can use theorems to find the Rademacher complexity of Transformer

word masking if the norm restrictions for the theorems hold. □

In order to get the generalization bounds from this Rademacher complexity, we require a bounded loss function. Using clipped cross entropy has been shown to have advantages over unbounded cross entropy (Hurtik et al., 2022; Wei et al., 2023), so using such a loss will allow us to get our generalization bounds from this set up.

6 EMPIRICAL EXAMPLE: SPARSE MAJORITY

The above sections show that, with bounded norms on the weight matrices, our generalization gap should not grow with sequence length. Thus, in this section, we will discuss a simulated study to see empirically if we find results that match our theory. We run a single layer Transformer on a simulated sparse majority data set on a variety of sequence lengths and we look at three results: (1) The total 1-norm of the weights in the Transformer, (2) The cross entropy generalization gap of the best epoch, (3) The validation accuracy of the best epoch for each sequence length.

The first two will show whether or not our theoretical findings are found in practice as well. The last one is more of a practical concern–a situation where the generalization gap is small but the network does not learn is not very useful.

The dataset we create is a sequence of zeros and ones where the label is determined by a majority of a sparse set of the indices. More concretely, if we have a sequence S of length is T, we have a set of indices is I, $|I| < T$, the label is

$$y_i = \mathbb{1}_{\{\sum_{i \in I} S_i > \frac{|I|}{2}\}}$$

In order to more accurately emulate real uses of Transformers, we embed 0 and 1 each into a d-dimensional vector where these two are orthogonal from each other. We also add the positional encoding defined by Vaswani et al. (2017) to add positional information to the sequence.

For our experiment we used a single layer of Tensorflow's `MultiHeadAttention` layer along with a second layer that extracts the $[CLS]$ layer and linearly transforms it into a vector of size 2. The loss we use is the cross entropy loss. Notice how we can use Section 5 to make this fit in the regime we have been discussing in this paper; we can act as if the $[CLS]$ index is always masked and we have a vocabulary of size 2 (0 and 1).

The multihead attention layer has embedding dimension of 64 and 2 heads. The embedding dimension was chosen to be large while still allowing for moderate computation time. Only two heads were also chosen as well for computation time concerns. For our dataset, we had the sparse index set cardinality to 9 and used 300 training samples on sequence length 20, 40, 60, ..., 200 with a validation set size of 10000. The index set cardinality and training set size were chosen after finding a small enough size where the smaller sequence lengths could not always get perfect validation accuracy.

The Transformer trained on a NVIDIA Tesla V100 GPU for 200000 epochs with a batch size of 128. 200000 epochs, while a lot, was needed to allow for the larger sequence lengths to start to overfit. This batch size was also chosen after trial and error.

We trained our model for each sequence length 5 times (new data set each time) and recorded the 1-norms of the weights, the accuracy, and generalization gap of the best epoch. Figure 2 shows the worst generalization gap for each sequence length. Figure 1 shows the largest 1-norm of the weights for each sequence length. Figure 3 shows the best accuracy for each sequence length. We note that each of these do not necessarily represent the same run per sequence length.

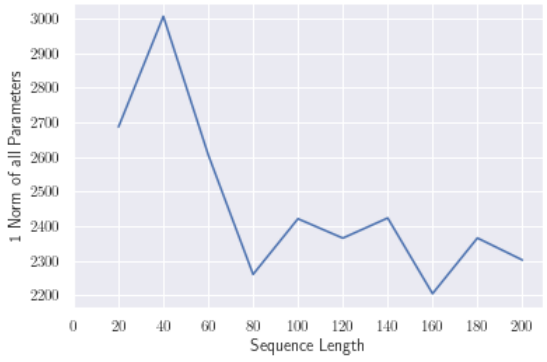

Figure 1: Plot of the max sum of the absolute value of all the weights across sequence lengths. The lack of any increasing trend further validates our assumption of bounded weights being credible.

As we can see, the figure 1 shows that the weights do not increase with sequence length, lending strength to our matrix norm assumptions.

We can also see in figure 2 the generalization gap also

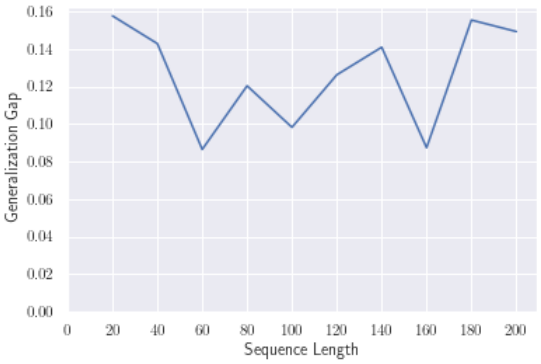

Figure 2: Plot of the max generalization gap across sequence lengths. There is no discernible trend in the plot, giving empirical validation to our theoretical results

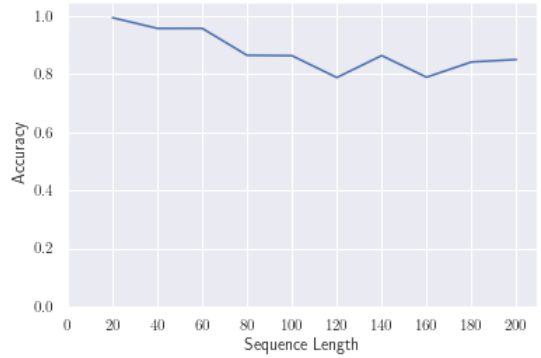

Figure 3: Plot of the maximum accuracy across sequence lengths. While this paper makes no claims on the accuracy, we can see the graph does plateau. Therefore showing our models were learning well even at longer sequence lengths.

has no discernible trend and figure 3 shows the accuracy plateaus as sequence length increases. These results further help validate our theoretical findings and add evidence that longer sequence lengths do not inhibit how well Transformers learn.

The code for these experiments can be found here.

7 CONCLUSION AND FUTURE WORK

In this work, we give norm-based generalization bounds that do not grow with sequence length. This fills a hole in the literature where we can now have sequence length independent generalization bounds with the good properties the norm-based bounds give. We also give empirical evidence to validate our theoretical assumptions and theorems.

Future work could include sharpening the linear covering number bounds and generalizing them for more types of matrix/input norm bound combinations. Another avenue could be analyzing exactly how the norm-based bounds and parameter counting bounds trade off with each other.

Acknowledgements

This research was supported in part through computational resources and services provided by Advanced Research Computing at the University of Michigan, Ann Arbor. We also thank Unique Subedi for noting an error with our Lipshitz constant in Section 5.

References

Ba, J. L., Kiros, J. R., and Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*.

Bartlett, P. L., Foster, D. J., and Telgarsky, M. J. (2017). Spectrally-normalized margin bounds for neural networks. *Advances in neural information processing systems*, 30.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Dudley, R. M. (1967). The sizes of compact subsets of hilbert space and continuity of gaussian processes. *Journal of Functional Analysis*, 1(3):290–330.

Edelman, B. L., Goel, S., Kakade, S., and Zhang, C. (2022). Inductive biases and variable creation in self-attention mechanisms. In *International Conference on Machine Learning*, pages 5793–5831. PMLR.

Foster, D. J. and Rakhlin, A. (2019). l∞ vector contraction for rademacher complexity. *arXiv preprint arXiv:1911.06468*, 6.

Fu, H., Guo, T., Bai, Y., and Mei, S. (2023). What can a single attention layer learn? a study

through the random features lens. *arXiv preprint arXiv:2307.11353*.

Garg, V., Jegelka, S., and Jaakkola, T. (2020). Generalization and representational limits of graph neural networks. In *International Conference on Machine Learning*, pages 3419–3430. PMLR.

Golowich, N., Rakhlin, A., and Shamir, O. (2018). Size-independent sample complexity of neural networks. In *Conference On Learning Theory*, pages 297–299. PMLR.

Hurtik, P., Tomasiello, S., Hula, J., and Hynar, D. (2022). Binary cross-entropy with dynamical clipping. *Neural Computing and Applications*, 34(14):12029–12041.

Kakade, S. M., Sridharan, K., and Tewari, A. (2008). On the complexity of linear prediction: Risk bounds, margin bounds, and regularization. *Advances in neural information processing systems*, 21.

Katz, D. M., Bommarito, M. J., Gao, S., and Arredondo, P. (2023). Gpt-4 passes the bar exam. *Available at SSRN 4389233*.

Kontorovich, A. and Attias, I. (2021). Fat-shattering dimension of k-fold maxima. *arXiv preprint arXiv:2110.04763*.

Ledoux, M. and Talagrand, M. (1991). *Probability in Banach Spaces: isoperimetry and processes*, volume 23. Springer Science & Business Media.

Lin, S. and Zhang, J. (2019). Generalization bounds for convolutional neural networks. *arXiv preprint arXiv:1910.01487*.

Pettersson, J. and Falkman, P. (2023). Comparison of lstm, transformers, and mlp-mixer neural networks for gaze based human intention prediction. *Frontiers in Neurorobotics*, 17:1157957.

Pisier, G. (1981). Remarques sur un résultat non publié de b. maurey. *Séminaire d'Analyse fonctionnelle (dit" Maurey-Schwartz")*, pages 1–12.

Shalev-Shwartz, S. and Ben-David, S. (2014). *Understanding Machine Learning: From Theory to Algorithms*. Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press.

Truong, L. V. (2022). On rademacher complexity-based generalization bounds for deep learning. *arXiv preprint arXiv:2208.04284*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, Q., Li, B., Xiao, T., Zhu, J., Li, C., Wong, D. F., and Chao, L. S. (2019). Learning deep transformer models for machine translation. *arXiv preprint arXiv:1906.01787*.

Wei, C., Chen, Y., and Ma, T. (2022). Statistically meaningful approximation: a case study on approximating turing machines with transformers. *Advances in Neural Information Processing Systems*, 35:12071–12083.

Wei, H., Zhuang, H., Xie, R., Feng, L., Niu, G., An, B., and Li, Y. (2023). Mitigating memorization of noisy labels by clipping the model prediction.

Wu, H., Meng, K., Fan, D., Zhang, Z., and Liu, Q. (2022). Multistep short-term wind speed forecasting using transformer. *Energy*, 261:125231.

Xu, J., Sun, X., Zhang, Z., Zhao, G., and Lin, J. (2019). Understanding and improving layer normalization. *Advances in Neural Information Processing Systems*, 32.

Zhang, T. (2002). Covering number bounds of certain regularized linear function classes. *Journal of Machine Learning Research*, 2(Mar):527–550.

Zhang, Y., Liu, B., Cai, Q., Wang, L., and Wang, Z. (2022). An analysis of attention via the lens of exchangeability and latent variable models. *arXiv preprint arXiv:2212.14852*.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. Not Applicable
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. Not Applicable
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. Not Applicable

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. Yes
   (b) Complete proofs of all theoretical results. Yes, if not in main paper they are in the appendix.
   (c) Clear explanations of any assumptions. Yes

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). Yes
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). Yes
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). Not Applicable. No error bars were used in the figures as the exact numbers themselves are not too important, but the trend is. It was thought error bars would clutter the figures and make the message more convoluted.
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). Yes
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. Not Applicable
   (b) The license information of the assets, if applicable. Not Applicable
   (c) New assets either in the supplemental material or as a URL, if applicable. Not Applicable
   (d) Information about consent from data providers/curators. Not Applicable
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. Not Applicable
5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. Not Applicable
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. Not Applicable
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. Not Applicable

Appendix

A Conversion from Scalar Valued Linear Covering Number Bound to Linear Mapping Covering Number Bound

Suppose we have sets $\mathcal{X}, \mathcal{M} \subset \mathbb{R}^d$, where $\forall w \in \mathcal{M}$, $\|w\|_r \leq B_w$ and $\forall x \in \mathcal{X}$, $\|x\|_s \leq B_x$ for some positive values r and s. Suppose we also have a function class $\mathcal{F} = \{x \to w^\top x \mid w \in \mathcal{W}\}$ and a log covering number C for this function class on inputs $\{x_i\}_{i=1}^n \subset \mathcal{X}$. Let $\mathcal{W} \subset \mathbb{R}^{k \times d}$, where $\forall W \in \mathcal{W}$, $\forall i \in [k]$, $\|W_i\|_r \leq B_w$.

Now, given a $W \in \mathcal{W}$, let us choose

$$\hat{W} = \begin{bmatrix} \hat{W}_1 \\ \hat{W}_2 \\ \vdots \\ \hat{W}_k \end{bmatrix} \in \mathcal{W}$$

where $\hat{W}_j^\top$ is the column vector that would be chosen to cover $W_j^\top$ in the scalar case. Then notice for a positive value q and for any $t \in [n]$:

$$\left\|(W - \hat{W})x_t\right\|_q^q = \sum_{j=1}^{k} ((W_j - \hat{W}_j)x_t)^q \leq k\epsilon^q$$

Thus we can see

$$\log N_\infty \left(\mathcal{F}, k^{1/q}\epsilon, N, \|\cdot\|_q\right) \leq kC$$

Therefore, if C does not rely on n, neither does this bound.

B Proof of Lemma 3.2

Notice how the right hand side is a lower bound for the left hand side. Thus, we need to show $\mathcal{N}_\infty \left(\mathcal{F}, \epsilon, \{B_x e_1, \ldots, B_x e_d\}, \|\cdot\|_q\right) \geq \mathcal{N}_\infty \left(\mathcal{F}, \epsilon, N, \|\cdot\|_q\right)$. To do this, let $\hat{\mathcal{F}}$ be a set of size $\mathcal{N}_\infty \left(\mathcal{F}, \epsilon, \{B_x e_1, \ldots, B_x e_d\}, \|\cdot\|_q\right)$ such that it covers $\mathcal{F}$ on the set $\{B_x e_1, \ldots, B_x e_d\}$ to size ϵ. We claim $\hat{\mathcal{F}}$ also covers $\mathcal{F}$ over any set of size N in our input space. Let $\hat{\mathcal{W}}$ refer to the matrices used in $\hat{\mathcal{F}}$. Let $W \in \mathcal{W}$ and let $\hat{W} \in \hat{\mathcal{W}}$ be the matrix we would choose to cover $\mathcal{W}$. Notice for any $\|x\|_1 \leq B_x$, we have

$$\left\|(W - \hat{W})x\right\|_q = \left\|\sum_{i=1}^{d} (W - \hat{W})x_i e_i\right\|_q \leq \sum_{i=1}^{d} |x_i| \left\|(W_i - \hat{W}_i)\right\|_q \leq B_x \max_{i \in [d]} \left(\left\|(W_i - \hat{W}_i)\right\|_q\right) =$$

$$\max_{i \in [d]} \left(\left\|(W - \hat{W})B_x e_i\right\|_q\right) \leq \epsilon$$

Thus for any set of $\{x_i\}_{i=1}^N$ where $\|x\|_1 \leq B_x$ we have

$$\sup_{j \in [N]} \left\|(W - \hat{W})x_j\right\|_q \leq \sup_{j \in [N]} \max_{i \in [d]} \left(\left\|(W - \hat{W})B_x e_i\right\|_q\right) = \max_{i \in [d]} \left(\left\|(W - \hat{W})B_x e_i\right\|_q\right) \leq \epsilon$$

which shows that it is also an upper bound and thus we have an equality.

C Proof of Single Layer Bound (Theorem 4.1)

Before we start the analysis, for any vectors $v, u \in \mathbb{R}^d$, $\|v\|_1 \leq B_v$, notice the following inequality:

$$v^\top u \leq B_v \max_{j \in [d]} |e_j u| = \max_{j \in [d], s \in \{-1,1\}} s e_j u$$

We will also need this lemma that can be found in Edelman et al. 2022

Lemma C.1. *(Corollary A.7 in Edelman et al. 2022) For $\theta_1, \theta_2 \in \mathbb{R}^p$, we have*

$$\|softmax(\theta_1) - softmax(\theta_2)\|_1 \leq 2 \|\theta_1 - \theta_2\|_\infty$$

Using the fist inequality above we can see we get:

$$\mathbb{E}\left[\sup_{w, W_c, W_v, W_{QK}} \sum_{i=1}^{m} \epsilon_i w^\top W_c^\top \sigma\left(W_v^\top X_{(i)}^\top \text{softmax}\left(X_{(i)} W_{QK} x_{[CLS]}\right)\right)\right] \leq$$
$$B_w \mathbb{E}\left[\sup_{s \in \{\pm 1\}, j \in [d] W_c, W_v, W_{QK}} s \sum_{i=1}^{m} \epsilon_i e_j^\top W_c^\top \sigma\left(W_v^\top X_{(i)}^\top \text{softmax}\left(X_{(i)} W_{QK} x_{[CLS]}\right)\right)\right]$$

But now notice that $e_j^\top W_c^\top$ is also just a row vector. Thus we can use the inequality again to get:

$$B_w B_{W_c} \mathbb{E}\left[\sup_{s \in \{\pm 1\}, j \in [k], W_v, W_{QK}} s \sum_{i=1}^{m} \epsilon_i e_j^\top \sigma\left(W_v^\top X_{(i)}^\top \text{softmax}\left(X_{(i)} W_{QK} x_{[CLS]}\right)\right)\right] \leq$$

Since σ is applied elementwise, we can bring $e_j^\top$ inside of the function. Also, since $\sigma(0) = 0$, we can see that if we have W_v be the zero matrix, we have 0 in our function class. Therefore, we can get rid of the sign function by using the well known property of Rademacher complexities of:

$$Rad_m(\mathcal{F} \cup -\mathcal{F}, S) \leq 2 Rad_m(\mathcal{F}, S)$$

This, along with the contraction inequality Ledoux and Talagrand (1991) allows us to upper bound the above by

$$2 B_w B_{W_c} L_\sigma \mathbb{E}\left[\sup_{j \in [k], W_v, W_{QK}} \sum_{i=1}^{m} \epsilon_i e_j^\top W_v^\top X_{(i)}^\top \text{softmax}\left(X_{(i)} W_{QK} x_{[CLS]}\right)\right]$$

Continuing using the first inequality in this section again, we see

$$2 B_w B_{W_c} L_\sigma \mathbb{E}\left[\sup_{j \in [k], W_v, W_{QK}} \sum_{i=1}^{m} \epsilon_i e_j^\top W_v^\top X_{(i)}^\top \text{softmax}\left(X_{(i)} W_{QK} x_{[CLS]}\right)\right] \leq$$
$$2 B_w B_{W_c} L_\sigma B_{W_v} \mathbb{E}\left[\sup_{s \in \{\pm 1\}, j \in [d], W_{QK}} \sum_{i=1}^{m} s \epsilon_i e_j^\top X_{(i)}^\top \text{softmax}\left(X_{(i)} W_{QK} x_{[CLS]}\right)\right]$$

Let us call the expectation above E. We will now use covering numbers and Dudley's integral to bound E to get our final generalization bound. First, we will use our covering number bound at scale $\epsilon' = \epsilon / 2B_x^2$. Specifically, we will show that a set with a log size of $\ln(2d) + \frac{4B_x^4 C}{\epsilon^2}$ covers the function class in the expectation above. The $\ln(2d)$ comes from the fact that we will have to modify $\hat{\mathcal{W}}$ to have it work for us. We do this by using the covering function class

$$S = \{X \to s e_j^\top X^\top \text{softmax}\left(X \hat{W}_{QK} x_{[CLS]}\right) \mid s \in \{-1, 1\}, j \in [d], \hat{W}_{QK} \in \hat{\mathcal{W}}\}$$

Notice this allows for every $\hat{W}_{QK} \in \hat{\mathcal{W}}$, we have every combination of s and e_j.

Now, using the lemma C.1 and the linear algebra results of $\|Pv\| \leq \|P\|_{2,\infty} \|v\|_1$ and $\|X\|_{2 \to \infty} = \left\|X^\top\right\|_{2,\infty}$,

we get:

$$\begin{aligned}
&\left\| se_j^\top X_{(i)}^\top \text{softmax}\left(X_{(i)} W_{QK} x_{[CLS]}\right) - se_j^\top X_{(i)}^\top \text{softmax}\left(X_{(i)} \hat{W}_{QK} x_{[CLS]}\right)\right\| \leq \\
&\left\| X_{(i)}^\top \text{softmax}\left(X_{(i)} W_{QK} x_{[CLS]}\right) - X_{(i)}^\top \text{softmax}\left(X_{(i)} \hat{W}_{QK} x_{[CLS]}\right)\right\| \leq \\
&\left\| X_{(i)}^\top \right\|_{2,\infty} \left\| \text{softmax}\left(X_{(i)} W_{QK} x_{[CLS]}\right) - \text{softmax}\left(X_{(i)} \hat{W}_{QK} x_{[CLS]}\right)\right\|_1 \leq \\
&2\left\| X_{(i)}^\top \right\|_{2,\infty} \left\| X_{(i)} W_{QK} x_{[CLS]} - X_{(i)} \hat{W}_{QK} x_{[CLS]}\right\|_\infty \leq \\
&2\left\| X_{(i)}^\top \right\|_{2,\infty}^2 \left\| W_{QK} x_{[CLS]} - \hat{W}_{QK} x_{[CLS]}\right\| \leq \\
&2B_X^2 \frac{\epsilon}{2B_X^2} = \epsilon
\end{aligned}$$

Therefore, we can see the log covering number of S is $\ln(2d) + \frac{4B_x^4 C}{\epsilon^2}$. Also, notice, due to the softmax in S, the largest value the function class can be is B_x. Then, using Dudley's integral we have a constant c such that

$$\begin{aligned}
E/c &\leq \inf_{\delta \geq 0} \delta + \int_\delta^{B_x} \sqrt{\frac{\ln(2d)}{m} + \frac{\frac{4B_x^4 C}{\epsilon^2}}{m}} d\epsilon \leq \\
&\inf_{\delta \geq 0} \delta + (B_x - \delta)\sqrt{\frac{\ln(2d)}{m}} + \int_\delta^{B_x} \sqrt{\frac{\frac{4B_x^4 C}{\epsilon^2}}{m}} d\epsilon \leq \\
&\inf_{\delta \geq 0} \delta + (B_x - \delta)\sqrt{\frac{\ln(2d)}{m}} + \frac{2B_x^2\sqrt{C}}{\sqrt{m}} \ln(B_x/\delta)
\end{aligned}$$

When $m > \ln(2d)$ standard analysis can find the minimum for δ is when

$$\delta = \frac{\frac{2B_x^2\sqrt{C}}{\sqrt{m}}}{1 - \sqrt{\frac{\ln(2d)}{m}}} = \frac{2B_x^2\sqrt{C}}{\sqrt{m} - \sqrt{\ln(2d)}}$$

Substituting this in and rearranging we get

$$\begin{aligned}
&\frac{\frac{2B_x^2\sqrt{C}}{\sqrt{m}}}{1 - \sqrt{\frac{\ln(2d)}{m}}}\left(1 - \sqrt{\frac{\ln(2d)}{m}}\right) + B_x\sqrt{\frac{\ln(2d)}{m}} + \frac{2B_x^2\sqrt{C}}{\sqrt{m}} \ln\left(\frac{B_x(\sqrt{m} - \sqrt{\ln(2d)})}{2B_x^2\sqrt{C}}\right) = \\
&\frac{2B_x^2\sqrt{C}}{\sqrt{m}} + B_x\sqrt{\frac{\ln(2d)}{m}} + \frac{2B_x^2\sqrt{C}}{\sqrt{m}} \ln\left(\frac{\sqrt{m} - \sqrt{\ln(2d)}}{2B_x\sqrt{C}}\right)
\end{aligned}$$

Thus, multiplying this by c and substituting this in for E gives us our desired result.

D Corollaries of Theorem 4.1

Corollary D.0.1. *Let us have the requirements needed for Theorem 4.1 along with* $\|x\|_1 \leq B_x$ *and* $\|W_{QK}\|_{1,\infty} \leq B_{W_{QK}}$. *Let*

$$B = B_w B_{W_c} L_\sigma B_{W_v}$$

and let

$$\alpha = 2B_{W_{QK}}\sqrt{d\log(2d+1)}$$

Then we have our Transformer Rademacher complexity being less than

$$O\left(B\left(\frac{B_x^3\alpha}{\sqrt{m}}\left(1 + \ln\left(\frac{\sqrt{m} - \sqrt{\ln(2d)}}{B_x^2\alpha}\right)\right) + B_x\sqrt{\frac{\ln(2d)}{m}}\right)\right)$$

Corollary D.0.2. *Let us have the requirements needed for Theorem 4.1 along with* $\|x\|_1 \leq B_x$ *and* $\|W_{QK}\|_{2,1} \leq B_{W_{QK}}$ *Let*

$$B = B_w B_{W_c} L_\sigma B_{W_v}$$

and let

$$\alpha = 2B_{W_{QK}}\sqrt{2\log(d)}$$

Then we have our Transformer Rademacher complexity being less than

$$\hat{O}\left(B\left(\frac{B_x^3\alpha}{\sqrt{m}}\left(1+\ln\left(\frac{\sqrt{m}-\sqrt{\ln(2d)}}{B_x^2\alpha}\right)\right)+B_x\sqrt{\frac{\ln(2d)}{m}}\right)\right)$$

Where the $\hat{O}$ *denotes normal* O *but with some logarithms not containing* T, d, k *being omitted from inside the formula.*

E Proof of Multiple Layers Covering Number (Theorem 4.2)

We will first start with some useful lemmas stated in Edelman et al. (2022). The proofs of these lemmas will not be reproduced for ease of reading.

Lemma E.1. *(Lemma A.8 in Edelman et al. 2022) For* $\epsilon, C_i, \beta_i \geq 0$, $i \in [n]$ *the solution to*

$$\min_{\epsilon_1,\ldots,\epsilon_n}\sum_{i=1}^n \frac{C_i}{\epsilon_i^2}$$

given

$$\sum_{i=1}^n \beta_i\epsilon_i = \epsilon$$

is $\frac{\gamma^3}{\epsilon^2}$ *where*

$$\gamma = \sum_{i=1}^n C_i^{1/3}\beta_i^{2/3}$$

and

$$\epsilon_i = \frac{\epsilon}{\gamma}\left(\frac{C_i}{\beta_i}\right)^{1/3}$$

The proof is by using Lagrange multipliers.

Lemma E.2. *(Lemma A.15 from Edelman et al. 2022) Suppose* $W^{1:i+1}$, $\hat{W}^{1:i+1}$ *satisfy our norm bounds. Then we have*

$$\begin{aligned}
&\left\|(g_{block}^{(i+1)}(X;W^{1:i+1}) - g_{block}^{(i+1)}(X;\hat{W}^{1:i+1}))^\top\right\|_{2,\infty} \leq \\
&\left\|(W_c^{(i)} - \hat{W}_c^{(i)})^\top\sigma\left(\Pi_{norm}\left(f(g_{block}^{(i)}(X;\hat{W}^{1:i});\hat{W}^{(i)})\right)\right)^\top\right\|_{2,\infty} + \\
&L_\sigma B_{c2}B_{v2}(1+4B_{QK2})\left\|(g_{block}^{(i)}(X;W^{1:i}) - g_{block}^{(i)}(X;\hat{W}^{1:i}))^\top\right\|_{2,\infty} + \\
&2L_\sigma B_{c2}B_{v2}\left\|(W_{QK}^{(i)} - \hat{W}_{QK}^{(i)})^\top g_{block}^{(i)}(X;\hat{W}^{1:i})^\top\right\|_{2,\infty} + \\
&L_\sigma B_{c2}\left\|(W_v^{(i)} - \hat{W}_v^{(i)})^\top g_{block}^{(i)}(X;\hat{W}^{1:i})^\top\right\|_{2,\infty}
\end{aligned}$$

Lemma E.3. *(Lemma A.16 from Edelman et al. 2022) Given* $W^{1:i+1}$, $\hat{W}^{1:i+1}, w, \hat{w}$, *and* $g_{scalar}(X;W^{1:L+1},w) = w^\top g_{block}^{(L+1)}(X;W^{1:L+1})_{[CLS]}$, *we have:*

$$\begin{aligned}
&\left|g_{scalar}(X;W^{1:L+1},w) - g_{scalar}(X;\hat{W}^{1:L+1},\hat{w})\right| \leq \\
&\|w\|\left\|g_{block}^{(L+1)}(X;W^{1:L+1})_{[CLS]} - g_{block}^{(L+1)}(X;\hat{W}^{1:L+1})_{[CLS]}\right\| + \\
&\left\|(w-\hat{w})^\top g_{block}^{(L+1)}(X;\hat{W}^{1:L+1})_{[CLS]}\right\|
\end{aligned}$$

The main proof ideas behind the above two lemmas is to unroll them, then use some norm properties and the triangle inequality to split them up.

Now given these, we will prove the multiple layers covering number theorem. Suppose we have our linear covering bound described in the theorem statement. Let $X_1, \ldots, X_m$ be the inputs that is within our norm bounds. Let $\mathcal{W}_v^{(i)}, \mathcal{W}_c^{(i)}, \mathcal{W}_{QK}^{(i)}$ be the sets of all possible values for $W_v^{(i)}$, $W_c^{(i)}, W_{QK}^{(i)}$ respectively. Let $\hat{\mathcal{W}}_v^{(i)}$ cover the function class $\{x \to W_v^\top x \mid W_v \in \mathcal{W}_v^{(i)}, \|x\|_2 \leq 1\}$, let $\hat{\mathcal{W}}_c^{(i)}$ cover the function class $\{x \to W_c^\top x \mid W_c \in \mathcal{W}_c^{(i)}, \|x\|_2 \leq 1\}$ and let $\hat{\mathcal{W}}_{QK}^{(i)}$ cover the function class $\{x \to W_{QK} x \mid W_{QK} \in \mathcal{W}_{QK}^{(i)}, \|x\|_2 \leq 1\}$ except for $\hat{\mathcal{W}}_{QK}^{(1)}$, which covers the same function class, but with $\|x\| \leq B_x$. Let all of these classes be covered with mT points and let $\epsilon_v^{(i)}$, $\epsilon_c^{(i)}$, $\epsilon_{QK}^{(i)}$ be the resolution for each cover. Also, let $\hat{\mathcal{W}}$ cover $\{x \to w^\top x \mid w \in \mathcal{W}, \|x\| \leq 1\}$ at resolution ϵ_w. The exact value of these resolutions will be shown at the end.

We will show that for any $\epsilon > 0$ and for any $W^{1:L+1}$ that satisfies our norm bounds that there exists a

$$\hat{W}^{1:L+1} \in \hat{\mathcal{W}}_c^{(1)} \otimes \hat{\mathcal{W}}_v^{(1)} \otimes \hat{\mathcal{W}}_{QK}^{(1)} \otimes \cdots \otimes \hat{\mathcal{W}}_c^{(L)} \otimes \hat{\mathcal{W}}_v^{(L)} \otimes \hat{\mathcal{W}}_{QK}^{(L)} \otimes \hat{\mathcal{W}}$$

such that

$$\left| g_{scalar}(X; W^{1:L+1}, w) - g_{scalar}(X; \hat{W}^{1:L+1}, \hat{w}) \right| \leq \epsilon$$

To start, we will use lemma E.3 to get

$$\begin{aligned}
&\left| g_{scalar}(X; W^{1:L+1}, w) - g_{scalar}(X; \hat{W}^{1:L+1}, \hat{w}) \right| \leq \\
&\|w\| \left\| g_{block}^{(L+1)}(X; W^{1:L+1})_{[CLS]} - g_{block}^{(L+1)}(X; \hat{W}^{1:L+1})_{[CLS]} \right\| + \left\| (w - \hat{w})^\top g_{block}^{(L+1)}(X; \hat{W}^{1:L+1})_{[CLS]} \right\| \leq \\
&\|w\| \left\| (g_{block}^{(L+1)}(X; W^{1:L+1}) - g_{block}^{(L+1)}(X; \hat{W}^{1:L+1}))^\top \right\|_{2,\infty} + \epsilon_w
\end{aligned}$$

Now, we use lemma E.2 to see

$$\begin{aligned}
&\left\| (g_{block}^{(L+1)}(X; W^{1:i+1}) - g_{block}^{(L+1)}(X; \hat{W}^{1:i+1}))^\top \right\|_{2,\infty} \leq \\
&\left\| (W_c^{(L)} - \hat{W}_c^{(L)})^\top \sigma \left(\Pi_{norm} \left(f(g_{block}^{(L)}(X; \hat{W}^{1:L}); \hat{W}^{(L)}) \right) \right)^\top \right\|_{2,\infty} + \\
&L_\sigma B_{c2} B_{v2} (1 + 4B_{QK2}) \left\| (g_{block}^{(L)}(X; W^{1:L}) - g_{block}^{(L)}(X; \hat{W}^{1:L}))^\top \right\|_{2,\infty} + \\
&2L_\sigma B_{c2} B_{v2} \left\| (W_{QK}^{(i)} - \hat{W}_{QK}^{(i)})^\top g_{block}^{(i)}(X; \hat{W}^{1:i})^\top \right\|_{2,\infty} + \\
&L_\sigma B_{c2} \left\| (W_v^{(L)} - \hat{W}_v^{(L)})^\top g_{block}^{(L)}(X; \hat{W}^{1:L})^\top \right\|_{2,\infty}
\end{aligned}$$

Notice that if $C^{(i)} \in \mathbb{R}^{d \times T}$, $i \in [m]$, $W \in \mathbb{R}^{k \times d}$

$$\max_{i \in [m]} \left\| (W - \hat{W}) C^{(i)} \right\|_{2,\infty} = \max_{i \in [m], t \in [T]} \left\| (W - \hat{W}) C_t^{(i)} \right\|$$

Therefore we can use our covering number bounds to bound the values in the $\|\cdot\|_{2,\infty}$.

Thus, we get

$$\begin{aligned}
&\left\| (g_{block}^{(L+1)}(X; W^{1:i+1}) - g_{block}^{(L+1)}(X; \hat{W}^{1:i+1}))^\top \right\|_{2,\infty} \leq \\
&\epsilon_c^{(L)} + L_\sigma B_{c2} \epsilon_v^{(L)} + 2L_\sigma B_{c2} B_{v2} \epsilon_{QK}^{(L)} + L_\sigma B_{c2} B_{v2} (1 + 4B_{QK2}) \left\| (g_{block}^{(L)}(X; W^{1:L}) - g_{block}^{(L)}(X; \hat{W}^{1:L}))^\top \right\|_{2,\infty}
\end{aligned}$$

Now note how we can iteratively do this for $\left\| (g_{block}^{(L)}(X; W^{1:L}) - g_{block}^{(L)}(X; \hat{W}^{1:L}))^\top \right\|_{2,\infty}$ until we have gotten to the base case of $g_{block}^{(1)}(X; \hat{W}^{1:1})) = X$. Thus, if we let

$$\alpha_i = \prod_{j=i+1}^{L} L_\sigma B_{c2} B_{v2} (1 + 4B_{QK2})$$

we can see that

$$\max_{i\in m}\left|g_{scalar}(X_i;W^{1:L+1},w)-g_{scalar}(X_i;\hat{W}^{1:L+1},\hat{w})\right|\leq$$
$$\epsilon_w+B_w\left(\sum_{i=2}^{L}\alpha_i(\epsilon_c^{(i)}+L_\sigma B_{c2}\epsilon_v^{(i)}+2L_\sigma B_{c2}B_{v2}\epsilon_{QK}^{(i)})\right)+B_w\alpha_1\left(\epsilon_c^{(1)}+L_\sigma B_{c2}\epsilon_v^{(1)}+2L_\sigma B_{c2}B_{v2}\epsilon_{QK}^{(1)}\right)$$

We left the first layer outside of the sum so we can recall $\epsilon_{QK}^{(1)}$ has a different input bound than the rest. Now, we can use lemma E.1 to get our desired sizes for our different ϵ's and get our covering number stated in the theorem.

F Other Corollaries of Theorem 4.2

Corollary F.0.1. *Suppose we have the norm bounds required in lemma 3.3 for each $W_c^{(i)},W_v^{(i)},W_{QK}^{(i)},w$ and let the maximum be B. Let B_x be the input bound. Suppose we also have the bounds needed for theorem 4.2. Let*

$$\alpha_i=\prod_{j=i+1}^{L}L_\sigma B_{c2}B_{v2}(1+4B_{QK2})$$
$$\tau_i=\alpha_i^{2/3}+(2\alpha_i L_\sigma B_{c2}B_{v2})^{2/3}+(\alpha_i L_\sigma B_{v2})^{2/3}$$
$$\gamma=\left(dB^2B_x^2\ln(2k+1)\right)^{1/3}(2L_\sigma B_{c2}B_{v2}\alpha_1 B_w)^{2/3}+$$
$$\left(dB^2\ln(2k+1)\right)^{1/3}\left(1+(B_wL_\sigma B_{v2})^{2/3}\right)$$
$$\eta=\left(dB^2\ln(2k+1)\right)^{1/3}\left(B_w^{2/3}\sum_{i=2}^{L}\tau_i\right)$$

Then, the log covering number of g_{scalar}^{L+1} is

$$\frac{(\gamma+\eta)^3}{\epsilon^2}$$

Corollary F.0.2. *Suppose we have the norm bounds required in lemma 3.4 for each $W_c^{(i)},W_v^{(i)},W_{QK}^{(i)},w$ and let the maximum be B. Let B_x be the input bound. Suppose we also have the bounds needed for theorem 4.2. Let*

$$\alpha_i=\prod_{j=i+1}^{L}L_\sigma B_{c2}B_{v2}(1+4B_{QK2})$$
$$\tau_i=\alpha_i^{2/3}+(2\alpha_i L_\sigma B_{c2}B_{v2})^{2/3}+(\alpha_i L_\sigma B_{v2})^{2/3}$$
$$\gamma=\left(B^2B_x^2\ln(dk)\right)^{1/3}(2L_\sigma B_{c2}B_{v2}\alpha_1 B_w)^{2/3}+$$
$$\left(B^2\ln(dk)\right)^{1/3}\left(1+(B_wL_\sigma B_{v2})^{2/3}\right)$$
$$\eta=\left(B^2\ln(dk)\right)^{1/3}\left(B_w^{2/3}\sum_{i=2}^{L}\tau_i\right)$$

Then, the log covering number of g_{scalar}^{L+1} is

$$\frac{(\gamma+\eta)^3}{\epsilon^2}$$

G Proof Cross Entropy with Softmax is ℓ_∞ Lipshitz

We want to show that $L(\hat{y},y)=\sum_{i=1}^{K}-y_i\log(\text{softmax}(x)_i)$ satisfies the following:

$$\forall\hat{y}_1,\hat{y}_2\in\mathbb{R}^K\quad|L(\hat{y}_1,y)-L(\hat{y}_2,y)|\leq 2\left\|\hat{y}_1-\hat{y}_2\right\|_\infty$$

where y is a one-hot encoded vector where index i is hot. Then, notice by the mean value theorem and Holder's inequality, we have:

$$\exists x \in \mathbb{R}^K \quad |L(\hat{y}_1, y) - L(\hat{y}_2, y)| \leq \|\nabla L(x, y)(\hat{y}_1 - \hat{y}_2)\|_1 \leq \|\nabla L(x, y)\|_1 \|(\hat{y}_1 - \hat{y}_2)\|_\infty$$

Thus, if we can bound $\|\nabla L(z, y)\|_1$, we are done. Notice we can rewrite our loss as:

$$L(x, y) = \sum_{i=1}^{K} -y_i \log \left(\frac{e^{x_i}}{\sum_{t=1}^{K} e^{x_t}} \right)$$

Only one value in the sum survives since only the i^{th} value in y is set to 1 and the others are set to 0. Thus, for i and for $j \neq i$:

$$\frac{\partial L(x, y)}{\partial x_i} = -\frac{\sum_{t=1}^{K} e^{x_t}}{e^{x_i}} \frac{e^{x_i} \left(\sum_{t=1}^{K} e^{x_t} \right) - e^{x_i} e^{x_i}}{\left(\sum_{t=1}^{K} e^{x_t} \right)^2} = -\frac{\sum_{t \neq i}^{K} e^{x_t}}{\sum_{t=1}^{K} e^{x_t}}$$

$$\frac{\partial L(x, y)}{\partial x_j} = \frac{1}{\sum_{t=1}^{K} e^{x_t}} e^{x_j} = \frac{e^{x_j}}{\sum_{t=1}^{K} e^{x_t}}$$

Thus for any x, we have:

$$\|\nabla L(x, y)\|_1 = \frac{\sum_{j \neq i}^{K} e^{x_j}}{\sum_{j=1}^{K} e^{x_j}} + \sum_{j \neq i} \frac{e^{x_j}}{\sum_{t=1}^{K} e^{x_t}} = \frac{2 \sum_{j \neq i}^{K} e^{x_j}}{\sum_{t=1}^{K} e^{x_t}} \leq 2$$