A Main Theorem Proof

To reduce notation clutter we drop layer index $l$ and re-state the theorem:

**Theorem 3.1.** Let $G(m, y; \theta_d) = \text{Attn}(m, y, m)$, assuming that $\|\partial L/\partial G\| = \Theta(1)$, then $\Delta G \triangleq G(m - \eta \frac{\partial G}{\partial m}, y; \theta_d - \eta \frac{\partial G}{\partial \theta_d}) - G(m, y; \theta_d)$ satisfies $\|\Delta G\| = \Theta(\eta/L_d)$ when:

$$\|v\|^2\|w\|^2 + \|w\|^2\|m_i\|^2 + \|v\|^2\|m_i\|^2 = \Theta(1/L_d)$$

for all $i = 1, \ldots, n$.

**Proof.** Since we are only considering the magnitude of the update, it is sufficiently instructive to study the case where $d = d' = 1$. In this case the projection matrices reduce to scalars $k, q, v, w \in \mathbb{R}$, and $m$ is a $n \times 1$ vector. Recall that for a single query $y$ the attention block is defined as follows:

$$G(m, y; \theta_d) = \text{softmax} \left( \frac{1}{\sqrt{d}} yqk m^T \right) mw$$

Let $s_i = \frac{kwqy}{\sum_{j=1}^n e^{kwqy}}$ and $\delta_{ij} = 0$ if $i = j$ and 0 otherwise, we have:

$$\frac{\partial G}{\partial k} = \frac{1}{\sqrt{d}} vwyq \sum_{i=1}^n m_is_i \left( m_i - \sum_{j=1}^n s_j m_j \right)$$

$$\frac{\partial G}{\partial y} = \frac{1}{\sqrt{d}} vwyk \sum_{i=1}^n m_is_i \left( m_i - \sum_{j=1}^n s_j m_j \right)$$

$$\frac{\partial G}{\partial q} = \frac{1}{\sqrt{d}} vwyk \sum_{i=1}^n m_is_i \left( m_i - \sum_{j=1}^n s_j m_j \right)$$

$$\frac{\partial G}{\partial v} = w \sum_{i=1}^n s_i m_i$$

$$\frac{\partial G}{\partial w} = v \sum_{i=1}^n s_i m_i$$

$$\frac{\partial G}{\partial m_i} = vwqy \sum_{j=1}^n \frac{\partial s_j}{\partial m_i} x_j$$

$$= vwqy \sum_{j=1}^n m_j s_j (\delta_{ij} - s_i) \frac{1}{\sqrt{d}} kqy$$

$$= vwqy + \frac{1}{\sqrt{d}} vwqys_i (m_i - \sum_{j=1}^n m_j s_j)$$

Combining these expressions we get that the total change $\Delta G$ is given by:

$$\Delta G =$$

$$-\eta \frac{\partial L}{\partial G} \left( \frac{v^2 w^2}{d} \left( \sum_{i=1}^n s_i m_i \left( m_i - \sum_{j=1}^n s_j m_j \right) \right)^2 \left( q^2 y^2 + q^2 k^2 + y^2 k^2 \right) + \left( \sum_{i=1}^n s_i m_i \right)^2 \left( w^2 + v^2 \right) \right)$$

$$+ v^2 w^2 \sum_{i=1}^n s_i^2 \left( 1 + \frac{1}{d} k^2 q^2 y^2 (m_i - \sum_{j=1}^n s_j m_j)^2 + \frac{1}{\sqrt{d}} kqy (m_i - \sum_{j=1}^n s_j m_j) \right)$$
By the assumption of the Theorem $\|\eta \frac{\partial L}{\partial \eta}\| = \Theta(\eta)$, so we need to bound the term inside the main parentheses by $\Theta(\frac{1}{L})$. Note that $s_i \geq 0$ and $\sum s_i = 1$, which implies that each summation with $s$ and $m$ is $\Theta(m)$. The desired magnitude $\Theta(\frac{1}{L})$ is smaller than 1 so terms with lower power are leading: $v^2 w^2, w^2 m_1^2, v^2 m_2^2$. The result follows.

## B Derivation of Sufficient Conditions

In Section 3.2 we set the goal to make model update bounded in magnitude independent of model depth:

**GOAL**: $f(x, y; \theta)$ is updated by $\Theta(\eta)$ per optimization step as $\eta \to 0$. That is, $||\Delta f|| = \Theta(\eta)$, where $\Delta f \triangleq f \left( x - \eta \frac{\partial L}{\partial x}, y - \eta \frac{\partial L}{\partial y}; \theta - \eta \frac{\partial L}{\partial \theta} \right) - f(x, y; \theta)$. To achieve this, we study the forward and backward passes. Given the encoder $f_e$ and decoder $f_d$, the Transformer model can be written as $f(x, y; \theta) = f_d(m, y; \theta_d)$ where $m = f_e(x; \theta_e)$ is the memory output of the encoder. The total change after model update is then given by:

$$
\Delta f = \Delta f_d \overset{\text{def}}{=} f_d \left( \tilde{m}, y - \eta \frac{\partial L}{\partial y}; \theta_d - \eta \frac{\partial L}{\partial \theta_d} \right) - f_d(m, y; \theta_d)
$$

where $\tilde{m} = f_e \left( x - \eta \frac{\partial L}{\partial x}; \theta_e - \eta \frac{\partial L}{\partial \theta_e} \right)$ is the updated memory. Analogous to Zhang et al. (2019b), without loss of generality, we make the following assumptions to simplify derivations:

1. All relevant weights are positive with magnitude less than 1.
2. Encoder and decoder have the same number of layers $N$, with $L_e = 2N$ and $L_d = 3N$ blocks in the encoder and decoder respectively.
3. Embedding dimension $d$ is 1 and the size of the input encoder sequence is $n$.
4. Derivative of $f$ with respect to the loss function $\frac{\partial L}{\partial \theta_d}$ is of order $\Theta(1)$

**Forward Pass** The Transformer encoder consists of $L_e$ residual blocks $G_1, \ldots, G_{L_e}$, alternating between self-attention and MLP blocks. Let $x_1 = x$ and $x_{l+1} = x_l + G_l(x_l; \theta_d)$ denote the output of the $l$-th block such that $m = x_{L_e}$. When $l$ is odd, $G_l$ is a self-attention block with parameters $\theta_{el} = \{k_{el}, q_{el}, v_{el}, w_{el}\}$, and when $l$ in even $G_l$ is an MLP with parameters $\theta_{el} = \{v_{el}, w_{el}\}$. We have:

$$
x_{l+1} = x_l + v_{el} w_{el} x_l
$$

$$
x_l \overset{\oplus}{=} x \left( 1 + \sum_{i=1}^{l} v_{el} w_{el} \right)
$$

$$
m \overset{\oplus}{=} x \left( 1 + \sum_{i=1}^{L_e} v_{el} w_{el} \right)
$$

The decoder computation is similar with the addition of encoder-attention blocks:

$$
y_2 = y_1 + G_1(y_1; \theta_{d1})
$$

$$
y_3 = y_2 + G_2(m, y_2; \theta_{d2})
$$

$$
y_4 = y_3 + G_3(y_3; \theta_{d3})
$$

$$
\vdots
$$

$$
f_d(m, y; \theta_d) = y_{L_d} + G_{L_d}(y_{L_d}; \theta_{dL_d})
$$
where $y_1 = y$. When $\ell \neq 0$, $G_l$ is an attention block with parameters $\theta_{dl} = \{k_{dl}, q_{dl}, v_{dl}, w_{dl}\}$. Otherwise, $G_l$ is an MLP with parameters $\theta_{dl} = \{v_{dl}, w_{dl}\}$. We have:

$$
\begin{align*}
  y_2 &= y_1 + v_{dl} w_{dl} y_1 \\
  y_3 &= y_2 + v_{dl} w_{dl} m \\
  y_4 &= y_3 + v_{dl} w_{dl} y_3 \\
  & \vdots \\
  f(m, y; \theta_d) &= y_{L_d} + v_{d_{L_d}} w_{d_{L_d}} x_{L_d}
\end{align*}
$$

from which it follows that $y_l = y \left(1 + \sum_{i=1}^{l} v_{dl} w_{dl}\right) + m \sum_{i=2}^{l} v_{dl} w_{dl}$.

**Backward Pass** With $\theta_E = \{x, \theta_e\}$ and $\theta_D = \{x, \theta_d\}$ denoting full encoder and decoder parameters (including input embeddings), by Taylor expansion we have:

$$
\Delta f = \frac{\partial f}{\partial \theta_D} \Delta \theta_D + \frac{\partial f}{\partial \theta_E} \Delta \theta_E + O \left(||\Delta \theta_D||^2 + ||\Delta \theta_E||^2\right)
$$

$$
\begin{align*}
  &= \frac{\partial f}{\partial \theta_d} \Delta \theta_d + \frac{\partial f}{\partial \theta_e} \Delta \theta_e + \frac{\partial f}{\partial \theta_d} \Delta x + \frac{\partial f}{\partial \theta_e} \Delta y + O \left(\eta^2\right) \\
  &= -\eta \frac{\partial f_e}{\partial \theta_d} \frac{\partial f_d}{\partial \theta_d} T \frac{\partial \mathcal{L}}{\partial f_d} - \eta \frac{\partial f_e}{\partial \theta_e} \frac{\partial f_e}{\partial \theta_e} T \frac{\partial \mathcal{L}}{\partial f_e} - \eta \frac{\partial f_e}{\partial \theta_d} \frac{\partial f_d}{\partial \theta_d} T \frac{\partial \mathcal{L}}{\partial f_d} + O(\eta^2)
\end{align*}
$$

(1)

Note that to reach our goal, it is sufficient for each of the terms to be of order $O(\eta)$. We derive necessary conditions to achieve that by studying each partial derivative in Equation 1 and its contribution to $\Delta f$. By assumption 4 we have that $\frac{\partial \mathcal{L}}{\partial \theta_e} = 1$. From the additive block-based architecture of the encoder:

$$
\begin{align*}
  f_e(x; \theta_e) &= x_1 + G_1(x_1; \theta_{e1}) + G_2(x_2; \theta_{e2}) + \ldots + G_{L_e}(x_{L_e}; \theta_{eL_e})
\end{align*}
$$

we have:

$$
\frac{\partial f_e}{\partial x} = \frac{\partial x_2}{\partial x} + \frac{\partial G_2(x_2; \theta_{e2})}{\partial x_2} + \ldots + \frac{\partial G_{L_e}(x_{L_e}; \theta_{eL_e})}{\partial x_{L_e}} \\
\quad \equiv 1 + \frac{\partial G_1(x_1; \theta_{e1})}{\partial x_1}
$$

so derivative magnitude is independent of the model depth. Following analogous derivation for $\frac{\partial f_e}{\partial \theta_e}$ we get that for each layer $l$:

$$
\begin{align*}
  &\frac{\partial f_e}{\partial \theta_e} = \frac{\partial G_1(x_1; \theta_{e1})}{\partial \theta_{e1}} \\
  &\quad + \frac{\partial G_{l+1}(x_{l+1}; \theta_{e(l+1)})}{\partial x_{l+1}} \frac{\partial G_1(x_1; \theta_{e1})}{\partial \theta_{e1}} \\
  &\quad + \ldots \\
  &\quad + \frac{\partial G_{L_e}(x_{L_e}; \theta_{eL_e})}{\partial x_{L_e}} \frac{\partial G_1(x_1; \theta_{e1})}{\partial \theta_{e1}} \\
  &\equiv \frac{\partial G_1(x_1; \theta_{e1})}{\partial \theta_{e1}}
\end{align*}
$$

And it follows that the magnitude of $\frac{\partial f_e}{\partial \theta_e}$ is bound by:

$$
\frac{\partial f_e}{\partial \theta_e} \equiv \left(\frac{\partial G_1(x_1; \theta_{e1})}{\partial \theta_{e1}}, \frac{\partial G_2(x_2; \theta_{e2})}{\partial \theta_{e2}}, \ldots, \frac{\partial G_{L_e}(x_{L_e}; \theta_{eL_e})}{\partial \theta_{eL_e}}\right)
$$
with the corresponding inner product:

\[
\frac{\partial f_c}{\partial \theta_c} \frac{\partial f_e}{\partial \theta_e}^T = \Theta(1) \sum_{l=1}^{L_c} \frac{\partial G_l(x_l; \theta_{cl})}{\partial \theta_{cl}} \frac{\partial G_l(x_l; \theta_{el})}{\partial \theta_{el}}^T
\]

(2)

Similar analysis for the decoder gives:

\[
\frac{\partial f_d}{\partial \theta_d} \frac{\partial f_d}{\partial \theta_d}^T = \sum_{l=1}^{L_d} \sum_{l' \neq 2}^{L_d} \frac{\partial G_l(y_l; \theta_{dl})}{\partial \theta_{dl}} \frac{\partial G_l(y_l; \theta_{dl})}{\partial \theta_{dl}}^T + \sum_{l=1}^{L_d} \sum_{l' \neq 2}^{L_d} \frac{\partial G_l(m_l; \theta_{dl})}{\partial \theta_{dl}} \frac{\partial G_l(m_l; \theta_{dl})}{\partial \theta_{dl}}^T
\]

(3)

Finally, the order of the term \(\frac{\partial f_c}{\partial \theta_c} \frac{\partial f_c}{\partial \theta_c}^T\) in Equation 1 depends on \(\frac{\partial f_c}{\partial \theta_f} \frac{\partial f_c}{\partial f_c}^T\) and \(\frac{\partial f_c}{\partial \theta_f} \frac{\partial f_c}{\partial \theta_f}^T\). Since encoder and decoder are linked by memory, we have:

\[
\frac{\partial f_d}{\partial \theta_d} \frac{\partial f_d}{\partial \theta_d}^T = \sum_{l=1}^{L_d} \frac{\partial G_l(m_l; \theta_{dl})}{\partial \theta_{dl}} \frac{\partial G_l(m_l; \theta_{dl})}{\partial \theta_{dl}}^T
\]

(4)

Equations 2, 3 and 4 cover all the major terms in the total change \(\Delta f\), so we focus on them to derive the target bound. Expanding the terms in Equation 2 and applying Theorem 3.1 we get the following:

\[
\frac{\partial f_c}{\partial \theta_c} \frac{\partial f_c}{\partial \theta_c}^T = \sum_{l=1}^{L_c} \frac{\partial G_l(x_l; \theta_{cl})}{\partial \theta_{cl}} \frac{\partial G_l(x_l; \theta_{el})}{\partial \theta_{el}}^T + \frac{\partial G_l(x_l; \theta_{cl})}{\partial x_l} \frac{\partial G_l(x_l; \theta_{el})}{\partial x_l}^T
\]

\[
= \sum_{l=1}^{L_c} (v_{el}^2 + w_{el}^2) x_l x_l^T + v_{el}^2 w_{el} 1_{m \times m}
\]

(5)

Similarly, expanding Equation 3 we get:

\[
\frac{\partial f_d}{\partial \theta_d} \frac{\partial f_d}{\partial \theta_d}^T = \sum_{l=1}^{L_d} (v_{dl}^2 + w_{dl}^2) y_l y_l^T + v_{dl}^2 w_{dl} 1_{n \times n} + \sum_{l=1}^{L_d} (v_{dl}^2 + w_{dl}^2) m_l m_l^T + v_{dl}^2 w_{dl} 1_{n \times n}
\]

(6)

And finally for Equation 4 we have:

\[
\frac{\partial f_d}{\partial \theta_f} \frac{\partial f_d}{\partial \theta_f}^T = \sum_{l=1}^{L_d} v_{dl}^2 w_{dl} 1_{n \times n}
\]

(7)

To achieve the target goal it is sufficient to make Equations 5, 6 and 7 of order \(\Theta(1)\). Assuming that all weights are initialized to the same order of magnitude \(v_{cl} = \Theta(v_c), w_{cl} = \Theta(w_c)\) etc., for all \(l\), the sufficient condition for Equation 5 can be derived as follows:

\[
1 \Theta \sum_{l=1}^{L_c} (v_{cl}^2 + w_{cl}^2) \left( 1 + \sum_{i=1}^{l} v_{ci} w_{ci} \right)^2 x^2 + v_{cl}^2 w_{cl}^2
\]

\[
= L_c \left( v_c^2 + w_c^2 \right) \left( 1 + \sum_{i=1}^{l} v_{ci} w_{ci} \right)^2 x^2 + v_c^2 w_c^2
\]

\[
= L_c \left( \|v_c\| \|x\|^2 + \|w_c\| \|x\|^2 + \|v_c\| \|w_c\|^2 \right)
\]

(8)
Similar derivation for Equation 6 gives:

\[
L_d(\|v_d\|^2\|w_d\|^2 + \|v_d\|^2\|y\|^2 + \|w_d\|^2\|y\|^2 \\
+ \|v_d\|^2\|w_d\|^2 + \|v_d\|^2\|m\|^2 + \|w_d\|^2\|m\|^2) \Theta = 1
\] (9)

And for Equation 7 we have:

\[
L_d(\|v_d\|^2\|w_d\|^2) \Theta = 1
\] (10)

C Encoder Initialization

Recall that \(L_e = 2N\) and \(L_d = 3N\), substituting these into gradient expressions for the encoder and decoder we get:

\[
\frac{\partial f_e}{\partial \theta_c} \frac{\partial f_e}{\partial \theta_e} ^T \Theta = 2N((v_e^2 + w_e^2)(1 + 2Nv_e w_e)^2 x x^T + v_e^2 w_e^2 1_{m \times m})
\]

\[
\frac{\partial f_d}{\partial \theta_d} \frac{\partial f_d}{\partial \theta_d} ^T \Theta = 2N ((v_d^2 + w_d^2)yy^T + v_d^2 w_d^2 1_{n \times n}) + N((v_d^2 + w_d^2)m^T m + v_d^2 w_d^2 1_{n \times n})
\]

\[
\frac{\partial f_d}{\partial f_e} \frac{\partial f_d}{\partial f_e} ^T \Theta = 3Nv_d^2 w_d^2 1_{n \times n}
\]

Note that if \(\|v_e\|\|w_e\| < \Theta(1/N)\) then \(\|m\| \Theta = \|x\|\). With this in mind, we let \(\|v_d\| \Theta = \|w_d\| \Theta = \|y\| \Theta = \|x\|\) \(\Theta = (9N)^{-\frac{1}{4}}\), which by design gives:

\[
\frac{\partial f_e}{\partial \theta_c} \frac{\partial f_e}{\partial \theta_e} ^T \Theta = 2N((v_e^2 + w_e^2)(9N)^{-\frac{1}{4}} + v_e^2 w_e^2) 1_{m \times m}
\]

\[
\frac{\partial f_d}{\partial \theta_d} \frac{\partial f_d}{\partial \theta_d} ^T \Theta = 2N ((3(9N)^{-1}) + N(3(9N)^{-1}) 1_{n \times n}
\]

\[
\frac{\partial f_d}{\partial f_e} \frac{\partial f_d}{\partial f_e} ^T \Theta = 3N(9N)^{-1} \Theta = 1_{n \times n}
\]

We then solve for the magnitude of \(v_e\) and \(w_e\) that achieves \(\frac{\partial f_e}{\partial \theta_e} \frac{\partial f_e}{\partial \theta_e} ^T \Theta = 1_{n \times n}\). Assuming that \(\|v_e\| = \|w_e\|\) due to symmetry, we obtain \(\|v_e\| = \|w_e\| = \left(\frac{\sqrt{27}-2}{6}\right)^2 N^{-\frac{1}{4}} \approx 0.67N^{-\frac{1}{4}}\).

D Training Hyper-Parameters

<table>
<thead>
<tr>
<th>Parameters</th>
<th>IWSLT’14 small</th>
<th>WMT’18 base</th>
<th>WMT’17 base</th>
<th>WMT’17 deep</th>
<th>WMT’17 big</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>De-En</td>
<td>Fi-En</td>
<td>En-De</td>
<td>En-De</td>
<td>En-De</td>
</tr>
<tr>
<td>Starting learning rate</td>
<td>0.0005</td>
<td>0.0006</td>
<td>0.0007</td>
<td>0.0004</td>
<td>0.0004</td>
</tr>
<tr>
<td>Decay steps</td>
<td>4000</td>
<td>4000</td>
<td>4000</td>
<td>4000</td>
<td>4000</td>
</tr>
<tr>
<td>Dropout</td>
<td>0.5</td>
<td>0.4</td>
<td>0.2</td>
<td>0.4</td>
<td>0.4</td>
</tr>
<tr>
<td>Batch size (tokens)</td>
<td>4k</td>
<td>80k</td>
<td>25k</td>
<td>25k</td>
<td>25k</td>
</tr>
<tr>
<td>Max updates</td>
<td>300k</td>
<td>90k</td>
<td>1M</td>
<td>500k</td>
<td>500k</td>
</tr>
<tr>
<td>Mixed precision</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 1: Hyper-parameters for T-Fixup models on each dataset.