DeLight: Deep and Light-Weight Transformer

Alessandro Danesi, Constantin Bône, Mohamed Karim Abid
Sorbonne University

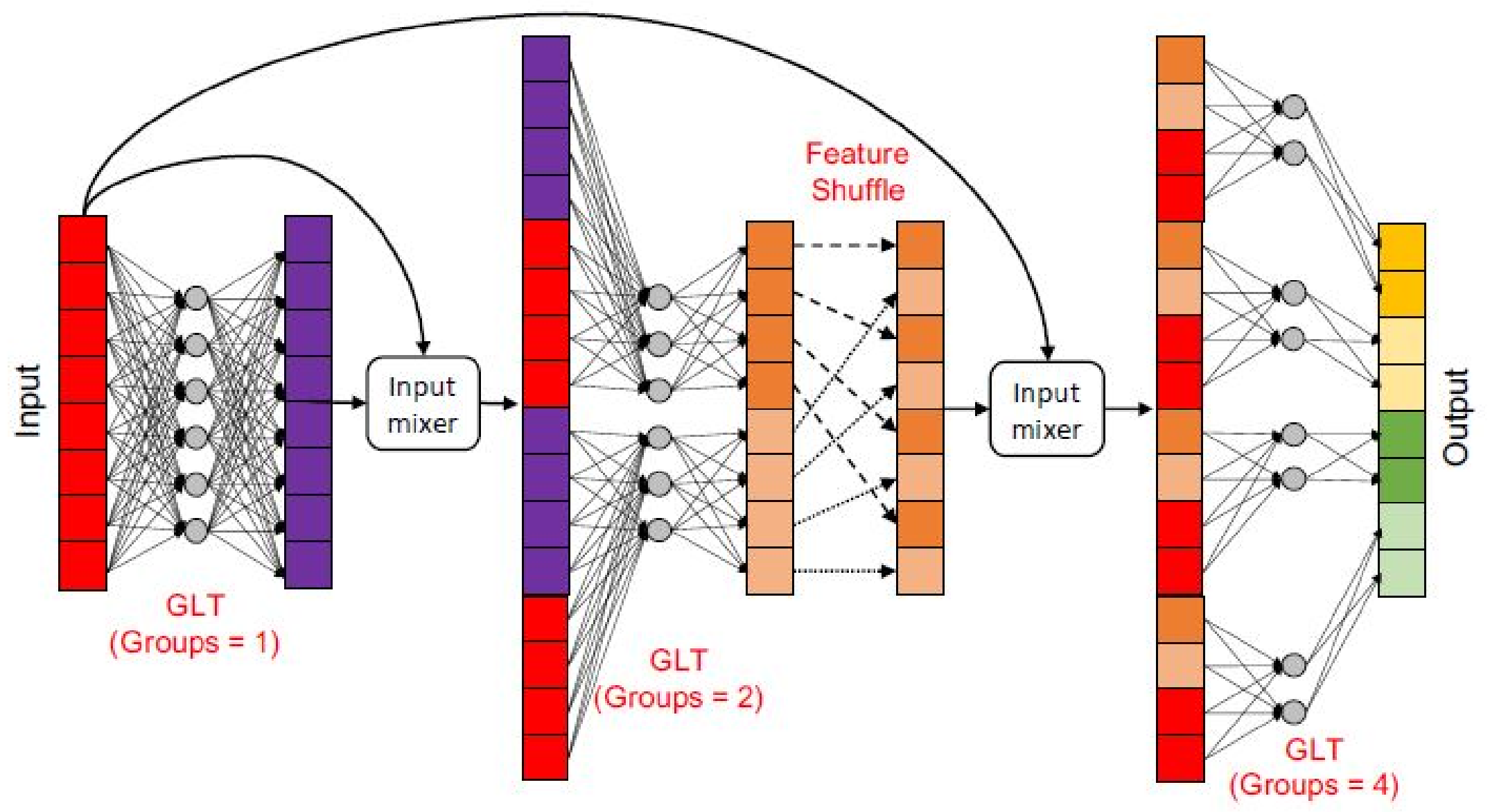
Introduction

The DeLight transformer is a deep and light-weight architecture that extends the classical transformer introduced by Vaswani et al. (2017) achieving better results but with significantly fewer parameters and operations.
The main changes in the DeLight transformer are:

- The DeLight transformation block, that uses group linear transformations (GLTs) that increase depth and width of the network and generate a smaller dimensions dataset for computing attention, requiring fewer operations.
- Replace multi-head attention and feed forward network (FFN) layers with single-head attention and a light-weight FFN.
- Exploit a block-wise scaling among blocks instead of uniform stacking in order to learn representations efficiently.

DeLight Transformation

DeLighT transformation maps a d_m dimensional input vector into a high dimensional space (expansion phase) and then reduces it down to a d_o dimensional output vector (reduction phase) using group linear transformations (GLTs). To learn global representations, the DeLighT transformation shares information between different groups in the group linear transformation using feature shuffling. Moreover, it uses input-mixer connection in order to stabilize the training overcoming the vanishing gradient problem and learn deeper representations.



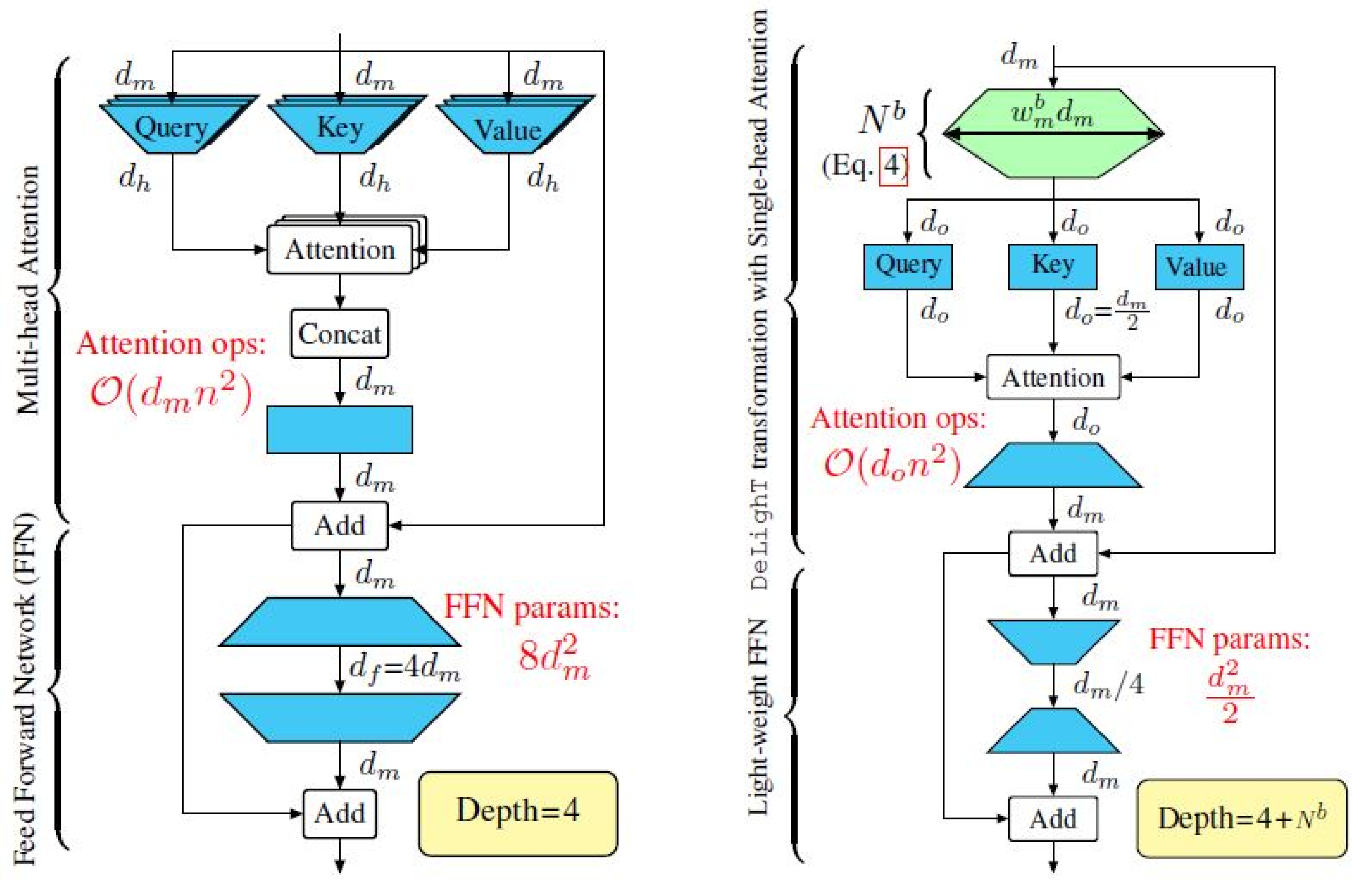
In the expansion phase, the DeLighT transformation projects the d_m-dimensional input to a high-dimensional space, $d_{max} = w_m d_m$, linearly using $\lfloor \frac{N}{2} \rfloor$ layers. In the reduction phase, the DeLighT transformation projects the d_{max}-dimensional vector to a d_o-dimensional space using the remaining $\lfloor \frac{N}{2} \rfloor$ layers. Mathematically, we define the output **Y** at each GLT layer l as:

$$\mathbf{Y}^l = \begin{cases} \mathcal{F}(\mathbf{X}, \mathbf{W}^l, \mathbf{b}^l, g^l), & \text{if } l = 1 \\ \mathcal{F}(\mathcal{H}(\mathbf{X}, \mathbf{Y}^{l-1}), \mathbf{W}^l, \mathbf{b}^l, g^l), & \text{otherwise} \end{cases}$$

where $\mathbf{W}^l = \{\mathbf{W}^l_1, ..., \mathbf{W}^l_{g_l}\}$ and $\mathbf{b}^l = \{\mathbf{b}^l_1, ..., \mathbf{b}^l_{g_l}\}$ are the learnable weights and biases of group linear transformation $\mathcal{F}$ with g^l groups at the *l*-th layer. The $\mathcal{F}$ function splits the input into g^l non-overlapping groups and operates a linear transformation. The function $\mathcal{H}$, instead, first shuffles the output of each group in $\mathbf{Y}^{l-1}$ group and then combines it with the input **X** using the input mixer connection using the input mixer connection.

DeLight Block

We integrate the DeLight transformation into the transformer block to improve it's efficiency.



(a) Transformer block (b) DeLighT block

The transformation takes a d_m-dimensional input to produce a d_o-dimensional output with $d_o < d_m$. This output is then fed to a single head attention followed by a light-weights FFN. The FFN consists in two linear layers. The first one reduces the dimensionality from d_m to d_m/r with r the reduction factor. The second one expands the dimentionality from d_m/r to d_m. The DeLight block is so composed by: (1) a DeLight transformation with N GLTs, (2) three parallel linear layers for key, query and value, (3) a projection layer and (4) two linear layesr in the FNN. Thus the depth of DeLight block is N + 4 compared to 4 of standard transformer block.

Block-wise scaling

We introduce block-wise scaling that creates a network with variably-sized DeLighT blocks, allocating shallower and narrower DeLighT blocks near the input and deeper and wider DeLighT blocks near the output. To do so, we introduce two network-wide configuration parameters: minimum N_{min} and maximum N_{max} number of GLTs in a DeLight transformation. For the *b*-th DeLight block we compute the number of GLTs N^b and the with multiplier w_m^b as:

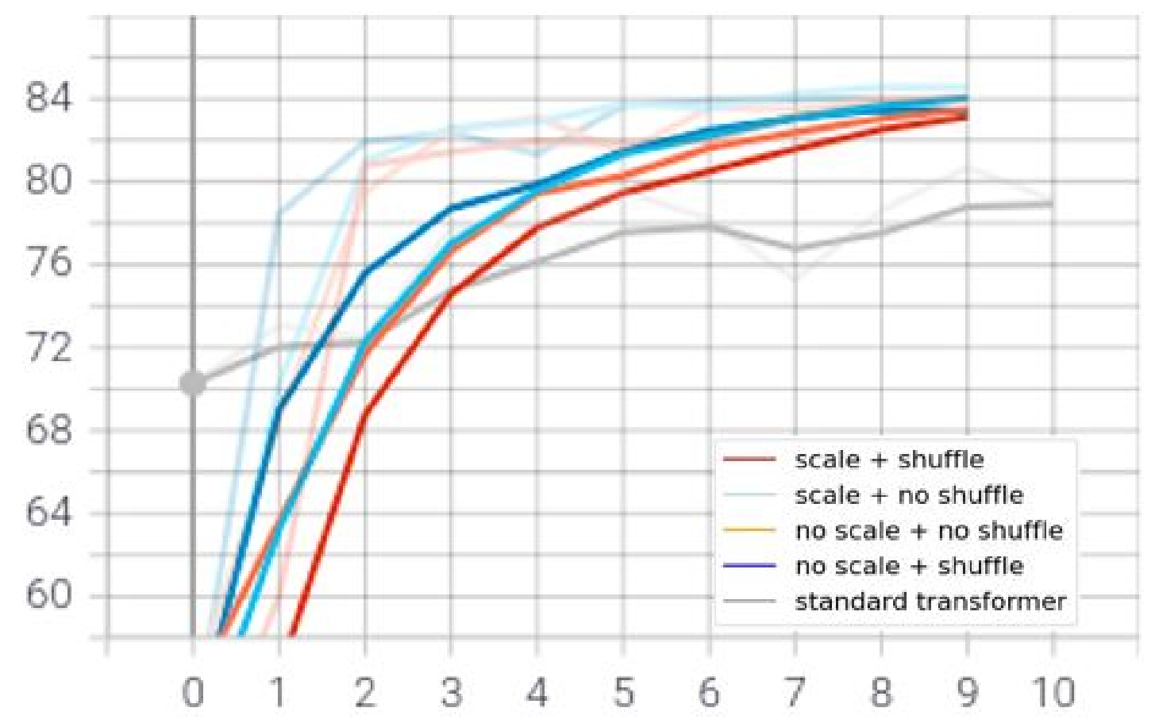
$$N^b = N_{min} + \frac{(N_{max} - N_{min})b}{\mathcal{B} - 1}$$

$$w_m^b = N_{min} + \frac{(N_{max} - N_{min})b}{N_{min}(\mathcal{B} - 1)}$$

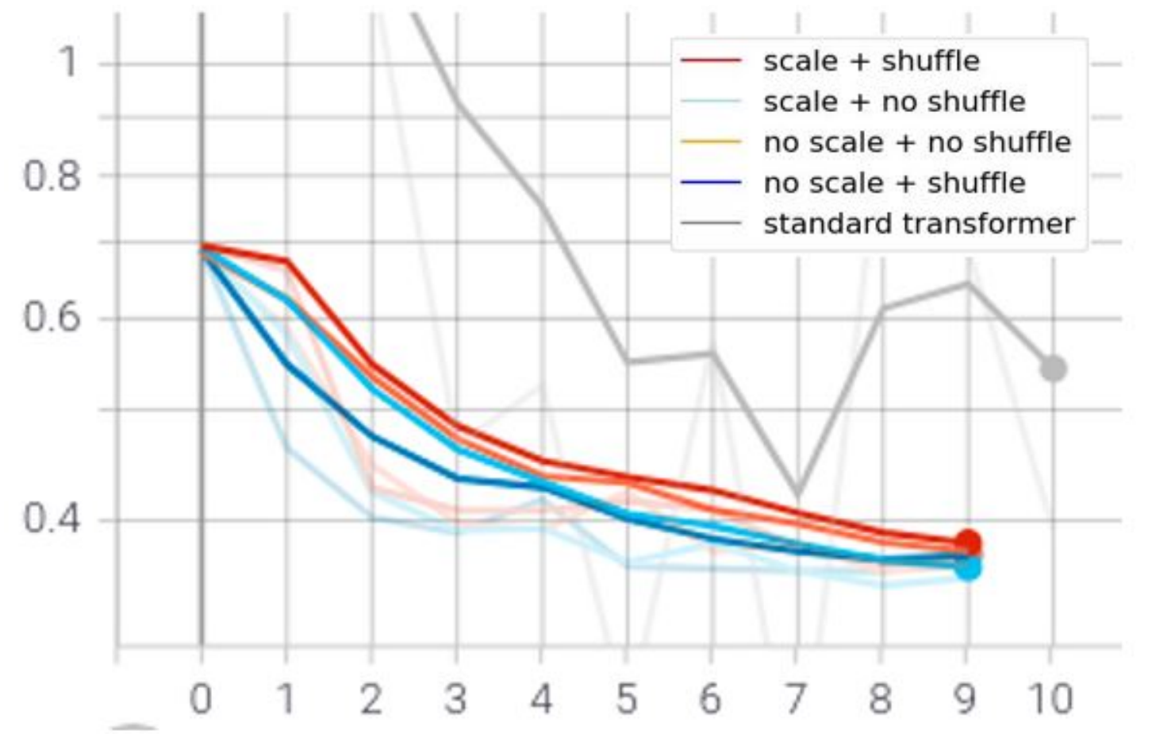
for $0 \leq b \leq \mathcal{B} - 1$, where $\mathcal{B}$ denotes the number of DeLighT blocks in the network.

Datas and results

We implement the DeLight decoder for Sentiment Classification in a similar way to the TP10 network based on self-attention. We use 3 blocks of DeLight Transformer.
Test Accuracy curves for different DeLight models and standard Transformer.



Test Loss curves for the same models.



The table of execution times for different models

Model	Execution time(s)
DeLight Scale+Shift	2220
DeLight Scale	2204
DeLight Shift	2206
DeLight	2198
Transformer	2230

The table with the number of parameters for different models:

Model	Number of parameters
DeLight Scale	268325
DeLight NO Scale	224701
Standard Transformer	222337

Conclusion

This article offers us Delight, a deeper and lighter-weights architecture than state-of-the-art transformer architecture with similar or better performance.

We tested the DeLight model combining GLTs shuffling and block-scaling in all manners for a sentiment classification problem. Comparing accuracies and training losses with respect to standard transformer, we can say that DeLight achieves better results. In particular, shuffling seems giving an improvement to the network.

Scaling doesn't seem significant for this task, but maybe with other applications (language modeling or translations) it could be a great technique to improve the quality of the network.