GALAGAIN Calvin
calvin.galagain@gmail.com

Lab session # 1
ALTEGRAD 2022

11/13/22

1 Question 1

The basic idea of the mechanism of self-attention can be improved by changing the process of representing sentences [1]. Indeed, one idea would be to represent sentences using ”internal attention” which will seek to replace a vector with a matrix of self-attention.

The paper proposes to generate a sequence which concatenates all the words embeddings where there are all independent with each others. To do this, the sentence is divided into a list corresponding to the words, which will make it possible to obtain a matrix after the sentence has been broadcast in a bidirectional LSTM network. The size of this matrix H is of size N*2*U with N the number of words of the sentence and U the hidden dimension of the LSTM.
With an attention matrix A calculated with several attentions, we can obtain the final and new representation of our sentence:

$$M = A \times H \tag{1}$$

The advantage of this new matrix will be to better manage sentences with different semantics but which have similarities in several attentions.
The matrix M will have to be processed to increase the diversity of attention. A penalty is added to the previous model with a Frobenius norm:

$$P = ||(AA^T - I)||_F^2 \tag{2}$$

If attentions are close in A then P will be larger. So this implies that by seeking to have a diversity of our attentions, we will reduce P which will induce a reduction in the redundancies of M.

2 Question 2

The transformer model [3] has brought many changes from the point of view of the architecture of the solutions but also from the point of view of improving the results of the state of the art. Through a combination of attention-based layers, transformers become much better than complex recurrent neural networks.

This difference can be explained in several ways. The first very important point is the complexity of the layers that are in the models. Indeed, with n the size of the sequence and k the dimension of the representation, the complexity of attention is O(d*n^2) against O(n*d^2) for recurrent layers. This explanation makes all the more sense considering that the Google team had as data approximations $n \approx 100$ and $d \approx 1000$. It is much more advantageous to have a square on n than on d. This implies that we can linearly increase the dimension of the representation of sentences without increasing the computation time too much.

Recursive networks are sequential, which prevents parallelization and enormously increases the computation time because the memory is not infinite. Indeed, the recurrent networks calculate layer by layer whereas the transformers have a structure which makes it possible to get around. The sequential complexity of operations goes from O(n) for recurrent networks to O(1) for transformers.

The third point of change will be in relation to the analysis of the length of paths between dependencies. The complexity goes from O(n) to O(1) again. With more attention transformers can handle long sentences better.
The architecture of transformers therefore has a constant amount of computational steps, a constant amount of operations and reduced computational complexity which makes them much better than recurrent networks. Furthermore, there is a gain in explicability which makes it possible to better understand the links between words and sentences.

3 Question 3

Our job was to predict sentiment in site reviews. For each sentence, we tried to say whether it was positive or negative. The weights are therefore trained for this purpose by doing a weighted average in the attention layer.

The majority of words have approximately the same value. The more positive words will have a higher coefficient and the more negative words will approach zero as shown in Fig. 1.

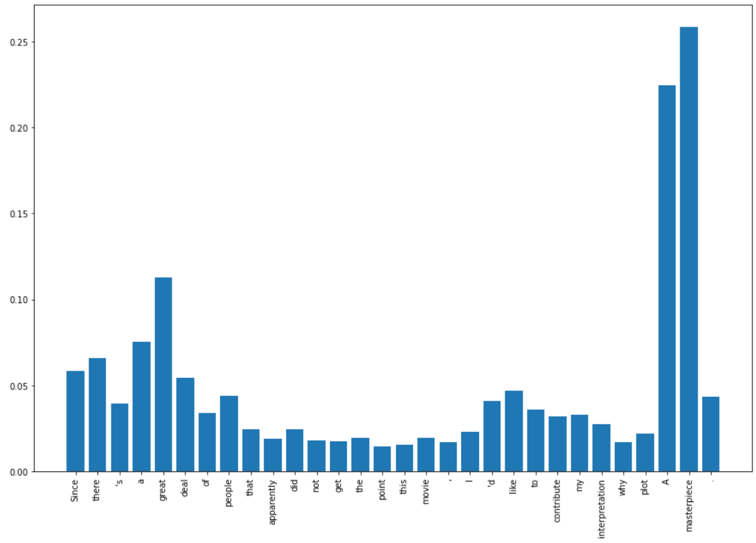

Figure 1: Attention Over Words in Each Sentence.

4 Question 4

The Hierarchical Attention Network (HAN) suffers from a big problem [2]. Indeed, each sentence is encoded without taking into account its neighbors and its representation of the sentences does not allow it to solve problems such as high redundancy. To limit this kind of problem it is possible to use a Context-aware HAN to obtain richer representations. The goal is to change the architecture of HANs to make attentions decisions based on contextual information.

To work, the HAN needs an alignment vector e:

$$e_{it} = u_s^T \tanh(W_s h_{it} + b_s) \tag{3}$$

where $u_s \in \mathbb{R}^{2d_s}$, $W_s \in \mathbb{R}^{2d_s \times 2d_s}$ and $h_{it} \in \mathbb{R}^{T_i \times 2d_s}$ is a sequence of T_i $2d_s$-dimensional of hidden states.

To solve the HAN problem, it is possible to inject a context vector c_i into the self-attention mechanism. Thus, we will be able to change its behavior which did not consider any form of contextual information:

$$e_{it} = u_s^T \tanh(W_s h_{it} + W_c c_i + b_s) \tag{4}$$

The article [2] thus proposes several formulas for c_i. Among them, we can find:

$$\overrightarrow{c_i} = \sum_{i'=1}^{i-1} (s_{i'}) \tag{5}$$

$$\overrightarrow{c_i} = \frac{1}{i-1} \sum_{i'=1}^{i-1} (s_{i'}) \tag{6}$$

$$\overrightarrow{c_i} = \overrightarrow{h_{i-1}} \tag{7}$$

References

[1] Zhouhan Lin, Minwei Feng, Cícero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. A structured self-attentive sentence embedding. *CoRR*, abs/1703.03130, 2017.

[2] Jean-Baptiste Remy, Antoine Jean-Pierre Tixier, and Michalis Vazirgiannis. Bidirectional context-aware hierarchical attention network for document understanding. *CoRR*, abs/1908.06006, 2019.

[3] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.