Zakaria ECHCHAIR
Zakaria.echchair@polytechnique.com

Lab session #1 Hierarchical Attention Network
ALTEGRAD 2022

11/14/22

1 Question 1

when we take $a = softmax(w_{s2} * tanh(W_{s1} * H^T))$ and the softmax() here ensures all the computed weights sum up to 1. Then we sum up the LSTM hidden states H according to the weight provided by a to get a vector representation m of the input sentence.

However this approach is limited, since this vector representation typically focuses on a specific sentence component. because the nature of softmax gives always a high importance to one component and penalize the others.

As a result, it is expected to reflect a semantic aspect or component in a sentence. However, multiple components in a sentence can contribute to the overall semantics of the sentence, especially in long sentences.

So to overcome this problem, we need to have multiple embedding of one sentence that focuses on multiple components. thus, we can make w_{s2} a matrix of size (number of embedding representations that we want * number units) instead of using a vector, which result in having a matrix M instead of vector embedding m such that $M = A * H$.

However this M matrix can suffer from redundancy problems if the attention mechanism always provides similar summation weights for all hops. which pushes us to think consider a penalization approach.

Penalisation

Kullback Leibler has many drawbacks and it isn't fully convenient to this case. indeed we need to optimize the annotation matrix A to have a lot of sufficiently small or even zero values at different softmax output units, and these vast amount of zeros is making the training unstable. with respect to fact that we want each individual row to focus on a single aspect of semantics. In this regard,we can introduce a new penalization term, Which consumes only one-third of the computation when compared to the KL divergence penalization. As a measure of redundancy, we subtract the dot product of A and its transpose from an identity matrix.
$P = \left\| \left(AA^T - I \right) \right\|_F^2$
Here $\| \cdot \|_F$ stands for the Frobenius norm of a matrix. this approach is similar to L2 regularization term. so the purpose here is to minimize P
normally all A row should sum up to 1, thanks to softmax function. we can even consider them as a proba distribution.
For any non-diagonal elements $aij(i \neq j)$ in the AA^T matrix, it corresponds to a summation over elementwise product of two distributions:

$$0 < a_{ij} = \sum_{k=1}^{n} a_k^i a_k^j < 1$$

where a_k^i and a_k^j are the k-th element in the a^{i} and a^j vectors
so when tehre is no overlap between the two distributions. the result would of sum product would be 0 . else it will positive, however in the extreme case when they have almost the same distribution (almost the exact vector), so the sum would be 1. and we are looking to avoid this case of having the same distribution . so We subtract an identity matrix from AA^T so that forces the elements on the diagonal of AA^T to approximate 1 . because to minimize P the optimiser will prefer giving to the diagonal almost one then after substract, it will be 0 .

This technique provide an efficient way to force the M matrix to have different rows and avoid having the redundancy

2 Question 2

- total computational complexity per layer, where the self attention is faster than rnn when the sequence length is smaller than dimension d . which is often the case.

- The amount of computation that can be parallelized with self-attention, as measured by the minimum number of sequential operations required.

- The path length between long-range dependencies in the network. which is a challenge in many sequence tasks. for instance the length of the path forward and backward. indeed the shorter those paths the easier to learn long-range dependencies.

- A self-attention layer connects all positions with a constant number of sequentially executed operations. however RNN requires $\theta(n)$ sequential operation.

- self attention could be restricted to use only r neighborhood in order to improve computational performances.

3 Question 3

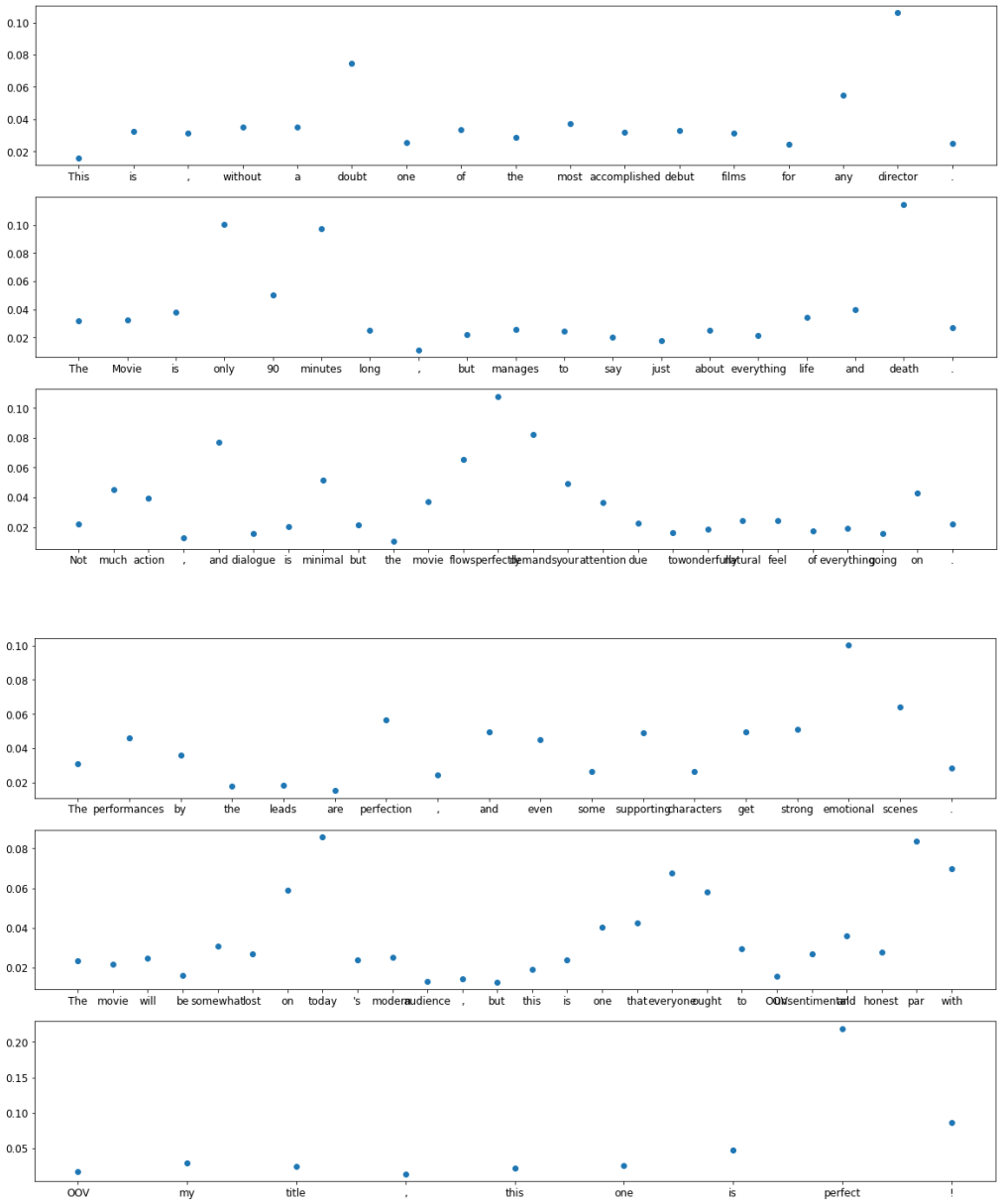

Figure 1: Attention coefficients for a document

4 Question 4

- It cannot model periodic finite-state languages, nor hierarchical structure, unless the number of layers or heads increases with input length.

- Each sentence is encoded in complete isolation,This lack of communication is obviously suboptimal.