Report

Emanuele Alessi 1486470

1. DESCRIPTION

In this homework, I have implemented a word sense disambiguation model using the neural approach. The architecture of the neural network is rapresentate in 3 layers: the first layer contains the embedding layer (I used the Glove embeddings with 100 dimensions); the second layer contains a bidirectional LSTM, and the last layer is the softmax layer. I create a function that parses the semcor traning set and exploits 2 dictionaries: the first one is the lemma dictionary which associates to each lemma a unique id, and the second one is the sense dictionary, this dictionary associates to each lemma/synset a unique id in this way: for each word in the given sentence, if the word has a synset, than will be associated the synset with a unique id, otherwise will be associated the respective lemma with a unique id; the function returns the formatted train set, the sense dictionary and the reversed sense dictionary. The formatted train set is represented as a matrix that contains for each row many pairs (lemma_id, sense_id). In order to generate a good train set, I have implemented also padding technique, in which each sentence will have same dimension (according to the *window_size* parameter), if the sentence is shorter than the window size, many zeros will be added.

2. TEST RESULTS

To train the BLSTM I choose the following parameters:

Batch size	**LSTM hidden size**	**Window size**	**Optimizer**	**Learning rate**
16	100	20	AdaDelta	0.03

After the training phase, I have obtained with 45 epochs the following F1 (Macro) scores for the development set:

Senseval2	**Senseval3**	**SemEval07**	**SemEval13**	**SemEval15**
32,4 %	30,1 %	26%	28,9%	26,4%

3. PREDICTION TECHNIQUE

I created a function that parses the development set and returns a dictionary which groups the development data by the respective *Senseval/SemEval* dataset, in order to calculate for each dataset the F1 score, which is calculated using *scikit-learn* library. The prediction phase is computed as follow: each sentence of the development set will be preprocessed similarly to the training phase and will be given in input to the trained neural network, which returns a matrix of the scores. For each word for which we must predict the meaning, if the word has associated senses *S*, the function returns the sense *s* with the maximum score (with $s \in S$), otherwise the function returns the sense with maximum score on all possible predicted senses.

REFERENCES

Bahdanau D., Cho K., Bengio Y. (2016) – Neural machine translation by jointly learning to align and translate

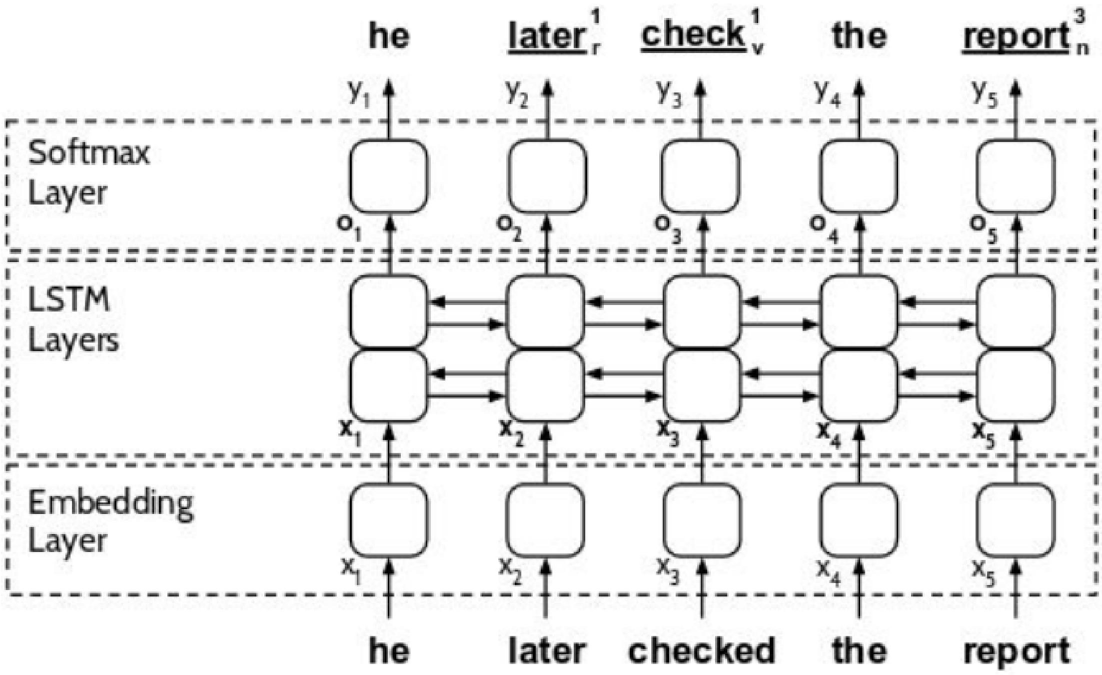
Goldberg Y. (2015) – A primer on neural network models for natural language processing

Wang P., Qian Y., Soong F.K., He L., Zhao H. (2015) – Part-of-speech tagging with bidirectional long short-term memory recurrent neural network

Kageback M., Salomonsson H. (2016) – Word sense disambiguation using a bidirectional LSTM

Raganato A., Delli Bovi C., Navigli R. (2017) – Neural sequence learning models for word sense disambiguation

Pesaranghader A., Pesaranghader A., Matwin S., Sokolova M. (2018) – One single deep bidirectional LSTM network for word sense disambiguation



1 - Neural network structure that it was implemented