Deep Learning for Natural Language Processing

Antonin Berthon

January 2019

1 Multilingual word embeddings

Let $X \in \mathcal{R}^{n*d}$ represent the embeddings of n different words of a language, mapped to a d dimensional latent space, and let $Y \in \mathcal{R}^{n*d}$ represent n words of a different language mapped to an other d dimensional latent space. We want to find a transformation of X that minimizes the distance to Y, that it:

$$W^* = \underset{W \in \mathcal{O}^d}{\arg\min} ||WX - Y||_F$$

To find a closed form solution of this optimization problem, we write: $\forall W \in \mathcal{O}^d$,

$$\begin{aligned} ||WX - Y||_F &= \mathrm{Tr}\left((WX - Y)(WX - Y)^T\right) \\ &= \mathrm{Tr}(WXX^TW^T) - 2Tr(YX^TW^T) + \mathrm{Tr}(YY^T) \\ &= \mathrm{Tr}(XX^T) - 2\,\mathrm{Tr}(YX^TW^T) + \mathrm{Tr}(YY^T) \end{aligned}$$

since $WW^T = I_d$. Therefore the optimization problem is equivalent to:

$$W^* = \underset{W \in \mathcal{O}^d}{\arg\max}\, \mathrm{Tr}(YX^TW^T)$$

With U, Σ and V be the singular value decomposition of $YX^T \in R^d$:

$$\begin{aligned} \mathrm{Tr}(YX^TW^T) &= \mathrm{Tr}(U\Sigma V^TW^T) \\ &= \mathrm{Tr}(\Sigma V^TW^TU) \\ &\leq \mathrm{Tr}(\Sigma) \end{aligned}$$

The last inequality comes from the fact that, since U, V and W are orthogonal matrices, V^TW^TU is an orthogonal matrix, so its columns all have their norms equal to 1, therefore its diagonal terms are all upper bounded by 1.

Since this upper bound is reached when $V^TW^TU = I_d$, we have:

$$\boxed{W^* = UV^T}$$

2 Sentence Classification with Bag-of-Words

In order to approach the Stanford Sentiment Treebank fine-grained sentiment analysis task, we apply a logistic regression on top of a bag-of-words embeddings representation (averaged or weighted averaged). We obtain the following results:

	Train set	Validation set
Average	51.2%	44.5%
Weighted average	46.8%	43.5%

We also try to apply other classifiers on top of the bag-of-words embeddings: a random forest classifier and a MLP. The results are summarized below:

	Train set	Validation set
Random Forest	56.2%	36.2%
MLP	53.2%	42.2%

3 Deep Learning models for classification

We know approach the SST classification task using an encoder and a LSTM recurrent neural network. Since this is a 5-class classification problem, we use the categorical cross-entropy loss, which takes the form:

$$H(p,q) = -\sum_x p(x)\log(q(x))$$

Figure 1 shows the evolution of the train and validation accuracy with the number of epochs. We see that this model quickly begins to overfit the data, and that it does not perform better than the BoV + logistic regression method on the validation set.

Finally, we no longer try to learn the word embeddings in the network, but rather we use the embeddings using from word2vec. The idea is that learning the embeddings involves learning lots of parameters, which increases the risk for over fitting. Moreover, since the latent space should account for relationship between worlds, it is essential to learn it from a really vast corpus. For the rest of the network we keep a simple LSTM recurrent neural network, and we decrease the learning rate progressively during training. Figure 2 shows the evolution of the accuracy on the train and validation sets. This network achieves a validation score of 46.3%.

The results of both neural networks are summarized below:

	Train set	Validation set
Encoder + LSTM	76.0%	37.4%
Word2vec + LSTM	54.0%	46.3%

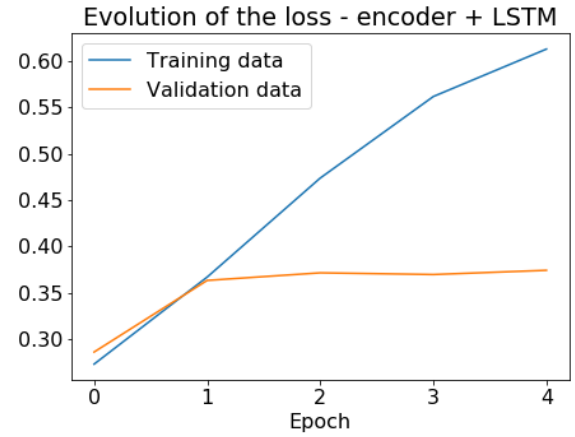

Figure 1: Evolution of the accuracies on training and validation sets with the number of epochs for an encoder+LSTM network.

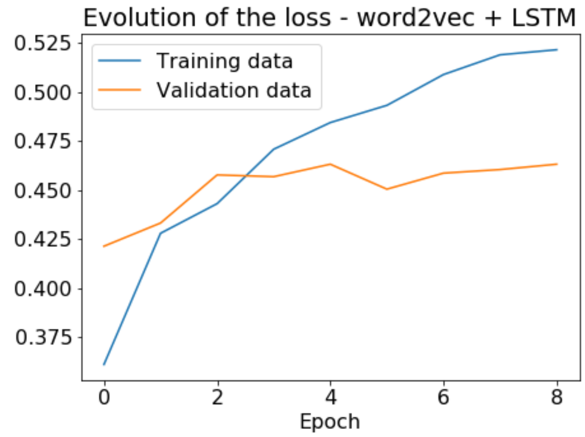

Figure 2: Evolution of the accuracy on training and validation sets with the number of epochs for a W2V+LSTM network.