Advanced NLP

Assignment 2

Tathagato Roy (2019111020)

1 Theory

1.1 How does ELMo differ from CoVe ? Discuss and differentiate both the strategies used to obtain contextualized representations with equations and illustrations as necessary ?

There are various differences and similarities between ELMo and CoVe.Both try to learn contextualized word embeddings although in different ways. The primary difference is between how they are trained. ELMo is trained on the self-supervised Bi Directional Language Modelling Task. A multilayer BiLSTM is used to train for this objective after which the outputs of the BiLSTM is combined with a dense layer to get predicted vocabulary distribution. The embeddings are obtained by getting a weighted sum of the various layers of BiLSTM hidden states. Whereas CoVe is trained in a supervised fashion in the Language Translation task (Specifically English - German translation). a Neural Machine Translation (NMT) is trained with GlOVe embeddings as input for the BiLSTM. The final embeddings are hidden states of the Encoder of the NMT along with the GloVE embeddings.

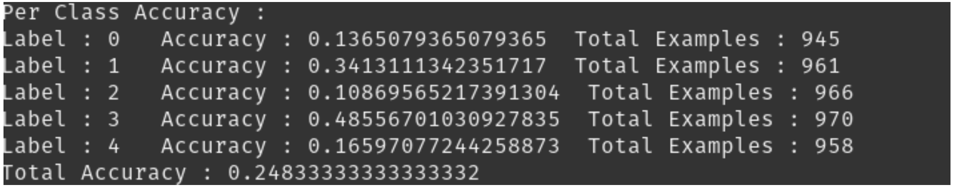
1.2 The architecture described in the ELMo paper includes a character convolutional layer at its base.Find out more on this and describe this layer.Why is it used ? Is there any alternative to it ?

Character Convolution layer used to generate non-contextual word embeddings of each word. By using Multiple Kernels the character Convolution layers learns to represent each letter in a vector which are then concatenated to form a non-contextual word embedding which is then fed as the input the main module of the ELMo.This benefits that it allows multiple kernel to learn multiple sense of the word while also taking into account subword information.It also can seamlessly(naturally) handle unseen / out of vocabulary words.

Other pretrained embeddings like Word2Vec or FastText can also be used instead of the Character Convolution layer. Word2Vec can't handle unseen words whereas hence need to be handled with tag ¡UNK¿ but FastText can.

2 Analysis

All the relevant Hyperparameters can be found in the config.py file in the src folder.
Loss = CrossEntropy Loss
Epochs : 15
Batch$_{S}ize : 100$
$FastTextEmbeddingDimension = 100$
$HiddenStateDimension = 100$
$SequenceLengthforBiDirectionalLanguageModelling = 75$
$SequenceLengthforDownstreamTask = 500$
$Optimizer = Adam$
$Learningrate = 0.001$



```
Per Class Accuracy :
Label : 0   Accuracy : 0.1365079365079365  Total Examples : 945
Label : 1   Accuracy : 0.3413111342351717  Total Examples : 961
Label : 2   Accuracy : 0.10869565217391304  Total Examples : 966
Label : 3   Accuracy : 0.48556701030927835  Total Examples : 970
Label : 4   Accuracy : 0.16597077244258873  Total Examples : 958
Total Accuracy : 0.24833333333333332
```