**Training Methodology**

ML Intern Assignment @Enterpret

## **Assignment 1**

**Problem Statement (ABSA - Aspect Based Sentiment Analysis)**

* Given a text and a *"phrase"* from it, detect the sentiment expressed towards the *"phrase"* in the text instance.

**Literature Survey :**

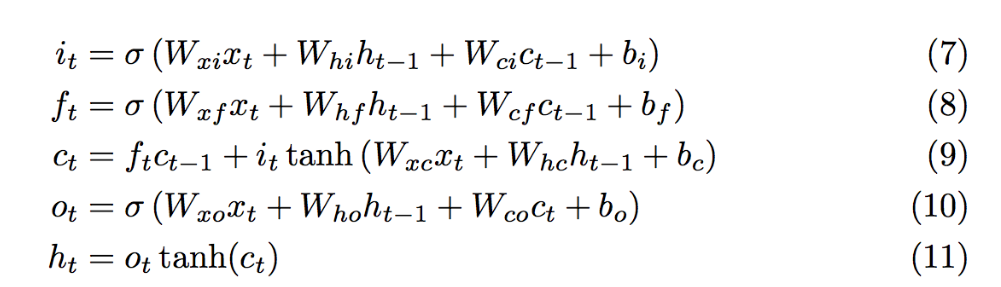
* The Aspect Based Sentiment Analysis is a classification process of phrases based on the aspect.
* The ABSA can also be termed as the next step of text classification. In text , different context words might have different sentiments which cannot be classified just based on text classification.

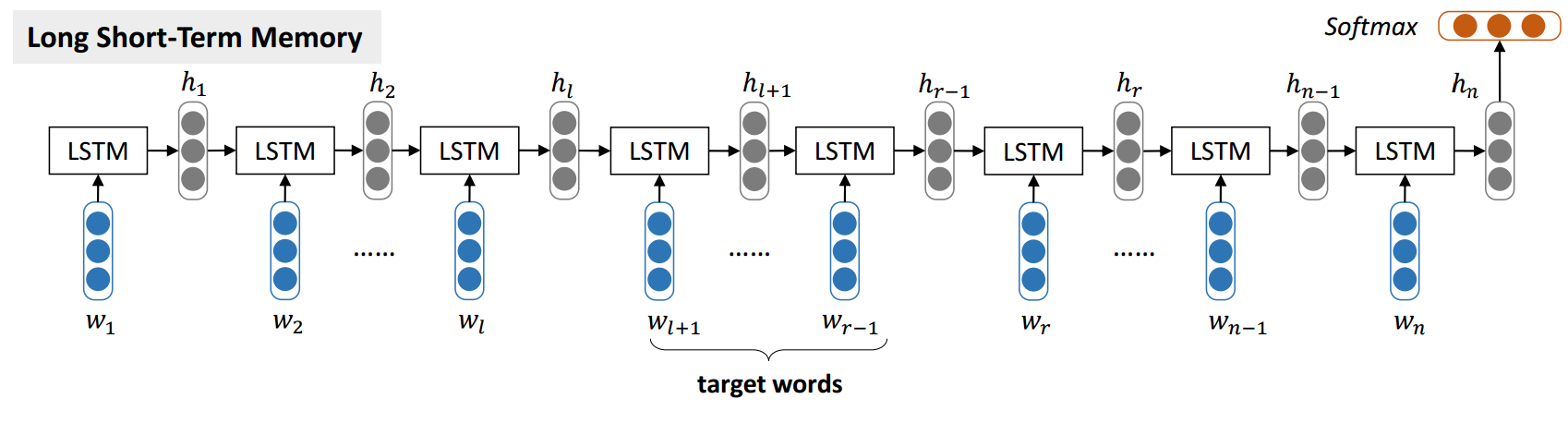
For example, let us consider the sentence: “I bought a new camera. The picture quality is amazing but the battery life is too short”. If the target string is picture quality, the expected sentiment polarity is “positive” as the sentence expresses a positive opinion towards picture quality. If we consider the target as battery life, the correct sentiment polarity should be “negative”.

* Effective LSTMs for Target-Dependent Sentiment Classification is the paper referred for the project. The paper is about ABSA with the LSTM model. The Target connection and Target dependent are the two methods explained in the paper, which surpass the original LSTM model.
* For the project we have used the Target Connection ABSA method.

LSTM Model :

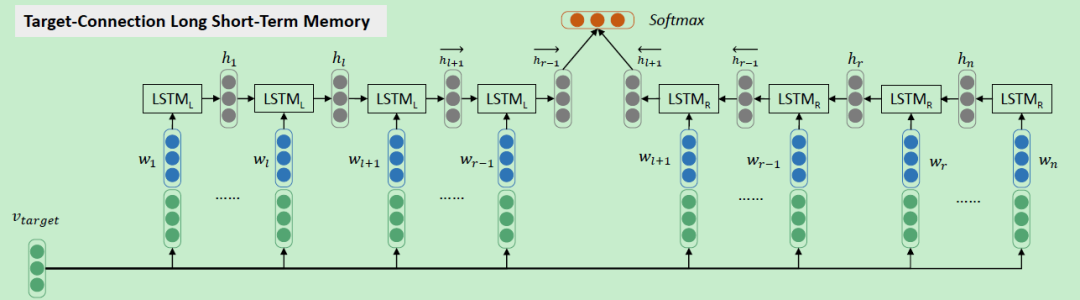
* The basic LSTM Model is shown in the picture below.for w word of length n. {w1,w2,..,wl} is preceding word and {wr, wr+1,..., wn} and {wl+1,...,wr-l} ar target.
* In LSTM each word is represented as low-dimension, continuous, and real-valued vectors as word embedding.
* All the word vectors are stacked in embedding matrix Lw ∈ R d×|V | , where d is the dimension of word vector and |V | is vocabulary size.





* The final layer consist of Sigmoid function t classify probability as negative, neutral and positive.

**Target-Connection LSTM (TC-LSTM**) :

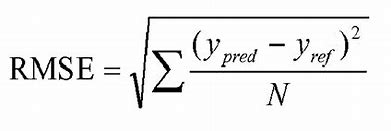


* As mentioned above the issue related to LSTM not considering the aspect word to analyze the Sentiments, Target Connections LSTM (TC-LSTM) provides the approach to deal with it.
* TC-LSTM goes one first head by concatenating the v\_target with word embedding. The v\_target can be obtained by averaging the words it contains.
* There inputs of TC-LSTM is a sentence consisting of n words {w1, w2, ...wn} and a target string t occurs in the sentence, where target t represents as {wl+1, wl+2...wr−1}.
* We split the text in 3 parts, first part to left lstm middle to target and remaining to right lstm.
* For embedding we used the pretrained glove model with 200 embedding.

Approaches:

* As stated above, I moved forward with TC-LSTM methods.
* Firstly we preprocessed the data as required. For TC-LSTM we are required to split text in 3 parts i.e left indices aspect , right indices.Next we feed these three inputs to our model and labels as target.
* Unlike the basic lstm we have two lstm in this method: left lstm and right lstm.
* The model is trained for 1000 epochs with accuracy of 70 and train loss 5.58e-03. The model is still learning as we increase the epochs.

Metrics:



The metrics calculation was done by RMSE.

The loss function is the cross-entropy error of sentiment classification and the optimizer used is Adam.

Ablation Table:

Method Accuracy Macro-F1

LSTM 0.665 0.647

TD-LSTM 0.708 0.690

TC-LSTM 0.715 0.695

The TD-LSTM and TC-LSTM have a bigger improvement in accuracy as compared to LSTM. TC-LSTM out performance both the models in accuracy and F1 score as per paper. Our Model produced an accuracy of 70%.

Conclusion :

* The Target Connection LSTM for Aspect Based Sentiment Analysis produces great results on test data as well. The final test\_data.csv is present in results.
* The predictions were really great with consideration of aspects.
* The task was quite challenging as the only reference available was the paper and paper repo.

Thank You

References -

* [\*1512.01100.pdf (arxiv.org)](https://arxiv.org/pdf/1512.01100.pdf)
* [songyouwei/ABSA-PyTorch: Aspect Based Sentiment Analysis, PyTorch Implementations. 基于方面的情感分析，使用PyTorch实现。 (github.com)](https://github.com/songyouwei/ABSA-PyTorch)