Deep Learning for Sentence Correction

Lakshay, Sahaja Mahanty, Mehta Karan Hemendra

[a]Under guidance of Prof. Rajeshkannan R
[b]Vellore Institute Of Technology

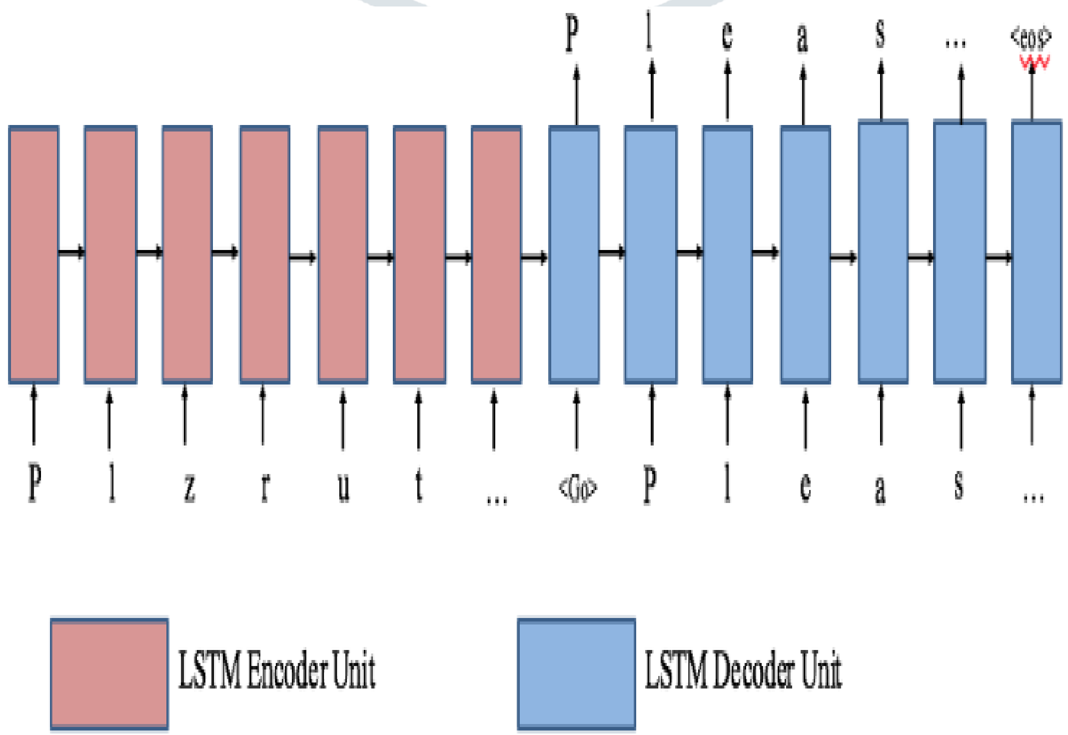
Abstract

One of the most important issue in assisting computers to learn languages is correction of sentences. Some errors in sentences that occurs frequently can not be tackled by machine translation based on stastics or defining grammar rules . So, need of hour is to correct the sentences using RNNs and NLP while preserving sentiments, named entities etc. We will be using Recurrent Neural Networks (RNNs), because they have the capability of dealing with capable of dealing with sequential data. Till now there are some systems available for sentence correction but the problem is that they dont have long range dependencies
.i.e. they just see the previous word and correct the next word like the most common word similar to previous word but with RNN we can have long range dependencies i.e. we can check what word will come next given a set of words.

Keywords: NLP - Natural Language Programming, RNN - Recurrent Neural Network

1. Introduction

Recurrent Neural Networks or simply RNNs are neural networks which has the special capability of dealing with sequential data. Another special thing about RNNs is that have some kind of inner structure which acts like a memory structure for the RNN. This memory structure is absent in other kinds of neural networks like CNN, MLP etc. But what really is the use of this memory structure. Memory structure is helpful for preserving the latent prop- erties of the sentence. Given a sequence of symbols the RNN scans the sequence multiple times (the number of time a sequence is scanned depends on the number of symbols in the sequence). After each scan the RNN learns more about the sequence. We will be construct- ing a RNN that will can sentences as sequence of words. Hence, the number of scans will depend on the number of words in the sentence. For example, in a sentence of 7 words will be scanned 7 times an after each scan the neural network is able to get a better knowledge about the essence of the sentence and each time the essence is stored in the memory state.



There are different versions of RNNs available. These versions differ on the basis of the structure of the memory structure. Some of the versions of RNNs are vanilla RNN, GRU RNNs and LSTM RNNs. We will be using LSTM RNNs as they are capable of storing detailed relevant information in their memory systems and carefully exposing information at each step.

2. Literature Survey

While reaseaching on various sentence correctors, we came accross various models that were used to represent the languages like rule-based approaches in which grammatical knowl- edge is encoded or , stochastic models that learns from data. To capture dynamics of lan- gauges, generally Hidden Markov Model are used which provide good results. But problem with Hidden Markov Model is that they assume that there is no dependency between current word and previos words, so HMM's cannot work with long range dependencies.

Deep Learning can easily work with these long range dependencies, so we can use neural networks with Natural Language Processing to make robust models.

Review of Luong et al. (2015) uses attention based neural networks to build a similar system. Model was made to translating a source sentence, x1, x2, ..., xn, to a target sen- tence, y1, y2, ..., yn and calculating its conditional probability .So it can be divided into two parts
(a) Represent the source sentence using an encoder and
(b) Predict the target sentence using a decoder and finding their respective conditional prob- ability.

This is the system we chose to utilize for our task. We found that almost all recent related work such as (Kalchbrenner and Blunsom, 2013; Sutskever et al., 2014; Cho et al., 2014 Bahdanau et al., 2015 ; Luong et al., 2015; Jean et al., 2015) have the same NMT system.

Luong et. al 2015 approach's was very similar to our approach, but their aim was to develop the model for language translation and our aim is to correct sentences at charachter level. Their model had 4 layers, 1000 cells each , and embeddings of 1000 dimensions. As model was very complex and took lot of time to train so we could only try it with 3 layers without any parameter tuning.It predicts a target word based on all the previously generated target words and the context vectors associated with these source positions .

To handle long sentences, instead of representing the source sentence into fixed-length vector, the source sentence is represented by a number of vectors and while decoding the translation, subset of such vectors is used. Thus long range dependencies can be handled.

3. Methodology

3.1. Our Approach

To achieve sentence correction, we need to have a sufficiently large knowledge base first so that the output of the correction algorithm can produce semantically sound and appro- priate grammatically correct sentences. This has been done by the training of the dataset by providing input and output of the sentences at the same time through the encoder and the decoder.

The four layers used in the system contain 3 forgetting layers and once acceptance layer(tanh) which are the basis of the whole algorithm to work upon.

3.2. Algorithm

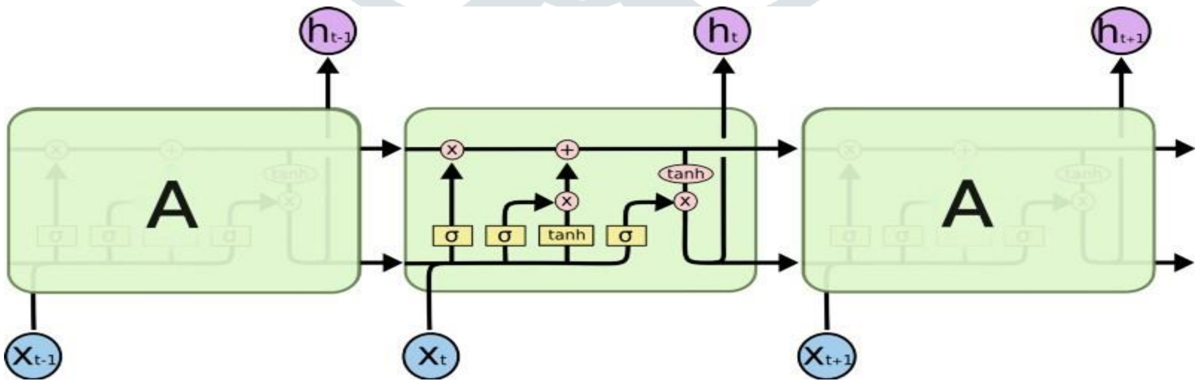
The correction model comprises of two RNNs - an encoder and a decoder. The encoder reads the information arrangement, word by word and emits a context (a component of last shrouded condition of encoder), which would in a perfect world catch the essence (semantic outline) of the input sequence. In view of this input sequence, the decoder produces the yield succession, single word at once while tlooking at the unique situation and the past word amid each timestep. This is an absurd misrepresentation, yet it gives you a thought of what occurs in the technique that we have made.

The setting can be given as the underlying condition of the decoder RNN or it can be associated with the concealed units at each time step. Presently our goal is to together augment the log likelihood of the output sequence based off of the input sequence.

3.2.1. Training Algorithm

```
function SENTENCE CORRECTION (input, output){ input <-
    intoEncoder()
    output <- outtoDecoder() foreach word in
    input:
        pass through layer[i]//variesfrom1to4
            iflayer[i] == forget {//3forgettinglayersand1acceptance checkforwords to
                be discarded
            }
            elseiflayer[i] == acceptance {
                checkforwords to be accepted in the correct sequence
            }
    match the output of the encoder to the output from the training set  SENTENCE
    CORRECTION(input, generatedOutput)
}
```

Our actual models differ from the above description in three important ways. First, we used two different LSTMs: one for the input sequence and another for the output sequence, because doing so increases the number model parameters at negligible computational cost and makes it natural to train the LSTM on multiple language pairs simultaneously [18]. Second, we found that deep LSTMs significantly outperformed shallow LSTMs, so we chose an LSTM with four layers. Third, we found it extremely valuable to reverse the order of the words of the input sentence. We found this simple data transformation to greatly improve the performance of the LSTM.



4. Expected Results

4.1. Data Set

We took National University of Singapore (NUS) NLP short messages dataset. This dataset contains 2000 short messages. It contains incorrect sentences and its corresponding correct sentence.

Incorrect : U wan me to ”chop” seat 4 u nt?

Correct: Do you want me to reserve seat for you or not?.

Incorrect : Yup. U reaching. We order some durian pastry already. U come quick. Correct:Yeap. You reaching? We ordered some Durian pastry already. You come quick.

Incorrect :They become more ex oredi... Mine is like 25... So horrible n they did less things than last time...
Correct:They become more expensive already. Mine is like 25. So horrible and they did less things than I did last time.

Incorrect :I'm thai. what do u do? Correct:I'm Thai. What do you do?

Incorrect :Hi! How did your week go? Haven heard from you for some time... Hows everything? Correct:Hi! How did your week go? Haven't heard from you for some time. How's everything?

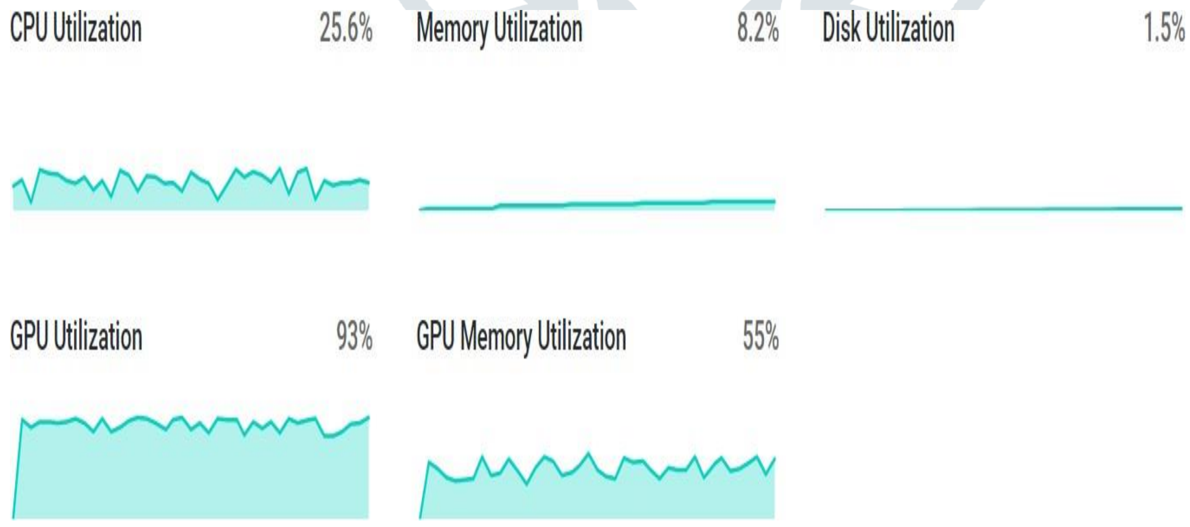
Incorrect :Haha... Okay... You going to mail her? Or you want me to reply... Correct:Haha. Okay. Are you going to mail her? Or do you want me to reply?

Incorrect :Look for it on glass table in front of tv Correct:Look for it on the glass table in front of TV.

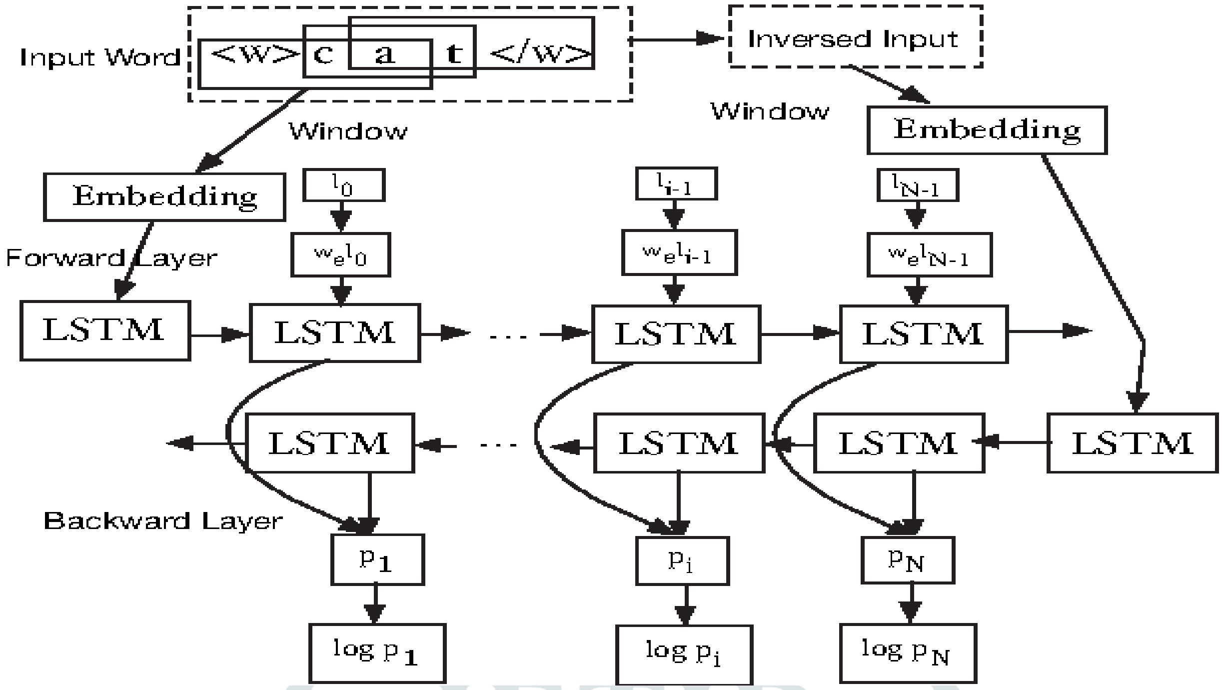
Incorrect :Nah im goin 2 the wrks with j wot bout u? Correct:No, I'm going to the WRKS with J. What about you?

Incorrect :Lea so wanna exchange hp number? Correct:Lea, so you want to exchange handphone number?

System Metrics
System utilization while training:



This shows the Utilization of the resources by the learning code. It includes the CPU, Memory, Disk, GPU and GPU Memory Utilization.



This figure shows the basic working of the architecture of the Sentence Corrector, where the input word is passed into the window, then into the multiple LSTMs, where it is processed through the forward and backward layers

5. Conclusion

In this paper, we proposed a new neural network architecture, called an RNN EncoderDecoder that is able to learn the mapping from a sequence of an arbitrary length to another sequence, possibly from a different set, of an arbitrary length. The proposed RNN EncoderDecoder is able to either score a pair of sequences (in terms of a conditional probability) or generate a target sequence given a source sequence. Along with the new architecture, we proposed a novel hidden unit that includes a reset gate and an update gate that adaptive control how much each hidden unit remembers or forgets while reading/generating a sequence.

We assessed the proposed show with the assignment of factual machine interpretation, where we utilized the RNN Encoder Decoder to score each expression combine in the expression table.

Subjectively, we could demonstrate that the new model can catch etymological regularities in the expression combines well and furthermore that the RNN Encoder Decoder can propose all around shaped target phrases.
The scores by the RNN Encoder Decoder were found to enhance the general interpretation execution.

Our subjective examination of the prepared model demonstrates that it for sure catches the phonetic regularities in different levels i.e. at the word level and expression level. This recommends there might be more regular dialect related applications that may profit by the proposed RNN Encoder Decoder.

References

[1] Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach.*. Pearson Education, 2 edition, 2003.
[2] Christopher D. Manning and Hinrich Schutze *Foundations of Statistical Natural Language Processing.* MIT Press, Cambridge, MA, USA, 1999.
[3] Daphne Koller and Nir Friedman *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning.*. The MIT Press, 2009.

[4]Andrej Karpathy, Justin Johnson, and Fei-Fei Li. *Visualizing and understanding recurrent networks.*. CoRR, abs/1506.02078, 2015

*[5]*Christopher D. Manning Minh-Thang Luong, Hieu Pham. Effective approaches to attentionbased neural machine translation. *AProceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015.*

[6]Kyunghyun Cho Dzmitry Bahdanau and Yoshua Bengio. *Neural machine translation by jointly learning to align and translate..*. ICLR, 2015.

[7]NUS Natural Language Processing Group, 2015.

[8] Sepp Hochreiter and Jurgen Schmidhuber. Long short-term memory. *Neural Comput., 9(8):17351780,*. November 1997.