Machine Translation using Seq2Seq model and Gender Analysis

Parth Shah
parthnay@buffalo.edu

Payraw Salih
payrawsa@buffalo.edu

Abstract

"Long Short Term Memory," effectively known as LSTM, is known to be an effective model for creating long range time dependencies [4]. LSTM blocks are used to encode and decode "sequences" of words from one language to another. Dependencies are found during the encoding process and translations are scored using the bleu scoring system [5]. Using this methodology, translation were made from the German language to English. A best Bleu score of 28.7 was achieved, which is similar to the 29.0 that has previously been published using this method [4]. Furthermore, we did gender analysis and found that 30 percent of sentences have gender bias amongst those sentences which were translated with a bleu score greater than 90.

1 Introduction

Human translation is slowly being aided by the use of machine learning alternatives [6]. Deep neural networks have been known to have major uses in solving complex problems in a cost effective manner. Of the different algorithms available, LSTM has been known to be effective in taking a general problem of translating from one language to another, into an effective vector based approach. The input sentence is taken and encoded into a vector of phrases. Phrases contain dependencies between words and meaning that goes beyond beyond simple word to word translations. LSTM has been known to be useful in creating such long range dependencies such as the dependencies that are present between words in a sentence (gender form, subjects etc.). The problem is therefore simplified into a general sequence to sequence model based approach [4].

The language vocabulary and sentences are sourced from WMT'16. The input is first taken and encoded using LSTM into a vector of time-dependent size. A second LSTM is trained on the input vector to produce the output vector, or sequence that matches the dependencies present in the input vector the closest. What follows is a diagram showing the process[4]:

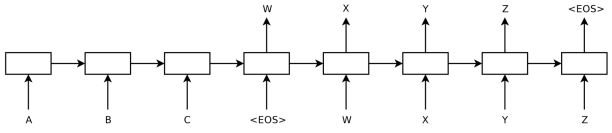


Figure 1: Our model reads an input sentence "ABC" and produces "WXYZ" as the output sentence. The model stops making predictions after outputting the end-of-sentence token. Note that the LSTM reads the input sentence in reverse, because doing so introduces many short term dependencies in the data that make the optimization problem much easier.

As explained in the figure 1 description, the input sentence is read in reverse order to introduce even more dependencies in the sentences [4]. In addition to this, the Bleu scoring system for translation was used to determine the accuracy of the LSTM translations. The scoring system works by taking

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

the geometric mean of precision using unigram, grouped unigram, and bigram scores as explained in the paper [5]. Penalties are given for number of incorrect words used, number of correct words used divided by the total number of words in the translated sentence and finally by taking two words and checking to make sure the order matches. Below is an example of the scoring metric for a reference sentence ”the cat is on the mat”:

Comparing metrics for candidate "the the cat"

Model	Set of grams	Score
Unigram	"the", "the", "cat"	$\frac{1+1+1}{3} = 1$
Grouped Unigram	"the"*2, "cat"*1	$\frac{1+1}{2+1} = \frac{2}{3}$
Bigram	"the the", "the cat"	$\frac{0+1}{2} = \frac{1}{2}$

Thus, LSTM encoding and decoding neural network was used to find dependencies that would be used for translations scored off the Bleu scoring system.

We then begin a discussion into gender bias. We define gender bias as the incorrect translation from one gender to another; or the addition of gender when it did not exist. An example could be assuming that a nurse should be female, or an engineer should be male. These gender biases become apparent when translating between languages that use genitive words (such as German) and languages that do not (such as English).

2 Methodology

2.1 Model

For this machine translation task we used seq2seq model which consist of layers of LSTM cells stacked one over another working in a Recurrent Neural Network. A Recurrent Neural Network or RNN is used when data has a time dimension, i.e. data is sequential. An RNN takes input of current data input as well as the previous state of network into consideration when calculating outputs. Layers of RNN consist of LSTM or Long Short Term Memory cells, which calculate activation value based on update, forget and output gates. These can be expressed by following equations:

$$c^{\tilde{<t>}} = tanh(W_c[a^{<t-1>}, x^{<t>}])$$
$$\Gamma_u = \sigma(W_u[a^{<t-1>}, x^{<t>}] + b_u), \text{Update gate}$$
$$\Gamma_f = \sigma(W_f[a^{<t-1>}, x^{<t>}] + b_f), \text{Forget gate}$$
$$\Gamma_o = \sigma(W_o[a^{<t-1>}, x^{<t>}] + b_o), \text{Output gate}$$
$$c^{<t>} = \Gamma_u c^{\tilde{<t>}} + \Gamma_f c^{<t-1>}$$
$$a^{<t>} = \Gamma_o tanh(c^{<t>})$$

Here $a^{<t>}$ is the activation value at timestep t.

The model is on most basic level an encoder-decoder structure where both are LSTM network. Encoder converts the input sentence into a form which can be interpreted as an understanding of sentence. The decoder uses this understanding to generate a sentence into another language.

Specifically our model is based on the Google Neural Machine Translation(GNMT) model. The encoder of our model consist of a single input Bidirectional LSTM layer which is followed by multiple unidirectional layers. The encoder generates a conditional probability distribution based on the all the words in sentence. This probability distribution is used by encoder as input at first

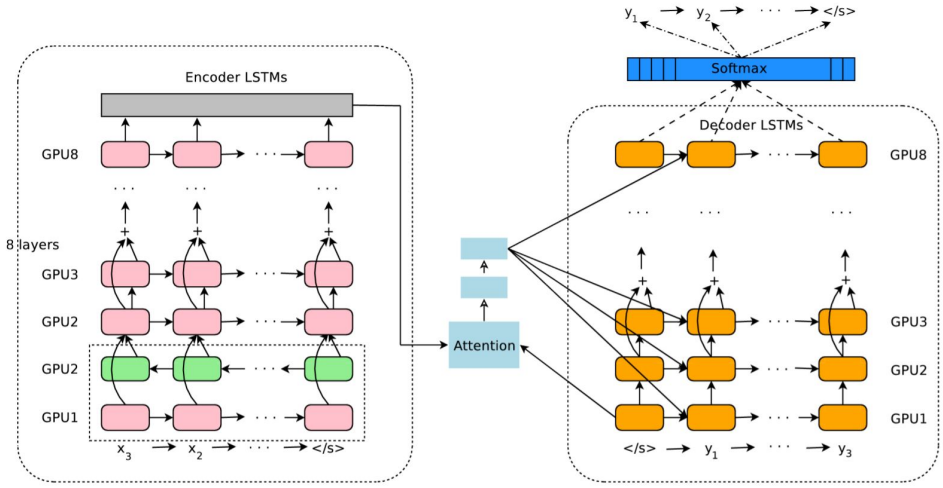
timestep which generate the first sentence of another language, this is again fed to the network to get next word and so on until end of sentence keyword is generated. This can be expressed as:

$$P(y_1, y_2, ...y_{T'}|x_1, x_2, ...x_T) = \Pi_{t=1}^{T'} P(y_t|v, y_1, y_2, ...y_{t-1})$$

In between the Encoder and Decoder there's also an Attention layer. The Attention model improves translation accuracy by increasing weightage of words which play much more important role in a sentence, thus can be said it makes Network to pay more attention to a particular section of sentence. For each prediction a context is taken into consideration, fed by attention layer into decoder layers. This can be expressed as:

$$c^{<t>} = AttentionFunction(y_{t'-1}, x_t) \; \forall t, 1 \leq t \leq T$$
$$p^{<t>} = e^{c^{<t>}} / \Sigma_{t=1}^{T} e^{c^{<t>}}$$
$$a^{<t'>} = \Sigma_{t=1}^{T} p^t . x^t$$

The Attention function is a single hidden layer Feed forward network



Model based on GNMT with attention.

Further the Network also implement Residual architecture by including skip connections in the unidirectional LSTM layers. This improves gradient flow in backward direction.

2.2 Training Details

We trained two models one containing 4 layers and another containing 2 layers. Inputs and outputs of network are of shape [VocabSize, EmbedSize]. Inputs are not actually onehot vectors of sentence to vocabulary but embeddings of size EmbedSize generated for each word using Vocabulary. The Embedding are trained along with the entire network. With a training batch size of 128 and a vocabulary size of 36548 the network is trained for 18998 epoc steps.

Loss function used is sparse cross entropy used in Adam optimizer with a variable learning rate, its value based on:

$$lrate = d_{model}^{0.5}.min(step\_num^{0.5}, step\_num.warmup\_steps^{1.5})$$

Gradient clipping is used to prevent exploding gradient problem. The maximum Norm Gradient value is 5.0.

3 Results & Analysis

3.1 Results

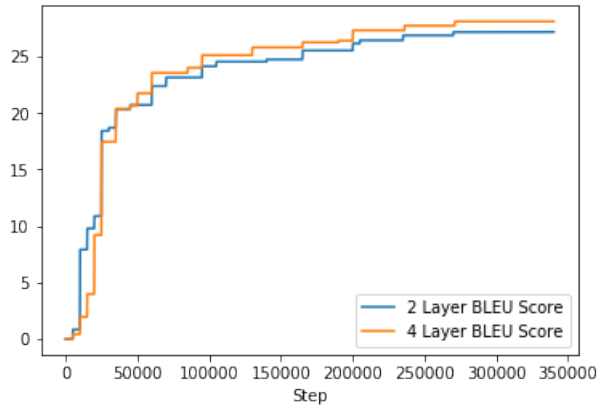


Figure 1: Bleu score of both models plotted against step

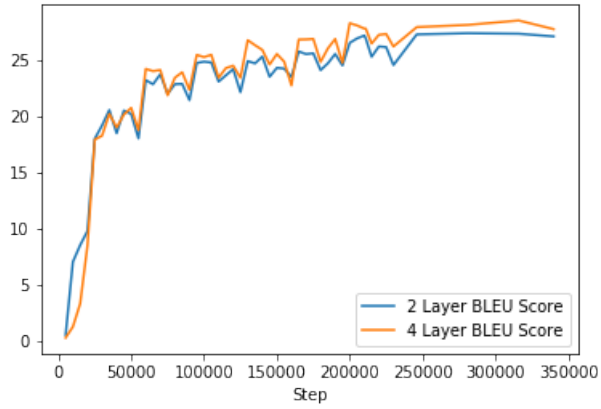


Figure 2: Test set Bleu score of both models plotted against step

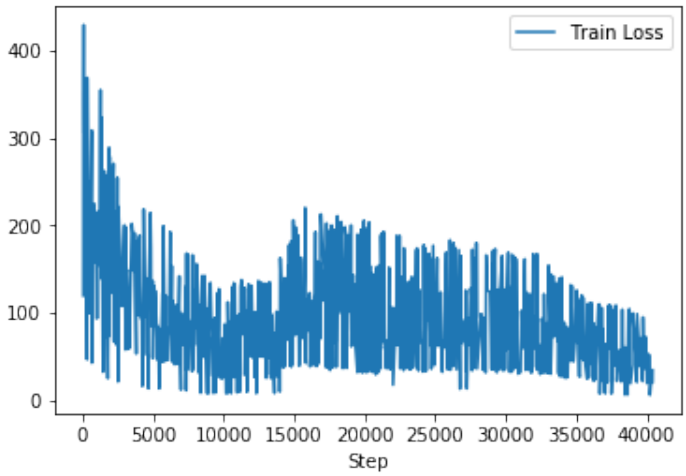


Figure 3: 2 Layer model training loss

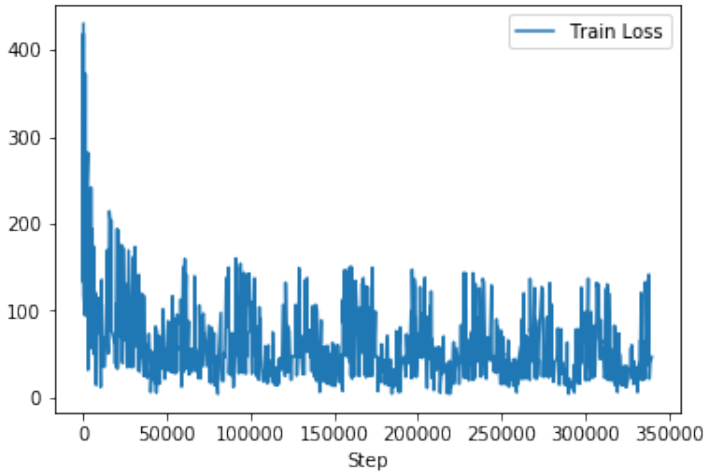


Figure 4: 4 Layer model training loss

System	Score
NMT (greedy)	27.6
NMT + GNMT attention (beam=10)	29.9
Our GNMT attention (2 layers)	27.6
Our GNMT attention (4 layers)	28.7

Table 1: BLEU Scores for the various models

Bleu Score	Percent Gender Bias
90+	36

Table 2: Percent of gender bias that was present

3.2 Analysis

The data used was taken from WMT16's German-English sentence pair dataset. This was obtained through the use of a shell script. This creates folders for storing the data, and downloads the

Europarl v7, Common Crawl, News Commentary v11 corpora into those folders as well as the appropriate test sets. After some extracting and converting, the script puts the separate corpora together and tokenizes them. This tokenization is done in order to add start of sentence and end of sentence tokens for the network to know when to start and end encoding and decoding. The data is also preprocessed in another way. The German words are broken into subword units as well. This was done because german words tend to have a high number of lexemes compared to their English counterparts. Weltmarktführer is a good example of this. This one word means "world market leader", which in English is expressed as three. So the script uses BPE to break this down into subword units, in this case "welt", "markt", and "führer", since these lexemes correspond to "world", "market", "leader" respectively. After doing this the vocabulary files are created for use by the network.

The models we built, the 2 layer GNMT attention model and the 4 layer GNMT attention model both achieved comparable BLEU scores. Table 1 shows the BLEU scores of our models, with scores of 27.6 and 28.7 for the 2 layer and 4 layer models respectively. These results are as expected due to their architectures. The interesting result is that even with cutting the 4 layer network's layers in half, giving us the 2 layer network, the BLEU only decreased by 0.9 points. This seems to suggest the model is rather robust even without the extra structure.

We took a sample of 10000 random sentence translation pairs. Of these 10,000 translation pairs, 9,000 were translated with a bleu score less than 0.9. Of the 1000 that had a bleu score higher than 0.9, only 100 received the perfect 1.0 score. We focus on higher bleu scores because we want to focus mainly on gender bias. From the 1000 remaining translations, 600 contained words that expressed gender. These words are: he, she, her, him, miss, mister, madame, sir, sister, brother, mother, father, daughter, son, girl, boy, women, woman, men, man, female, male, hers, his. We found that of the 600 sentences that contained gender, approximately 180 sentences had translated gender subjectively. An example of a subjective translation include translating the German word for nurse into "she". Example output that came from an input which didn't express gender "the she nurse went to work". This is, in our opinion, due to the fact that "krankenschwester" the word for nurse, is genitive and female. The correct translation would have been "the nurse went to work" If we extrapolate this data then out of 600 sentences that contained gender words; 180 had incorrect gender in the translation. This accounts for 30

4 Conclusions

In conclusion, LSTM DNN was found to be a good step towards a more accurate machine translation [4]. The process itself models input and output sentences as temporal vector based sequences. Translations were made using the dependencies created in the LSTM sequence to sequence model. Translations were scored using the Bleu scoring system, wherein we achieved a high Bleu score of 28.7 compared to the 29.0 found in the paper [4] for German to English translations. We also Did gender analysis and found that 30

5 References

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin. Attention Is All You Need. arXiv:1706.03762 [cs.CL]

[2] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. arXiv:1609.08144 [cs.CL]

[3] Sébastien Jean, Kyunghyun Cho, Roland Memisevic, Yoshua Bengio. On Using Very Large Target Vocabulary for Neural Machine Translation. arXiv:1412.2007 [cs.CL]

[4] Sutskever, I., Vinyals, O., Le, Q. V. Sequence to Sequence Learning with Neural Networks (rep.). Sequence to Sequence Learning with Neural Networks. nips. Retrieved from http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf

[5] ACL ”BLEU : a Method for Automatic Evaluation of Machine Translation.” (rep.). BLEU : a Method for Automatic Evaluation of Machine Translation. Retrieved from aclweb.org/anthology/P/P02/P02-1040.pdf

[6]Ahrenberg, L. Comparing Machine Translation and Human Translation: A Case Study (rep.). Comparing Machine Translation and Human Translation: A Case Study. Retrieved from ”https://www.researchgate.net/publication/322032518_Comparing_Machine_Translation_and_Human_Translation_A_Case_Study”

[7]ACL ”Shared Task: Machine Translation.” Retrieved from http://www.statmt.org/wmt16/translation-task.html