International Journal of Computer Trends and Technology
ISSN: 2231 – 2803 / https://doi.org/10.14445/22312803/IJCTT-V69I6P108

Volume 69 Issue 6, 50-56, June 2021
© 2021 Seventh Sense Research Group®

Original Article

Developing English to Dawurootsuwa Machine Translation Model using RNN

Elias Asefa[1], Hussien Seid[2]

[1]*Addis Ababa science and Technology University, Ethiopia*
[2]*Addis Ababa science and Technology University, Ethiopia*

Received Date: 07 May 2021
Revised Date: 11 June 2021
Accepted Date: 17 June 2021

Abstract *- The idea of language translation is developing recently to solve the issues of linguistic diversity. The translation of English texts into Amharic, Afaan-Oromoo, Tigrigna, China, French, and Somalia are developed. However, as the knowledge of the researchers is concerned, English to Dawurootsuwa machine translation is not developed. This thesis aims to develop unidirectional English to the Dawurootsuwa machine translation model by using Neural Network (NN) approaches. RNN predicted the output text based on the current input and previous output. Under in RNN, an LSTM and GRU contain a neuron, each neuron are replaced with cells having control gates. That used for a memory cell that maintains gates to manage the flow of sentences inaccurate order and fully connected to the model. A parallel corpus, which consists of 20,345 pairs of sentences is prepared from different sources and classified as a 90% training set and a 10% test set. A recurrent Neural Network model with 22 input nodes and 27 output nodes is developed and implemented using Keras toolkit of Python programing language and Adam algorithm. Totally they are contain four results based automatic (BLEU) score and manual evaluation (Arithmetic Mean Value) techniques with hidden layer size of 2. In simple RNN model the BLEU score is 0.5187 with the learning rate of 0.002 and AMV result is 0.60914. In embedding RNN model the BLEU score is 0.5245 with the learning rate of 0.003 and AMV result is 0.60914. In bidirectional RNN model the BLEU score is 0.5452 with the learning rate of 0.004 and AMV result is 0. 0.60914. Finally, in encoder-decoder model the BLEU score is 0.555 with the learning rate of 0.005 and AMV result is 0.60914. And after 0.005 learning rate there is similar score were recorded with the maximum threshold epochs of 100. From the result, concluded that, encoder decoder model of BLEU score 0.555 is fairly good accuracy achieved compare from the rest model and less achieved comparatively from AMV result. In further work to encourage English-Dawurootsuwa parallel corpus improve the accuracy more and minimize the loss and. develop the model from unidirectional to the multidirectional language model.*

Keyword *- Artificial Neural Network, Dawurootsuwa English, Machine Translation.*

I. INTRODUCTION

Language is structured system that regarded as the hallmark of human intelligence and core medium of communication [1] and Translation is a communication between a source-language into relative and equivalent target-language and core tools for the understanding the idea in know language [2]. Natural Language Processing is a motivating area of computational techniques for representing and analyzing of naturally occurring texts [3]. Machine translation is a part of natural language processing that an automated process of text or speech converts from a source language into equivalent and coherent target language. This thesis should implement and investigate the application of neural machine translation techniques for English to Dawurootsuwa machine translation. In Ethiopia, more than 85 languages spoken, in which most of the population only speak a small percentage of all the languages, could certainly benefit from accessing information in different languages [4]. In Ethiopian southern nations, nationalities, and peoples region there are 56 nations own culture, religion, and language. Dawro is a distinguished nation having its own identity expressed by the language of Omotic family, dressing style, feeding habit, way of life, culture and cooperation. The total population that spoken Dawurootsuwa is maximum one million and above. Dawurootsuwa language writing system is based on Latin scrip called Abachada. Abachada is organized by small and capital letters that consists of thirty-five basic letters. Among thirty-five basic letters, five of them are vowels, other five are double consonants and the remaining are consonants [4]. The sentence structure of Dawurootsuwa and English have quit differences in their syntactic structure. In English, the sentence structure is Subject-Verb-Object (SVO) where as in the case of Dawurootsuwa, the sentence structure is subject-object-verb (SOV).

Let E as English sentence and D as Dawurootsuwa sentence. An encoder RNN encode the Dawurootsuwa sentence and converts this encoded sentence $E_1, E_2, E_3 \ldots\ldots, E_M$ into fixed length of vectors.

This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

$$E_1, E_2, E_3 \dots\dots, E_M = \text{Encoder RNN}(E_1, E_2, E_3 \dots\dots, E_M) \quad (1.1)$$

Decoder takes encoded vector into the target sentence.

$$D_1, D_2, D_3 \dots\dots, D_N \quad (1.2)$$

The decoder outputs is using conditional probability at list one word at a time predict.

$$P(D/E) = P(D/E_1, E_2, E_3 \dots\dots, E_M) \quad (1.3)$$

Where $E_1, E_2, E_3 \dots\dots, E_M$ in the equation is the fixed-size vectors encoded by the encoder. In decoding the next word is predicted using symbols that were predicted till now. From the above expression to make a chain rule the conditional probability of the sequence $P(D/E)$ can be decomposed as below here [5].

$$P(D/E) = \prod_{i=1}^{N} P(D_i/D_0, D_1, D_2, \dots., D_{i-1}; E_1, E_2, E_3 \dots\dots, E_M) \quad (1.4)$$

Where D_0 is indicates as a beginning of a sentence. Each neuro is activated by the use of Softmax activation function. It considers about the information from all elements. A combination of an RNN network and softmax layer to implement decoder [6].

LSTM and GRU are different network architecture that has been specially addresses the vanishing and exploding gradient problems explicitly designed to solve the problem that allows learning of long-term dependencies. RNN can't capture long-term dependencies and has limited effectiveness and also learning rate becomes very slow [7]. So, RNN is replaced with LSTM and GRU in MT networks, because they succeed better in this setting. The translation of English texts into Amharic, Afaan-Oromoo, Tigrigna, china, French, and Somalia are developed. However, as the knowledge of the researchers is concerned, English to Dawurootsuwa machine translation is not developed. The basic factor that was initiated for this study is the unavailability of such technologies or systems from and to Dawurootsuwa. The overall objective of this research study is to develop English to the Dawurootsuwa machine translation model using RNN and evaluate the performance of the model. The model works only in one direction from English to Dawurootsuwa. The limitation of this study was lack of publically available corporal in the both language. Due to this reason small sample corpora was prepared.

II. RESEARCH METHOD

A. Data Set Collection

English to Dawurootsuwa parallel corpus were collected from different relevant domain including the Holy Bible, Dawro cultural museum and Dawro educational office. Secondary data sources, like books, articles, publications, unpublished, web and other previously written resources related to the topics.

B. Development Tools

To develop unidirectional English to Dawurootsuwa machine translation model by using a neural network approach, free available tools such as python, Keras toolkit and TensorFlow are used to implement the models. The required libraries and configure values for different parameters that are used in the code are imported. The main focus is on enabling the implementation of RNN models as fast and easy as possible for research and development. The experiment is executed on hp pavilion laptop machine with Intel core i5 5200U CPU and 4 GB RAM. Python 3.5 programming language is used for the development of recurrent neural network classifiers for the proposed language translation system.

C. Data requirements and Preparation

The normal requirement for efficient NMT performance is a large training corpus. The parallel corpus of English to Dawurootsuwa languages is prepared by the help of language experts. To prepare the corpus the following steps are performed.

- ✓ To check syntax, semantic and sense of sentence in both language.
- ✓ To prepare the data in the form of UTF-8 format.

The dataset consists of 20,345 both English sentences and Dawurootsuwa sentences. Simple sentences and complex sentences were collected and prepared. Using a batch size the collected documents were divided into a training set and validation set in such a way that 90% was used for training and the rest 10% was used for the validation set. Implementing a recurrent neural network as part of an end-to-end machine translation pipeline. The following procedures are applied to the collected corpus to make it ready for learning and testing. Such as, preprocessing, modeling, and prediction.

In preprocessing a text is converted into a sequence of integers. Tokenizing and padding are the criteria in preprocessing system. Tokenize and padding the English sentences and the Dawurootsuwa sentences that are longer or shorter than a certain length. It divides a sentence into the corresponding list of words and converts the words to integers. Word level models tend to learn better since they are lower in complexity. The padding of the English sentence and the Dawurootsuwa sentences to be varying the length however, an LSTM, GRU, bidirectional encoder and encoder-decoder are expects input as instances with the same length. So to convert the sentences into fixed-length vectors.

Neural machine translation models are based on the sequence of sequence architectures which, applied to an encoder-decoder architecture that contains two LSTM networks such as in the input (English) side encoder LSTM and in the output (Dawurootsuwa) side decoder LSTM. An encoder LSTM consists of the original language and the decoder LSTM consists of the translated language with a start-of-sentence token. Finally the actual target sentence is

the Dawurootsuwa sentence with an end-of-sentence token. End-of-sentence (Eos) is suffixed to English sentences and the resultant sentence is appended to the Dawurootsuwa sentences list. Similarly, start of sentence (SOS) is concatenated at the start of the translated sentences.

D. Building the Models in Neural Network Architecture

Create the models that take a sequence of integers as source data and returns to back possible translations. After training about the basic types of neural networks that are regularly used for machine translation, design and implement the model. The reason behind that, using this model for more accurate value and in master better understanding how to work the models. For more precise, developing English to Dawurootsuwa machine translation model based on the following criteria. Model 1 RNN implementation, Model 2 RNN with Embedding implementation, Model 3 Bidirectional RNN implementation and Model 4 Encoder-Decoder RNN implementation. Assign the number of batch size which is 64 and built the layers that train the neural network model. The model is contains 18310.5 train data and 2034.5 teste data. An algorithm of Adam optimizer and threshold epoch value of 100.

III. RESULTS AND DISCUSSION

In preprocessing to tokenize the input and output sentences. It can performs basically two tasks:

- It divides the whole sentences into the corresponding list of words.
- Next to converts the list of words to integers.

An example of tokenized sentences as show below.

- ✓ “Dr. Dawit listens to the news every night and then he goes to bed” [4, 5, 6, 1, 7, 8, 9, 10, 11, 12, 2, 13, 1, 14].
- ✓ “My aunt bring ten candies for my brother birth da y” [3, 15, 16, 17, 18, 19, 3, 20, 21, 22].
- ✓ “He must be very intelligent” [11, 183, 20, 72, 200 1].

Besides tokenization and integer conversion, words are the keys and the corresponding integers are the values. Display out the number of unique words and the length of the longest sentence in the input corpus:

Prints the number of unique words and the length of the longest sentence in the input corpus:

Unique English words	Unique Dawuro words	Length of longest sentence in English	Length of the longest sentence in dawuro
8603	20953	22	27

From the contrasting of unique words in the source part and the target part, to conclude that normally English sentences contain smaller and shorter number of words on average, compared to the translated Dawurootsuwa sentences. Next steps is to pad the source sentence. This is because the tokenized sentences can be varying the length, however, neural network layers expects input sentence and output with in same length dimension. Finally, to convert our sentence into fixed-length vectors. Among these sentences are dynamic in length, so to add padding into the end of the sequences to make them the same length. To see that English and Dawurootsuwa sentences are different in length. For example English longest sentences are 22 and Dawurootsuwa longest sentences are 27. Therefore, below 22 and 27 to make zero in empty index and print the output in both languages. Encoder input sequences shape: (20345, 22) Encoder input sequences [0]: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 11, 183, 20, 72 & 2001].

Decoder input sequences shape: (20345, 27) Decoder input sequences [0]: [2, 3341, 9466, 9467, 9468, 3542, 9469, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 and 0]. Pre-padding and post-padding process are made in encoder part and decoder part respectively. In encoder pre-padding are followed by the given values 11, 183, 20, 72 and 2001. Where as in decoder post-padding are done at the last part of the sequence such as 2, 3341, 9466, 9467, 9468, 3542 and 9469. This is because in encoder the original sentence are made in last part so, at the begging made zeros and in decoder the encoded sentence are found at the first part of the sequences so at the last position zeros are added.

A. Model 1 Simple RNN Implementation

This model is a good baseline for sequence data. Building an RNN that translates English texts to Dawurootsuwa texts. This is minimum one feedback connection in the networks. The BLEU score is 0.5187 with the learning rate of 0.002.

Table 3. Simple RNN layer

Type of Neural network Layers	Output Shapes	No of Parameters
simple_rnn_2 (SimpleRNN)	(20345, 20, 32)	1088
gru_2 (GRU)	(20345, 20, 32)	6240
lstm_2 (LSTM)	(20345, 20, 32)	8320
time_distributed_3 (TimeDist)	(20345, 20, 128)	4224
dropout_2 (Dropout)	(20345, 20, 128)	0
time_distributed_4 (TimeDist	(20345, 20, 2517)	324693
Total params: 344,565		
Trainable params: 344,565		
Non Trainable params: 0		

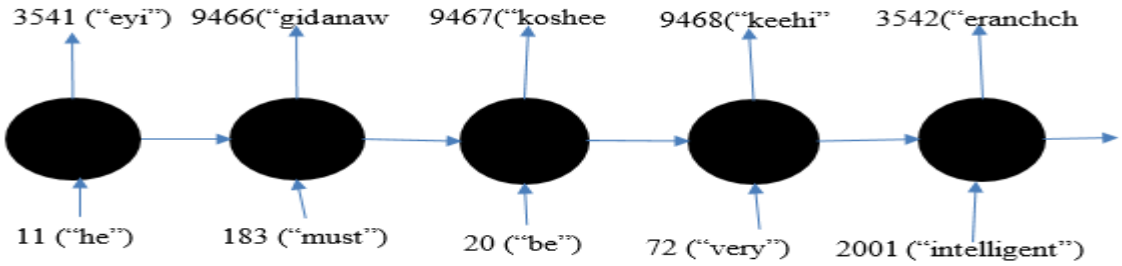

Fig. 3 Model 1 simple RNN implementation

B. Model 2 RNN with embedding implementation

Create an RNN model using embedding word to vector representation. Finally, to predict the output value got the BLEU score is 0.5245 with the learning rate of 0.003.

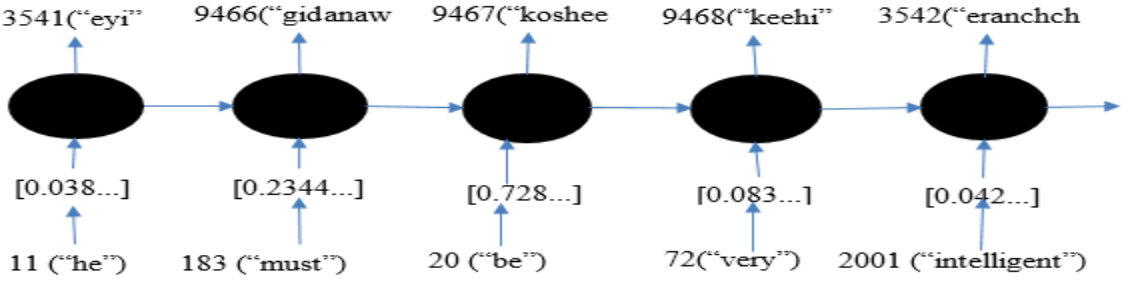

Fig. 3. 1 Model 2 RNN with embedding implementation

C. Model 3 Bidirectional RNN implementation

This has two recurrent cells that scan the information in different ways, one in the forward direction (positive time direction) and the other is backward direction (negative time direction). The reason behind that RNN output depends on previous output and present inputs but cannot see the future input. So that to solve such restriction by using RNN bidirectional recurrent neural networks that involves the future data. No need of delay when incorporating two time direction on one network system. The model was implemented by using tanh activation function. Finally, got the BLEU score is 0.5452 with the learning rate of 0.004.

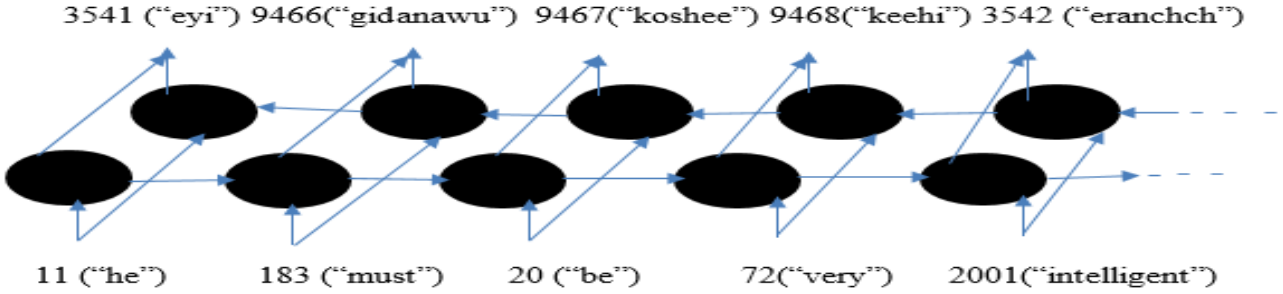

Fig. 3. 2 Model 3 Bidirectional RNN implementation

D. Model 4 Encoder-Decoder RNN Implementation

The number of input sequences and output sequences are vary in length at sequence to sequence prediction time. So an encoder decoder implementation is used to handle map between the sequence of input text and output text. Finally, got the BLEU score is 0.555 with the learning rate of 0.005.

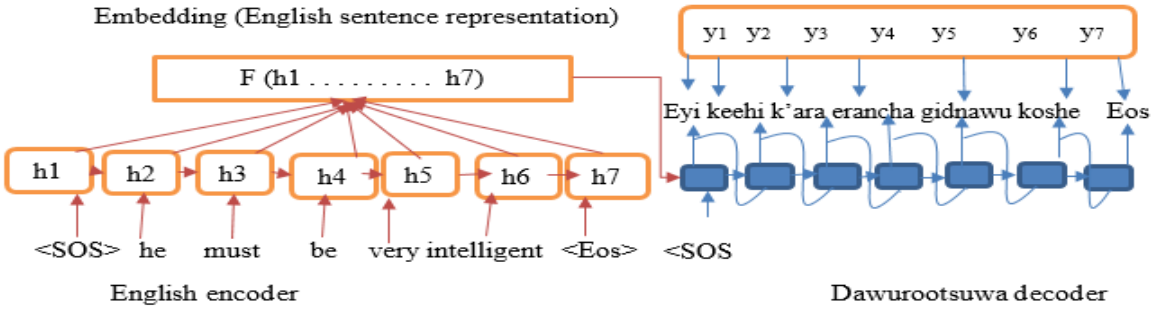

Fig. 3. 3 Model 4 Encoder-Decoder RNN implementation

E. System Architecture

The main component of our NMT model is facilitates the learning of long distance dependencies. The model of conditional probability $p(^{y}/_{x})$ translating a source sentence $x = x_1, x_2 \dots, x_n$ to a target sentence $y = y_1, y_2 \dots, y_m$.

$$minimize - \frac{1}{7}\left[log\left(y_1\ (\text{"eyi"})\right) + - \right. \\ \left. - + log\left(y_7\ (\text{"<EOS>"})\right)\right] \quad (3.1)$$

This model maximize the conditional probability of the parallel training corpus with back-propagation through time.

$$max\ \frac{1}{N}\sum_{n=1}^{N} \log p\theta(y^{(n)}\ |x^{(n)}) \quad (3.2)$$

Where $(y^{(n)}, x^{(n)})$ show that the n^{th} text in the parallel corpus of both size N and θ respectively that considers the set of all tunable parameter. The output of the previous layer is calculated by the multiplication of weight matrix and to add bias that passed on activation function.

$$yk = g(W_{yk-1} + b) \quad (3.3)$$

$$h^{(t)} = gh(W1x^{(t)} + WRh^{(t-1)} + bh \quad (3.4)$$

Here, $WRh^{(t-1)}$ are activity of the previous time-steps that multiplied by a recurrent weight matrix.

$$y^{(t)} = gy(Wyh^{(t)} + by \quad (3.5)$$

The encoding process can be visualized as the input sequence being compressed by the RNN into an intermediate representation in the form of a fixed dimensional vector. So, if the vector h_{t-1} describes the history of the sequence at time-step, the new internal state (the updated vector) h_t will be computed by the network, effectively compressing the preceding symbols x_1, x_2 ,…..x_{t-1} as well as the new symbol x_t . The following equation shows this:

$$h_t = \sigma_\theta(x_t, h_{t-1}) \quad (3.6)$$

Here, σ_θ *is* a function which takes the new information unit x_t and the hidden state h_{t-1} as input. It passed through a nonlinear function for the implementation of the recurrent neural network activation function σ_θ. This is shown by the following equation:

$$ht = tanh(Wxt + U_{ht-1} + b) \quad (3.7)$$

Here, Wxt is the input weight matrix, U_{ht-1} is the recurrent weight matrix and b denotes the bias vector. Now, our basic RNN model $p(xt|x < t)$ at each time t to be described by

$$p(xt|x < t) = g\theta(ht - 1) \quad (3.8)$$

$$ht - 1 = \phi\theta(xt - 1, ht - 2) \quad (3.9)$$

g_θOutputs a probability distribution that is conditioned on the entire history up to h_{t-1}. Thus, the RNN tries to predict the next token at each time-step.

IV. EXPERIMENT AND TESTING

In this section detail explanation of the experiment and show the result on graphically.

- ✓ In uniform distribution between -0.1 and 0.1 all LSTM parameters are initialized.
- ✓ Used Adam and stochastic gradient descent (SGD) optimization algorithm for updating network weight about fixed learning rate of 0.005. After 100 epochs, to begin the learning rate every half epoch with hidden layer size of 2. Adam optimizer is faster than SGD at training time.
- ✓ In each model there is Activation function(s) at each neuron which, is activating.

Recall that the total number of input sentences are 20345 and the total number of dawuro unique word is 20953 and dawuro longest sentence is 27 and the final shape of the Dawurootsuwa sentence should be include (input_size, length-out-size, and output_vocabulary_size) that means (20345, 27 and 20953) respectively. To make predictions, the dense layer is final layer of the model in this study.

Table 4. Flow chart of input RNN layer

input_1: InputLayer	Input:	(20345,22)
	Output:	(20345,22)
embedding_1: Embedding	Input:	(20345,22)
	Output:	(20345,22,100)
input_2: InputLayer	Input:	(20345,22)
	Output:	(20345,22)
lstm_1: LSTM	Input:	(20345,22,100)
	Output:	[(20345,32),(20345,32),(20345,32)]
embedding_2: Embedding	Input:	(20345,22)
	Output:	(20345,22)
lstm_2: LSTM	Input:	[(20345,27,32), (20345,32), (20345,32)]
	Output:	[(20345,27,32), (20345,32), (20345,32)]
dense_1: Dense	Input:	(20345,27,32)
	Output:	(20345,27,20953)

A table show that it contains two input layers. The first input indicate Input-1 that represents the placeholder for the encoder and passed through the lstm_1 layer. This lstm layer includes hidden layer, cell state and output. Among this process only the hidden state and cell state are passed through the decoder. The second input indicate Input_2 that represents the placeholder for the decoder LSTM. Finally the decoder output passed through dense layer and predict the output. For example to see translated sentence from English to Dawurootsuwa.

Step 1: he must be very intelligent -> Encoder -> Enc (h1, c1)

Enc (h1, c1) + <SOS> -> Decoder -> eye+ Dec (h1, c1)

Step 2: Enc (h1, c1) + eye -> Decoder -> gidanawu + Dec (h2, c2)
Step 3: Enc (h2, c2) + gidanawu -> Decoder -> koshee + Dec (h3, c3)
Step 4: Enc (h3, c3) + koshee -> Decoder -> keehi + Dec (h4, c4)
Step 5: Enc (h4, c4) + keehi-> Decoder -> erancha + Dec (h5, c5)
Step 5: Enc (h5, c5) + erancha-> Decoder -> <Eos> + Dec (h6, c6)

To understand from this translation the encoder output is input for decoder and predict the future word. If the output process is continues until the <Eos>.
Input
Word Ids: [1461, 53, 1072, 484, 954]
English Words: ['dawit', 'have', 'never', 'drink', 'milk']

Prediction
Word Ids: [579, 1999, 1995, 281, 275, 1]
Dawro Words: dawite maatha dola ushi erena. <Eos>

All the predicted outputs from the decoder are then concatenated to form the final output sentence. Plotting the graphs are provided below.

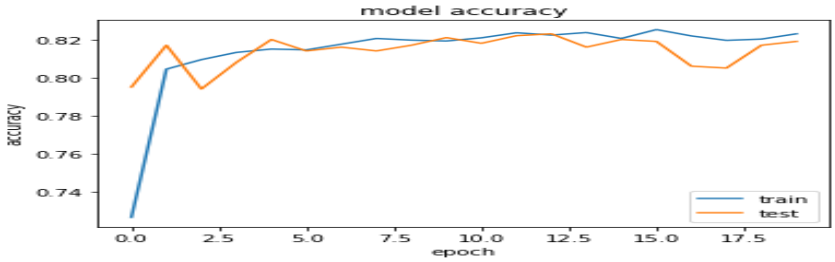

Fig. 4. 1 Plot of model accuracy on train and validation datasets

In graph 4.1 the accuracy of train and test datasets is still rising for the last 100 epochs.

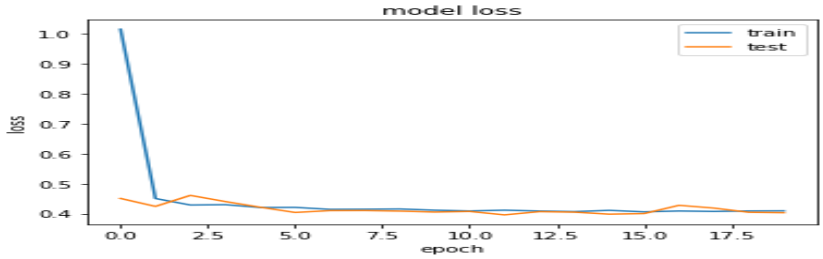

Fig. 4. 2 Plot of model loss on train and validation datasets

A. Manual Evaluation Techniques

As the knowledge of language experts English to Dawurootsuwa manual translated corpora was prepared and this prepared corpora was compare with automatic translated text. As this time to take 101 sample of sentences and evaluate the average value by using arithmetic mean value (AMV) got 0.60914.
To calculate AMV taking five language experts those are Mr. A, Mr. B, Mr. C, Mr. D and Mr. E. all this experts compare their translated text with machine translated text and putted their own decided value.

1st To calculate single text average value.

$$AMV\ on\ single\ text = \frac{calculate\ the\ sum\ of\ experts\ \ in\ single\ sentence}{number\ of\ experts}$$

So, number of experts are five and compared decide text value are: -

$$AMV\ on\ single\ text = ((0.79 + 0.785 + 0.79 + 0.78 + 0.775)/5)$$
$$= 0.784$$

2nd To calculate total text average value.

$$AMV\ on\ 101\ text = \frac{calculate\ the\ sum\ of\ experts\ \ in\ total\ sentence}{number\ of\ total\ sentence}$$

$$AMV\ on\ 101\ text = ((0.784 + 0.712 + 0.705 + \cdots + 0.485)/101)$$
$$= 60.914$$

Experts manual compare English to Dawurootsuwa text with machine translation text and assign the value in between zero (0) and one (1). Below the table shows sample of evaluation techniques.

Table 4. 1 Experts Manual Evaluated English to Dawurootsuwa Tex

No of S	Mr. A	Mr. B	Mr. C	Mr. D	Mr. E	Average Value
S_1	0.79	0.785	0.79	0.78	0.775	0.784
S_2	0.72	0.70	0.72	0.70	0.72	0.712
S_3	0.69	0.715	0.715	0.69	0.715	0.705
S_4	0.59	0.575	0.58	0.57	0.585	0.58

S_5	0.65	0.665	0.665	0.65	0.665	0.659
S_6	0.59	0.59	0.60	0.60	0.61	0.598
S_7	0.725	0.71	0.70	0.73	0.72	0.717
S_8	0.65	0.63	0.645	0.64	0.635	0.64
S_9	0.70	0.715	0.715	0.70	0.70	0.706
S_{10}	0.695	0.68	0.695	0.69	0.68	0.688
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
S101	0.45	0.50	0.475	0.49	0.51	0.485
						0.60914

V. CONCLUSION

In this research thesis, neural machine translation was applied. NMT with pre-trained word embedding performs better than complex translation techniques on Dawurootsuwa languages. Since we achieved fairly good accuracy so our model can be used for creating English-Dawurootsuwa translation applications that should be an advantage in domains such as tourism and education. Moreover, to explore the possibility of using the above techniques for various English Dawurootsuwa language translations.

Modeling of English to Dawurootsuwa machine translation is based on the neural machine translation approach in which tokenize and padding are used in the preprocessing step. The design process of modeling English to Dawurootsuwa machine translation by using Python programming language in Keras toolkit and training by using the different models such as simple RNN, RNN with embedding, bidirectional RNN and encoder-decoder RNN implementation. The system involves collecting English to Dawurootsuwa parallel corpus, corpus preparation that also involves dividing the corpus as a training set and test set. We have been at the beginning satisfied that the LSTM would fail on a long sentence due to its restriction on memory. The experiment was conducted by using the collected data set to check the accuracy and efficiency of the system by using neural network approaches. The system carried out based on the 90% trained and 10% tested dataset and consequences were 20345 data was recorded.

Totally they are contains four results based automatic (BLEU) score and manual evaluation (Arithmetic Mean Value) techniques with hidden layer size of 2. In simple RNN model the BLEU score is 0.5187 with the learning rate of 0.002 and AMV result is 0.60914. In embedding RNN model the BLEU score is 0.5245 with the learning rate of 0.003 and AMV result is 0.60914. In bidirectional RNN model the BLEU score is 0.5452 with the learning rate of 0.004 and AMV result is 0.60914. Finally, in encoder-decoder model the BLEU score is 0.555 with the learning rate of 0.005 and AMV result is 0.60914. And after 0.005 learning rate there is similar score were recorded with the maximum threshold epochs of 100. From the result, concluded that, encoder decoder model of BLEU score 0.555 is fairly good accuracy achieved compare from the rest model, but less achieved comparatively from AMV result. In future work develop the model from unidirectional to the multidirectional.

ACKNOWLEGMENTS

First, I'd like to pass my deepest gratitude to the Almighty God, who gave me the strength and health to gain whatever I have acquired so far. I was not able to complete this thesis without the kind assistance of many individuals. I desired to thank all the excellent people who supported and accompanied me during the progress of this thesis. I also thanks gratefully my guide, Dr. Hussien Seid, whose excellent and enduring guide shaped this work considerably and made the process of creating this thesis invaluable learning experience. My deepest acknowledgement is going to college of electrical and mechanical engineering for giving me financial support for research data collection. I also thanks department of electrical and computer engineering those who assign learning environment and research supervisor. I desired to thank Dawro Zone Educational Development for giving me valuable research data. Especially, Mr. Mitiku Woldesenbet, Mr. Dawit Mekonnin, Mr. Kidane Zeleke and Mr. Frehiwot Samuel, they have given me not only the essential data but also they arrange experts to check the overall linguistic information of the data. Finally special thanks to my best family who continually assisting and encouraging me with their exceptional wishes.

REFERENCES

[1] R. Sebastian, Neural transfer learning for natural language processing, Galway, (2019) 1-16.

[2] T. McARTHUR, The oxford companion to the english language, T. McARTHUR, Ed., New York: oxford unversity press, (1992) 10, 51-54.

[3] N. Prakash, O. Lucila and C. Wendy, Natural language processing, Journal of the American Medical Informatics Association, september (2011).

[4] W. Hirut, Revisiting Gamo: Linguists' classification versus self identification of the community, International Journal of Sociology and Anthropology, 5(9) (2013) 373-380.

[5] W. Yonghui, S. Mike, C. Zhifeng, V. Quoc and M. Norouzi, Bridging the Gap between Human and Machine Translation, computation and language, 1609 (2016).

[6] I. Sutskever, O. Vinyals and Q. V. Le, Sequence to sequence learning with neural networks, In Advances in Neural Information Processing Systems, (2014) 3104–3112.

[7] Y. Bengio and X. Glorot, Understanding the difficulty of training deep feedforward neural networks, in Proceedings of the Thriteenth International Conference on Artificial Intellegence and Statistics, Sardinia,Italy, (2010).