International Journal on Emerging Technologies **11**(1): 148-153(2020)

ISSN No. (Print): 0975-8364
ISSN No. (Online): 2249-3255

Corpus Augmentation for Neural Machine Translation with English-Punjabi Parallel Corpora

Simran Kaur Jolly[1] and Rashmi Agrawal[2]
[1]Research Scholar, Faculty of Computer Applications, MRIIRS, Faridabad, India.
[2]Professor Faculty of Computer Applications, MRIIRS, Faridabad, India.

(Corresponding author: Simran Kaur Jolly)

(Received 26 October 2019, Revised 8 January 2020, Accepted 18 January 2020)
(Published by Research Trend, Website: www.researchtrend.net)

ABSTRACT: Earlier research on machine translation showed that phrase -based sentence alignment approach was a robust approach for noisy text. As the data increased for low resource languages corpus-based machine translation approaches were used for aligning sentences in two different languages. The quality of a Neural Machine Translation system and Statistical Systems depends largely on the size of corpora being build. As the amount of data increased, an end to end system was used having less dependencies and low latency. This system was called as neural network machine translation system. The study described below uses different sentences and dataset's for sentence alignment in machine translation. Comparing all the models on corpus is a long and tedious process hence we try to identify a common parameter for development of a good corpus for low resource languages and improving the accuracy of the proposed algorithm. For low resource languages, it is not the situation here, so we use a data augmentation technique that targets least occurring words in the corpus and apply statistical and neural based models on the corpus.

Keywords: Parallel Corpus, RNN (Recurrent Neural Networks), LSTM (Long short-term memory), PBMT (Phrase based machine translation systems), NMT (Neural machine translation systems), SMT (Statistical Machine Translation Systems), alignment, source language, target language.

Abbreviations: RNN (Recurrent Neural Networks), LSTM (Long short-term memory), PBMT (Phrase based machine translation systems), NMT (Neural machine translation systems), SMT (Statistical Machine Translation Systems).

I. INTRODUCTION

A large-scale parallel corpus is an important resource for machine translation for filtering out the low-quality sentences in corpora. Large corpora are limited to similar languages but monolingual corpora for low resource languages are easily available. Parallel Text is an important resource for natural language processing tasks such as machine translation and word sense disambiguation. Sentence alignment is an important aspect of translation while modelling the relation between source sentence and target sentence [16].

Machine translation is a process of converting source sentence in one language to target sentence in another language. The first system for machine translation was started in 1949 by Weaver. These models progressed towards statistical phrase-based systems using lexicon and parallel corpora not producing accurate results. These models were dependent on phrases in the sentence for generating the output not capturing the long-term dependencies. Due to these limitations neural machine translation systems were introduced which is an end to end system translating long sentences as well. Various approaches have been applied for creating parallel corpus. For example, Lamb *et al*., (2016) proposed a pseudo parallel technique to create corpus based on machine translation [1]. The sentence alignment processes are based on length, lexicon or mixture of two techniques as reviewed by Torres-Ramos and Garay-Quezada (2015) [13].

Alignment models are collection of models related to statistical machine translation. These models train the translation model starting with lexical probabilities to word re-ordering. The problem in the sentence alignment is of existing approaches on equivalent translations from source and target language sentences. The second issue is aligning positions of source and target language sentence. These techniques perform well on close language pairs such as English-French parallel text but for remote languages like English-Punjabi sentence alignment is a challenging task. The third issue is compounding and modality in Indian languages. The sentence below shows distortion in alignment between languages. Sennrich *et al*., (2015) worked on back translation from target language to source language pair [2]. They automatically translated target language into source language and obtained a pseudo alignment between two language pairs.

The background of machine translation in Indian languages several systems were implemented on rule based and statistical based models. The major translation systems were ANGLABHARTI-II (English to Indian languages), ANUBHARTI-II (Hindi to any other Indian language), ANUVADAKSH (English to six other Indian languages), ANGLAMT etc. These systems were based rule-based models or hybrid models. Punjabi is a widely spoken language in Canada and India having more than 100 million users. The ANGLA-MT system translates English to Indian languages using a pseudo-interlingua approach.

The translation quality of ANGLA-MT compared to google translate was very poor. Google developed a neural machine translation system for Indian languages in 2017 including Punjabi.

The contribution of the paper: The main contribution of the paper is exploring different parameters that affect the machine translation quality from English to Punjabi. This paper also focuses on adding data augmentation technique to improve the existing model and how the sentence alignment parameter can affect the translation quality of our algorithm. The dataset used in the paper are sentences build in form of a corpus by crawling it from ted talks, TDIL, Wikipedia, Bible and Sri-Guru-granth-Sahib.

II. RELATED WORK ON SENTENCE ALIGNMENT

Most of the work done on sentence alignment earlier were focused on phrase-based models. In phrase-based models, sentence alignment approaches have been used for translating on the basis of phrase matching hence not capturing long term dependencies. These approaches were categorized on the basis of length, word match and cognate matching. Word based alignment model by Brown *et al.,* (1993) used a source channel model where target language is generated by a source language having some probability [6]. Parallel text has been used in many different ways for machine translation and Sentence alignment techniques. In statistical Machine translation aligned parallel documents are used for building phrase tables and computing n-gram probabilities out of the table. Manually aligning sentences by humans is quite a costly task as it requires lot of cost involved. So automatically aligned corpora is used for the purpose of machine translation as it increases the quality of target output. The length-based alignment technique works well on highly correlated languages like English-French but for languages having less correlation length-based techniques doesn't give accurate results. The Berkley aligner Liang *et al.,* (2006) [9] shows recent advances in word alignment using both supervised and unsupervised learning. It is basically extension of cross word aligner and has more advantages as it uses results from the previous corpora and aligned corpora. The aligner breaks down the document into source and target documents which further divides the documents into k partitions. Each partition is assigned a vector value '0' or '1', where '1' is the vector bin where partitions are aligned) are more robust approaches as it finds missing words in bilingual sentence pairs as well as word alignment errors. This approach tells us the relationship between confidence measure and alignment quality which further helps in improving sentence alignment. The LDC word aligner allows from many to many alignments by converting the entire sentence into a graph. If the graph is completely connected then the alignment is correct otherwise not. The problems that were raised while using length based and word-based techniques were the compounding and modality issue in the parallel language pair. Hence further the alignment techniques were based on generative alignment models. These models were more accurate as they solved the deficiency problem in both the source and target strings in generative models chunk based alignment is done by involving variables that affect the probability of occurrence of the chunks. The other aligners such as Microsoft aligner Moore (2002) [10], Hun align Varga *et al.,* (2007) are basically autonomous aligner tools that uses a word-based alignment from that texts to be aligned [7]. The limitation of these aligners are short sentences are not aligned that affects the performance of the tools. These aligners work on the word-based models but due to ubiquity of corpus-based techniques in the alignment process use of parallel text is given more consideration. van der Wees *et al*., (2017) presented a dynamic selection approach for filtering the out of the domain data and calculate its loss function [19].

Dhariya *et al*., (2017) proposed a hybrid approach for machine translation from Hindi to English using rule-based approach that applies grammar rules on the lexicon. The drawback of this approach was that large dictionary is needed for matching the grammar rules from one language to another language [18].

Wang *et al*., (2018) proposed a model that embeds both statistical and neural translation model as one single unit [5]. This modelling technique works well on parallel corpus that converts each and every word to target word and removes unk symbols in the translation.

In a probabilistic model translation is generated finding a sentence in target language that maximizes the probability of occurrence of the equivalent sentence in source language [10]. The probabilistic model for machine translating had several limitations, large number of components and lack of generalizability in the components. While, in neural machine translation model a parallel training corpus is fitted to maximize the translation probability arg max p (target | source). After learning the probability distribution of the model given the sentence in source language corresponding sentence in target language is searched by matching the random index in the vocabulary.

Cho *et al.,* (2014) was the first group to introduce the concept of neural machine translation: RNN (recurrent neural network) Encoder Decoder [3]. The firs neural machine translation system was successful by google and Facebook called as open NMT. They also added attention mechanism into their models for further accurate translations. The neural machine translation system consists of two main components: encoder and decoder. Recurrent neural networks with long short term memory units have better results for English to French translation task [4].

Bahdanau *et al*., (2015) proposed attention-based mechanism for neural machine translation adopted from encoder decoder mechanism [8]. The basic encoder-decoder mechanism suffered from limitation of translating long sequences in a corpus. Hence attention-based mechanism for translation was adopted. The sentences in corpus are sequence of words arranged by some rules. Translating source sentence to target sentence is done by hidden units in neural networks.

$$C_t = \mathrm{f}\,(W_x \mathrm{x(current\ word)} + W_r C_{t-1}$$

In the above equation C is the current state of the hidden network when input is fed into feed forward neural network, x is the current word in sequence that is dependent on output from previous function as well. Hence at each time step t it calculates the value of the C. Hence recurrent neural networks capture long term dependencies.

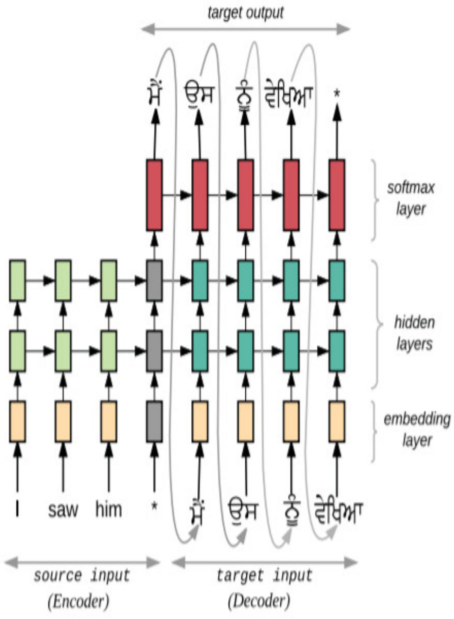


Fig. 1. Encoder-decoder.

III. PREVIOUS MODEL USING SUPERVISED LEARNING

The baseline model that has been implemented on our parallel corpus is encoder-decoder mechanism. In the parallel corpus crawled from internet and open sources, we have input language sentences (s) and output language sentences (t). In a neural machine translation system, it finds the maximum probability given the target sentence as output. The above is achieved through encoder-decoder mechanism. The encoder creates a vector representation for every sentence and decoder find the logarithmic value of probability, hence generating output sentence.

$\log p(\frac{t}{s}) = \sum_{t=1}^{m} \log p(^{t_t}/_{t1 - t_{t-1}}$, e)

Neural machine translation has shown good results for English and European language pairs like French, German and Spanish. The easily available neural network is seq to seq neural network called as recurrent neural network. There are different categories of rnn available depending on the number of layers and gates in the network. The most widely used neural network is lstm's (long short term memory) depending on their properties like layers, directionality and gates. In English to Punjabi translation the baseline model considered is lstm. The following steps are followed:

1. The lowermost layer takes input sentence form source language followed by delimiter signifying end of one sequence

2. These sentences are fed into embedding layers to get converted into continuous representations.

3. The initial state of the encoder is prepared via zero vector whereas decoder is primed using preceding state of the encoder. Lastly, the output from the top hidden layer from the decoder side is altered using SoftMax function into a likelihood distribution over the target language and a transformation is retrieved in form of target language sentence.

Candidate:['ਹੱਡੀਆਂ', 'ਵਿੱਚ', 'ਦਰਦ' 'ਨਿਰੰਤਰ', 'ਬੁਖਾਰ', 'ਚਾਹੇ', 'ਇਹ', 'ਘੱਟ', 'ਹੋਵੇ', 'ਜਾਂ', 'ਸ਼ਾਮ' ,'ਤੱਕ', 'ਵਧਦਾ', 'ਜਾਵੇ', 'ਹੱਡੀਆਂ', 'ਦਾ', 'ਵਿਗਾੜ', 'ਹੋਣ', 'ਦੇ', 'ਨਾਲ', 'ਨਾਲ', 'ਦਰਦ', 'ਵੀ', 'ਟੀ', 'ਦੇ', 'ਲੱਛਣ', 'ਹਨਹੈ']

Reference 1:
['ਹੱਡੀਆਂ', 'ਵਿੱਚ', 'ਦਰਦ', 'ਨਿਰੰਤਰ', 'ਬੁਖਾਰ', 'ਚਾਹੇ' ,'ਇਹ', 'ਘੱਟ', 'ਹੋਵੇ', 'ਜਾਂ', 'ਸ਼ਾਮ', 'ਤੱਕ', 'ਵਧਦਾ', 'ਜਾਵੇ', 'ਹੱਡੀਆਂ', 'ਦਾ', 'ਵਿਗਾੜ', 'ਹੋਣ', 'ਦੇ', 'ਨਾਲ', 'ਨਾਲ', 'ਦਰਦ', 'ਵੀ', 'ਟੀ', 'ਦੇ', 'ਲੱਛਣ', 'ਹਨ']

Reference 2:
['ਅਸਥੀਈਆਂ', 'ਵਿੱਚ', 'ਨਿਰੰਤਰ', 'ਬੁਖ਼ਾਰ', 'ਨੂਮ੍', 'ਦੁੱਖ', 'ਦੀਜਿਯੇ', 'ਕਿ', 'ਇਹ', 'ਹੇਠਾਂ', 'ਹੈ', 'ਨਹੀਂ','ਸੀ', 'ਪੀੜ', 'ਦੇ', 'ਨਾਲ', 'ਅਸਥੀਈਆਂ', 'ਦਾ', 'ਸ਼ਾਮ', 'ਦੀ', 'ਬਦਸੂਰਤੀ', 'ਨੇ', 'ਵਧਾਣਾ', 'ਟੀ' ,'ਬੀ', 'ਦਾ', 'ਲੱਛਣ', 'ਹਨ']

IV. PROPOSED UNSUPERVISED LEARNING FOR SENTENCE ALIGNMENT IN TRANSLATION

Despite the popularity of recurrent neural networks for machine translation, it is not able to capture long term dependencies and unknown words in corpus based neural machine translation. The limitation was the words in source sentences were converted to fixed size vectors. To overcome this limitation words that occur more frequently in source sentences to predict the target words in target sentences is deployed in the unsupervised learning. This mechanism is called attention mechanism in neural machine translation. In this mechanism the vectors depend on the number of words in the source sentence.

In this mechanism some words from source sentence are converted into vectors (s1…sn). The number of vectors in the source words are mapped to the attention vectors in the attention layer. The vectors in attention layer are the deciding factor to generate target words globally. The attention vector scores are generated by dot product of the current word vectors from source and target sentence.

In the proposed mechanism multiple neural translation models are trained on the single language pair individually with different parameters. The framework used for sentence alignment is the encoder-decoder framework. In the encoding stage the source sentence is converted into vectors h. in the decoding stage in a particular layer computation takes place as follows:

s_i^l = y

In the above equation s_i is the sentence and y are the word embedding of that sentence. When dealing with words in the corpus, there are million numbers of tokens in the corpus, so to avoid high computation wastage embeddings are used in the neural networks. To solve this limitation an extra layer is inserted into the neural network. Embedding layer are a fully connected layer having weights of the matrix. The multiplication of the matrix is ignored and value of weight matrix id grabbed. Instead of doing the matrix multiplication, we use the weight matrix as a lookup table. We encode the words as integers, for example "heart" is encoded as 958, "mind" as 18094. Then to get hidden layer values for "heart", you just take the 958th row of the embedding matrix. This process is called an embedding lookup and the number of hidden units is the embedding dimension.

In neural machine translation for sentence alignment we follow approach of translation augmentation which focuses on sentences having low frequency words [14]. This technique has been implemented in convolutional neural networks to change the image properties but preserving it labels. The approach works as follows:

1. If we have a source and target sentence pair (s, t), we change it in such manner that it doesn't changes the meaning of the sentence but changes the syntax.

2. There are number of instances to do it, such as rephrasing (parts of) S or T. but it is a tough task and does not guarantees good results. Hence a list of words that rarely occur is included in the dictionary.

3. Thus, the goal of our data augmentation technique is to give more importance to rare words and for this we search the entire monolingual corpora and replace frequently occurring word with rare words. For e.g.

Eng.: On Wednesday, August 8, a family to the west of the split were gathered/grouped in their lounge.

Punjabi: ਬੁੱਧਵਾਰ, 8 ਅਗਸਤਨੂੰ, ਵੰਡਦੇਪੱਛਮਵੱਲਇੱਕਪਰਿਵਾਰਨੂੰਉਨ੍ਹਾਂਦੇਲੌਂਜਵਿੱਚਇਕੱਠਾ/ ਸਮੂਹਕੀਤਾਗਿਆਸੀ.

Sentence Decoding Alignment Algorithm for Low Resource Languages (SAL): The sentence decoding alignment algorithm for machine translation proposed for low resource languages augments a cost-based approach along with the translation probabilities (statistical approach). In the algorithm we embed a stochastic gradient descent that selects sentences having lowest cost among the sample subset.

For e.g.: English to Punjabi translation "the picture is nice" is translated to "**ਤਸਵੀਰਚੰਗੀਹੈ**"

The picture: **ਤਸਵੀਰ** (0.9); The picture: **ਚੰਗੀ** (0.07);The Picture: **ਹੈ** (0)

Hence, we can see that translation probabilities related to the phrase pair is the highest hence it is the best candidate translation. Along with this we embed translation augmentation mechanism in our algorithm for reducing the out of vocabulary words as well. For all the set of sentences S in corpus C following input and output values are considered.

Input: Set of pair of the sentences: (e, p) e: English p: Indian language like Hindi/Punjabi; l: length of English sentence, m: length of Indian language sentence N: no. of sentences i: input language word, j: output language word

Output: Sentence alignment (A), t (p/e) (translational probability of target language given input language) In order to compute these parameters we need to pick sentences from different language and take a normalization factor called μ (which calculates the conditional probabilities of target language sentence conditioned on input language sentence.)

Procedure: Translate

(a) Initialize all parameters alignment and translation probability to random values.

(b) for each n in [1, ..., N] do

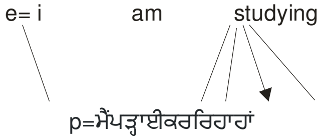
(c) for each i in [1, ..., i(n)] do

(d) for each j in [1, ..., j(n)] do

(e) if alignment = j then, (alignment of input language = alignment of output language)

(f) Count the p, e words> ++ (increment the alignments too). Count English words too

(g) for each Punjabi and English word: $\mathrm{p(p, a|e) = p(I|J)\pi p(a|J). p(p|e)}$



The alignment here is (1, 4)

(h) t(f/e) = count(e|p)/count(e) (count number of times two words are aligned in a corpus)

(This equation calculates the value of t parameter which counts the number of words of both input and output language.)

(i) A_{ml} (j/i, l, m) = (count(j|ilm)/count (i, l, m) (sentence alignment parameters)

(This equation will be calculating the sentence alignment of machine translation by counting the number of times word j appear in the sentence given i, l and m.)

The algorithm described above involves decoding over the source sentences using following heuristics:

– Aligned Target words: the model chooses middle point as alignment point between two sentences. The model uses nearest neighbor algorithm for alignment.

– Aligning source words: the model aligns source words by visiting them again for aligning untranslated source words.

V. DATASET DESCRIPTION

A good corpus plays an important role in machine translation tasks. The available parallel corpus is for English Hindi languages. We build English Punjabi parallel corpus by crawling corpus from ted talks, Wikipedia, newspaper articles, TDIL, EMILLE and domain-based corpus requested from TDIL. The TDIL corpus includes domain specific corpus for domains like health, tourism, agriculture and entertainment. There were several mismatches between source and target sentences and other languages in the corpus such as Malayalam.

VI. EXPERIMENTS

We evaluate the effectiveness of above proposed algorithm and Nmt system on the translation tasks between English and Punjabi.

For low resource language settings, we randomly sample 15% of the English and Punjabi bilingual corpus. For baseline experiments we are considering the iterative based statistical machine translation model for sentence alignment. In the below Table 1, we back-translate sentences from the target side that are not included in our model by keeping two constraints: here we keep 1:1 sentences, we also consider sentences having 1:2 and 1:3 alignments. We measure translation quality by single reference case-insensitive BLEU computed with the bleu metrics [12].

For evaluating the bleu score on the corpus tokenized dataset was used. The bleu score with the above described parameters is computed. This model learns the word order of English and Punjabi without any reordering dependencies as needed in statistical translation models. Once the dataset is preprocessed, the source and target files are fed into the encoder layer to prepare the vectors from the sentences. We have used Stochastic gradient descent (SGD) [13], an algorithm for training the built corpus which is embedded in our SAL algorithm. Here we tried to use different parameters and different layer sizes.

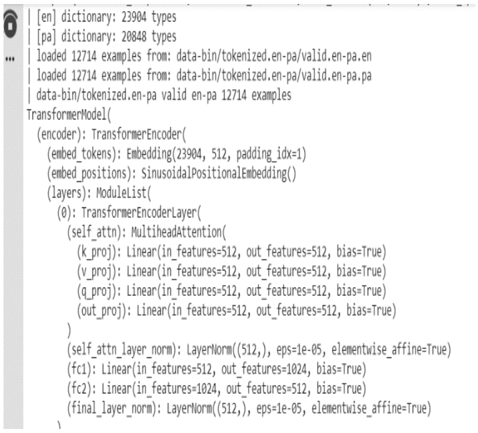
Table 1: Bleu score for statistical Machine Translation using Iterative sentence alignment.

Statistical Sentence Alignment	Precision(lax)	Gold pairs	1:1 1:2 1:3(bleu score)
Hun align	0.38	None (as it was length based)	0.38
Iteration-1(gale and church)	0.08	137	0.58
Iteration-2	0.025	193	0.73
Iteration-3	0.04	72	0.808
Iteration-4	0	25	0.71
Iteration-5	0.384	97	0.82(best bleu score)

We measure translation quality by single reference case-insensitive BLEU computed with the bleu metrics [12].

Table 2: Parameters for proposed algorithm.

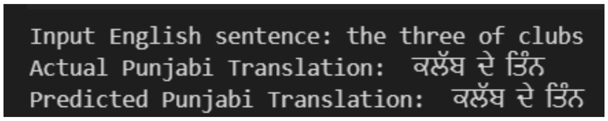
Parameters	Values
Data	Tokenized-pa
Arch	fconv_wmt_en_de
Lr	0.5
Clip normal	0.1
Max tokens	12000
Force anneal	50
BP	0.668
Label smoothing	0.1
Time taken	27.5 seconds
Sentence translated	15754



```
| [en] dictionary: 23904 types
| [pa] dictionary: 20848 types
| loaded 12714 examples from: data-bin/tokenized.en-pa/valid.en-pa.en
| loaded 12714 examples from: data-bin/tokenized.en-pa/valid.en-pa.pa
| data-bin/tokenized.en-pa valid en-pa 12714 examples
TransformerModel(
  (encoder): TransformerEncoder(
    (embed_tokens): Embedding(23904, 512, padding_idx=1)
    (embed_positions): SinusoidalPositionalEmbedding()
    (layers): ModuleList(
      (0): TransformerEncoderLayer(
        (self_attn): MultiheadAttention(
          (k_proj): Linear(in_features=512, out_features=512, bias=True)
          (v_proj): Linear(in_features=512, out_features=512, bias=True)
          (q_proj): Linear(in_features=512, out_features=512, bias=True)
          (out_proj): Linear(in_features=512, out_features=512, bias=True)
        )
        (self_attn_layer_norm): LayerNorm((512,), eps=1e-05, elementwise_affine=True)
        (fc1): Linear(in_features=512, out_features=1024, bias=True)
        (fc2): Linear(in_features=1024, out_features=512, bias=True)
        (final_layer_norm): LayerNorm((512,), eps=1e-05, elementwise_affine=True)
      )
```

Fig. 2. Parameters for training.

Since we have used GPU, training time for the neural network for different datasets for different architectures was in few hours only. For the experiment, we have used a different number of sentences for each data set. Every dataset is trained for 50 epochs at one time. The bleu score for the baseline system was evaluated on scale of 0-1 and it is computed in Table 3. The bleu score for the proposed algorithm using the Neural Machine Translation model was computed in Table 4. The bleu score is 25.8 for range 1-10 for ngram-4 model.



```
Input English sentence: the three of clubs
Actual Punjabi Translation:  ਕਲੱਬ ਦੇ ਤਿੰਨ
Predicted Punjabi Translation:  ਕਲੱਬ ਦੇ ਤਿੰਨ
```

Fig. 3. Translation after training of the corpus.

Table 3: Bleu Score for Supervised Learning.

Languages	Dev sentences	tokens	Avg sentence length	Bleu Score (Statistical)	Bleu score(seq2seq)	Bleu Score(attention)
English	1000	1359	14	0.38	0.56	0.67
Punjabi	1000	1359	14			

Table 4: Bleu score for Unsupervised Learning of the SAL Algorithm.

Languages	Test Sentences	tokens	Model	Bleu Score

English	15754	19440
		types
			BP=0.668,
			ratio=0.712,
			syslen=64493,
			reflen=90524
				28.01, 54.3/43.6/37.7/34.6

Punjabi	15754	17024 types		

VII. DISCUSSION

The sample representations of our algorithm are described below:

S-6568 Space is not important.
T-6568 ਜਗ੍ਹਾਮਹੱਤਵਪੂਰਣਨਹੀਂਹੈ. (google translate)
H-6568 -1.1491457223892212
ਡਰਾਇਵਉੱਤੇਕੋਈਖਾਲੀਨਹੀਂਹੈ (best)

S-354 Undo the last move
T-354 ਆਖਰੀਚਾਲਰੱਦਕਰੋ. (google translate)
H-354 -0.7714384198188782 ਆਖਰੀਚਾਲਵਾਪਸਲਿਆਓ (best)

The system was trained in an end to end system hence aligning the sentences and giving adequacy and fluency according to the vocabulary learned. It learns the word order while translating from English to Punjabi language as well. In future we will incorporate more rare words in the vocabulary and to translate the rare words while it is decoding.

VIII. CONCLUSION

Statistical Phrase-based Machine translation systems have been facing the problem of accuracy and condition of large data sets for an extended time, and in this study, we have considered the possibility of using a narrow RNN and LSTM based Neural Machine translator for solving the issue of Machine Translation for low resource languages. We have used quite a small amount of dataset and a smaller number of layers for our experiment due to system limitations. The results show that NMT can provide much better results for the bigger dataset and have a huge number of layers in encoder and decoder. Compared to contemporary SMT and PBMT systems, NMT based MT performs much better. In future we can use Bert fused neural networks for training of long sentences having rarely occurring words. It could be explored on Indian language pairs as-well. As Indian languages have same language structure so a high bleu value is expected while training the model on different corpus.

ACKNOWLEDGEMENTS

The dataset used in the study has been requested from TDIL (Technology Department for Indian Languages). I would like to thank TDIL for supporting in providing monolingual data for Punjabi language.

Conflict of Interest. No potential conflict of interest was reported by the authors.

REFERENCES

[1]. Lamb, A. M., Goyal, A. G. A. P., Zhang, Y., Zhang, S., Courville, A. C., & Bengio, Y. (2016). Professor forcing: A new algorithm for training recurrent networks. In *Advances In Neural Information Processing Systems* (pp. 4601-4609).

[2]. Sennrich, R., Haddow, B., & Birch, A. (2015). Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

[3]. Cho, K., Van Merriënboer, B., Bahdanau, D., & Bengio, Y. (2014). On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

[4]. Luong, M. T., Le, Q. V., Sutskever, I., Vinyals, O., & Kaiser, L. (2015). Multi-task sequence to sequence learning. *arXiv preprint arXiv:1511.06114*.

[5]. Wang, X., Tu, Z., & Zhang, M. (2018). Incorporating statistical machine translation word knowledge into neural machine translation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *26*(12), 2255-2266.

[6]. Brown, P. F., Pietra, V. J. D., Pietra, S. A. D., & Mercer, R. L. (1993). The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, *19*(2), 263-311.

[7]. Varga, D., Halácsy, P., Kornai, A., Nagy, V., Németh, L., & Trón, V. (2007). Parallel corpora for medium density languages. *Amsterdam Studies In The Theory And History Of Linguistic Science Series 4*, *292*, 1-7.

[8]. Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

[9]. Liang, P., Taskar, B., & Klein, D. (2006). Alignment by agreement. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics* (pp. 104-111). Association for Computational Linguistics.

[10]. Moore, R. C. (2002). Fast and accurate sentence alignment of bilingual corpora. In *Conference of the Association for Machine Translation in the Americas* (pp. 135-144). Springer, Berlin, Heidelberg.

[11]. Gale, W. A., & Church, K. W. (1991). A Program for Aligning Sentences in Bilingual Corpora. Proceedings of *29th Annual Meeting of the Association for Computational Linguistics*, 177–184.

[12]. Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics* (pp. 311-318). Association for Computational Linguistics.

[13]. Torres-Ramos, S., & Garay-Quezada, R. E. (2015). A Survey on Statistical-based Parallel Corpus Alignment. *Research in Computing Science*, *90*, 57-76.

[14]. Ghader, H., & Monz, C. (2017). What does attention in neural machine translation pay attention to?. *arXiv preprint arXiv:1710.03348*. 1-10

[15]. Kaur, S., & Agrawal, R. (2018). A Detailed Analysis of Core NLP for Information Extraction. *International Journal of Machine Learning and Networked Collaborative Engineering*, *1*(01), 33-47.

[16]. Jolly, S., & Agrawal, R. (2019). A Broad Coverage of Corpus for Understanding Translation Divergences. *International Journal of Innovative Technology and Exploring Engineering, 8*(8), 613-618.

[17]. Jolly, S., & Agrawal, R. (2018). Improvement in Machine Translation from English to Punjabi by Identifying the Morpheme Boundaries. *JETIR*, *5*(8), 565-570.

[18]. Dhariya, O., Malviya, S., & Tiwary, U. S. (2017). A hybrid approach for Hindi-English machine translation. In *2017 International Conference on Information Networking (ICOIN)* (pp. 389-394).

[19]. van der Wees, M., Bisazza, A., & Monz, C. (2017). Dynamic data selection for neural machine transaltion. In Proceedings of the *2017 Conference on Empirical Methods in Natural Language Processing*, p.p. 1400–1410.

How to cite this article: Jolly, Simran Kaur, & Agrawal, Rashmi (2020). Corpus Augmentation for Neural Machine Translation with English-Punjabi Parallel Corpora. *International Journal on Emerging Technologies*, *11*(1): 148–153.