Review of Statistical and Neural Machine Translation for Nguni Languages

Deevesh Beegun
bgndee001@myuct.ac.za
Department of Computer Science
University of Cape Town
South Africa

ABSTRACT

Machine translation is a prominent sub-field of Computational Linguistic. Its main purpose is to automatically translate text from one language to another using computers. While there are many variants of machine translation models that have been developed over the years, Statistical Machine Translation (SMT) and Neural Machine Translation (NMT) are the most dominant models in this field. SMT has been the state of art in the machine translation paradigm in the last decades. However, it was outperformed by the NMT which showed greater improvement in translation performance over other traditional translation methods. Nonetheless, NMT has a steeper learning curve with respect to the amount of training data thus underperforms when the amount of data is limited, as in the case of low resource languages. In this literature review, we will give some background on SMT and NMT and we will review the different techniques that have been proposed to solve this performance issue of NMT models when trained in low resource settings. Finally, we conclude and discuss on how our research will proceed.

CCS CONCEPTS

• Computing Methodologies → Artificial Intelligence → Natural Language Processing → Machine Translation.

KEYWORDS

Neural Machine Translation, Statistical Machine Translation, Low Resource Language, Statistical Models.

1 INTRODUCTION

Machine Translation - using computers to translate one language to another - is a prominent sub-field of Computational Linguistic. It was previously used to translate scientific and technical documents [12]. However, with the advent of the internet, the need for immediate online translation increased which gave rise to a large amount of demand for translation [12]. These demand mainly include translating a large amount of text which are then edited by a human translator (human aided machine translation), translating text in communication services such as emails, chats and many others [12]. Due to its various uses it has been extensively studied over the last decades.

In this literature review, we will look at Statistical Machine Translation (SMT) and Neural Machine Translation (NMT) which are two important sub-fields of the Machine Translation paradigm. While phrased based SMT has been the state of art for more than a decade, the emergence of NMT which showed better performance has rapidly replaced SMT [10]. However, NMT has a steep learning curve with respect to the amount of training data, thus underperforms when there is a lack of data, as in the case of low resource settings [17]. The fact that NMT can only be trained on parallel data, which are relatively sparse, especially in low resource languages further contributes to its degradation in performance.

Several approaches have been proposed to solve this issue. The approaches that we discuss in this paper include making use of monolingual data as additional parallel data [21], creating pseudo parallel data using data augmentation, using the knowledge learned from a model (parent model) trained on high resource language to train another model (child model) using transfer learning [26]. All of these approaches attempt in some way to improve the performance of NMT models on low resource languages.

Since Neural Machine Translation has become the main focus of current researches in the machine translation field, this paper will focus more on NMT and the underlying techniques being used to resolve the problem faced when translating low resource languages.

The rest of this paper is organised as follows: Section 2 will discuss the divergence in languages, specifically in Bantu languages. Section 3 will describe how the quality of translation gets evaluated. Section 4 will give an overview of Statistical Machine Translation. Section 5 will give some background on Neural Machine Translation and we will review some of the techniques in modelling low resource languages. Section 6 will provide some discussions about the different techniques that we saw in the previous section. Finally, in section 7 we conclude and discuss how our research will proceed.

2 DIVERGENCE IN LANGUAGES

While there are many similarities in the way people communicate in different languages. These languages often differ from each other in various ways [14]. This difference often makes machine translation a challenging problem.

IsiZulu and isiXhosa are both low resource, Bantu languages that belong to the Nguni language group [15]. Bantu languages have a rich noun class system, subject-verb-object (SVO) word order and a complex structure due to their agglutinating morphology [24].

3 TRANSLATION EVALUATION

The quality of translated sentences output from a Machine Translation model can be evaluated on two aspects, namely: fidelity and fluency. Fidelity is how well the translated sentence maintains the meaning of the source sentence and fluency is how clear or grammatically correct the translated sentence is [22].

3.1 Human evaluation

Two groups of human judges, namely: a monolingual group (understands only one language) and a bilingual group (understands two languages) can be formed to rate the fluency and fidelity of the translated sentences on a scale of 1 (very bad) to 5 (very good). The former can judge the output sentences based on how readable and fluent they are, while the latter can judge the fidelity of the output sentences [22].

While human evaluation has proven to be extremely valuable in evaluating machine-translated sentences. It is often time-consuming, expensive and not re-useable [8]. This can potentially result in a bottleneck in the development of machine translation models.

3.2 Automatic Evaluation: BLEU

The Bilingual Evaluation Understudy (BLEU) method of evaluating translation was proposed by Papineri et al [20]. It is fast, inexpensive, provides an objective view and strongly correlates to human evaluation [8]. It works by comparing n-grams (sequence of n words) of a machine translated output with n-grams of an equivalent human translated text and count the number of matches between them. This represents the precision measure, p_n which is modified to eliminate repetitions. Otherwise, over-generated words by the machine translation would result in absurd but high precision measurements [20]. This can be generalised for a multiple sentence test corpus as follows:

$$P_n = \frac{\sum_{S \in C} \sum_{ngram \in S} Count_{match}(ngram)}{\sum_{S \in C} \sum_{ngram \in C} Count(ngram)} \tag{1}$$

where S is the machine translation output, C is the complete test corpus.

To prevent sentences that are too short from getting a high precision, a Brevity penalty (BP) can be added over the corpus.

BP can be calculated as follows:

$$BP = \begin{cases} 1 & c > r \\ e^{\left(1-\frac{r}{c}\right)} & c \leq r \end{cases} \tag{2}$$

where r is length of the human translated text and c the length of the machine translated text.

Equation (2) outputs a 1 if the c is greater than r. Otherwise a penalty factor of $e^{\left(1-\frac{x}{y}\right)}$ is applied.

The BP can then be applied to the BLEU score as follows:

$$BLEU = BP \times \left(\prod_{n=1}^{4} p_n \right) \tag{3}$$

4 STATISTICAL MACHINE TRANSLATION

The idea of applying statistics to Machine Translation was first suggested by Warren Weaver [23]. Statistical Machine Translation (SMT) is based on this idea and has been the most researched Machine Translation Method over the last decades [18]. It uses statistical models where the parameters of the model are estimated from a parallel corpus - a large body of translated text from one language to another. This model can then be used to translate new sentences which it has not encountered before [18].

There are several ways to translate a sentence from one language (source) to another (target) which can often lead to ambiguity [2]. This ambiguity is a result of divergence in languages. As a result every target-source sentence pair can be assigned a probability $p(S \mid T)$, where T is the target sentence and S is the source sentence. [2]. This probability can be interpreted as the probability that T is the correct translation of the sentence S. It can be written as follows:

$$p(S \mid T) = p(S) \times p(T \mid S) \tag{4}$$

where $p(S)$ is the language model probability and $p(S \mid T)$ is the translation model probability.

The main objective of a machine translation model is given a target sentence T, to chose the source sentence S that maximises the probability $p(S \mid T)$

4.1 Language Model

A language model computes the probability of a word sequence occurring by predicting each individual word given the words preceding it in the sequence [2]. Given a sequence of words $W = (w_1, w2, w3, ...w_n)$ it's probability can be computed using the chain rule as follows: [16]

$$p(W) = p(w_1) \prod_{i=2}^{n} P(w_i \mid w_{1:i-1}) \tag{5}$$

Since, there might be a large number of words preceding a particular world in the sequence, it would be impossible to calculate this probability, thus n-gram models are used. N-gram models consider only n-1 preceding words to predict the n^{th} word. For instance, trigram models consider only the last two preceding words to predict the third one. [16]

4.2 Translational Equivalence

Trnaslational equivalence describes a set of rules to transform a source sentence to a target sentence. These rules can be extracted from the parallel corpus. Finite-state transducer (FST) and Synchronous context-free grammars are most used as translational equivalence models [18].

4.3 Finite State Transducers (FST)

Finite state Transducers are generalised from Finite state automata. In addition to the elements present in a finite automaton it consists of a finite set of output symbols. When an input is given to the FST in a particular state, it transitions to another state and produces an output [11].

4.3.1 Word-based models. This model produces a source sentence from a target sentence (target-to-source). Each target word generates a number of source words. This number is called the fertility of the target sentence. The length of the source sentence can be determined by the summation of the fertility of each word of the target sentence [18]. The translation of individual words is represented by a single target word from each source word. The translated words

are permuted into their final order at the end of the translation process[18].

4.3.2 Phrase based models. In phrase based model contiguous sequences of words are translated as a unit. The phrase word improves on the word based model by preventing the translated word to be in the incorrect order [18]. Each source phrase is translated to a target phrase and translation is atomic [18]. In general, phrase based translation perform better than word based translation.

5 NEURAL MACHINE TRANSLATION

Neural Machine Translation (NMT) is a recently introduced paradigm that has achieved state of the art performance outperforming traditional machine translation methods [1]. It models the machine translation process by using a single, large Neural Network that takes as input a sentence and outputs its corresponding translation one element at a time, in an end-to-end fashion [1]. This allows all parameters of the model to be simultaneously changed to maximise translation performance [1].

5.1 Encoder-Decoder Model

Most Neural Network translation relies on the encoder-decoder network [1]. The encoder in an encoder-decoder or sequence-to-sequence network takes an arbitrary length input and produces a fixed-length vector representation of the input, as output. This output is known as the context vector which is used by the decoder to produce a variable-length translation of the input[4].

5.1.1 RNN Encoder-Decoder. This Neural Network architecture was proposed by Cho et al [5]. Given an arbitrary length input sequence $x = (x_1, x_2, ...x_n)$ and another arbitrary length output sequence $y = (y_1, y_2, ...y_n)$. This model learns a conditional distribution over the two sequences as follows:

$$p(y \mid x) = \prod_{n=1}^{N} p(y_n \mid y_n, x) \tag{6}$$

where x is the source text and y the target text.

The RNN Encoder-Decoder consists of two Recurrent Neural Networks which can be simultaneously trained to maximise the conditional log-likelihood:

$$\max_{\theta} \frac{1}{N} \sum_{n=1}^{N} log p_\theta \Pr(y \mid x) \tag{7}$$

where θ is the set of parameters to be learn by the model, x the input sequence and y, the output sequence from a training set.

The first RNN, the encoder reads a sequence of inputs $x = (x_1, x_2, ...x_n)$ and updates some hidden states, h_t at time t according to the following equation:

$$h_t = \sigma(h_{<t-1>}, x_n) \tag{8}$$

where σ is a non-linear activation function.

After reading the input sequence, the last hidden state represents the context vector, c of the whole input sequence.

The second RNN, the decoder uses the context vector, c as its initial hidden state. It generates an output sequence by predicting y_t in a hidden state, h_t. y_t and h_t both depends on the previous hidden state, the previous output y_{t-1} and the context vector, c [5].

This can be formally represented as follows:

$$p(y_t \mid \{y_1, y2, ..., y_{t-1}, c\} = \sigma(y_{t-1}, h_t, c) \tag{9}$$

where σ is a non-linear activation function.

5.1.2 Attention mechanism. With the encoder and decoder separated, the decoder only knows about the source text through the context vector. The Neural Network must compress all the information from the source text into the final state of the encoder. This makes the Neural Network unable to give correct translations for longer sentences [1].

The attention mechanism proposed by Bahdaunau et al [1]. solves this issue by making all information from all the hidden states in the encoder available to the decoder instead of only the last hidden state. This is achieved by taking a weighted sum of all the hidden states in the encoder and creating a fixed-length vector from the weighted sum. This fixed-length vector is then used by the decoder. The weighted sum changes depending on the current token that is being processed by the decoder thus making the context vector dynamic [1].

5.1.3 Transformers. While Recurrent Neural Network and other sequence-to-sequence models have been the state of the art in the machine translation paradigm, their inherent sequential nature prevents them from being parallelised [22]. This causes their performance to degrade over long sentences, due to memory constraints limit batching across data [22]. A new model architecture, transformers, proposed by Vaswani et al. rely solely on attention mechanism allows more parallelism.

The transformer has a similar overall architecture to neural sequence models with an encoder that compresses an input sequence to a fixed-length vector and the decoder uses this fixed-length vector to produce an output sequence one element at a time [22]. However, the transformer uses stacked, pointwise, fully connected layers for both the encoder and decoder [22]. The encoder consists of six stacked identical layers with each layer containing two sublayers. The first layer is a multi-head self-attention mechanism, and the second layer is a fully connected feed-forward network. These two sublayers consist of a residual connection around them, followed by a layer normalization [22]. The decoder contains similar layers and connections as the encoder with the addition of a multi-head attention layer which performs multi-head attention on the output of the encoder [22]. Since the decoder is auto-regressive the self-attention layer stack is also modified to prevent the decoder from being conditioned on future words.

The attention function in the transformer takes as input a query and key-value pairs and returns the weighted sum of the values [22]. The query, key-value pairs and output by the function are all vectors. A compatibility function is used to compute how much weight to assign to each value. The transformer uses the Scaled Dot-product Attention where the dot product of the query of dimension, d_q is computed with the keys of dimension,d_k. Which is then divided by

$\sqrt{d_k}$ and a softmax function is applied to obtain the weights on the values.

5.2 Transfer Learning

Transfer Learning in Machine Translation is a way of improving the translation performance of low resource language model (child model) by making use of the parameters of a trained high resource language model (parent model) [26]. This allows the child model to start with the weights of the parent model, instead of starting with some random weight thus having some prior distribution over the child model [26]. In this way, the parent model transfers its knowledge to the child model thus reducing the amount of data required for training the low resource language model.

Zoph et al. [26] showed that transfer learning improves the performance of the baseline Neural Machine Translation model on low resource language and its performance being on par with or even outperforming a strong syntax-based machine translation (SBMT) system for one language pair. They also showed that without the use of transfer learning there is a large gap between the performance of strong syntax-based machine translation (SBMT) and Neural Machine Translation (NMT), with SBMT significantly outperforming NMT models. In addition, they found that the translation result can be optimized by fixing some parameters in the parent model and allowing the child model to change only some of the parameters. They claim to have obtained a large BLEU score for four language pairs that they have trained using this technique with one language pair even outperforming the SBMT baseline model. However, as mentioned by Dabre et al. [7], this study did not investigate the performance of other language pairs.

Q.Nguyen et al. [19] whose study extends from the study conducted by Zoph et al. showed that the performance of Neural Machine Translation on low resource language pairs can be improved by using the transfer learning method combined with Byte Pair Encoding (BPE) [1]. The two studies differ in the following ways: in the study conducted by Zoph et al., a parent model is trained on a high resource language pair and transfer the learned parameters to a child model which is then trained on a low resource language pair whereas in the study conducted by Q.Nguyen et al. a parent model is trained on a low resource language and transfer the learned parameters to a child model which is then trained on a low resource, but related language pair. Furthermore, unlike the transfer learning approach conducted by Zoph et al. which assigns a parent source word embedding to a random child source word, this method exploits vocabulary overlap between the parent model and the child model and similar vocabularies keeps their embedding when the transfer learning occurs. However, for this to work the data needs to be processed. This is done by transliteration – convert from one script to another and segmentation – break words into sub-words. This could add an overhead to the development time of the model. In addition, Q.Nguyen et al. showed that the transfer learning proposed by Zoph et al does not always work in low resource settings. Nevertheless, by combining it with BPE, NMT performance could be improved by exploiting its lexical similarity with another low resource language. It should be noted that this study was conducted on only agglutinative languages and thus may not work on other types of languages.

[1]BPE is a segmentation algorithm that treats words as sequences of character tokens and merges similar token pair into one [19].

5.3 Using monolingual data

The performance of a Machine Translation model depends on the amount of data the model has been trained on; consequently, the more data we have the better our model will perform. Neural Machine Translation uses only parallel data for training which are often sparse, especially for low resource languages [21]. On the other hand, there is a substantial amount of monolingual data available for the target language.

Sennrich et al. [21] showed that monolingual training data can be treated as additional parallel training data which could improve the quality of NMT systems by mixing monolingual target sentences into the training set. Their study exploits the fact that the Encoder-Decoder model of the Neural Machine Translation can also act as a language model in addition to a translation model. They proposed two techniques to achieve this: the first one treat monolingual training example as parallel examples with empty source side. This would result in an uninformative context vector thus the network relies entirely on previous target words for prediction. This makes the model perform two tasks at the same time, that is when the source side is empty the model performs the role of a language model otherwise, the model performs the role of a translation model. The second technique that the author proposed is to pair monolingual training instances with a synthetic source sentence from which a context vector can be approximated. These synthetic data is obtained by translating the monolingual target text into the source language (back-translation). The main benefit of this approach is that the neural machine translation architecture does not need to change to incorporate monolingual training data. It can thus be used for other Neural Machine Translation systems. However, as mentioned by Bulot et al., [3] generating artificial data using back-translation are computationally expensive since it needs to translate a large amount of data.

In another a study conducted by Vu Hoang et al. [11], they showed that applying back-translation multiple times improves the quality of translation compared to simple back-translation. This is based on the idea that better artificial data can be created by using a better back-translation system. They conclude that this approach can improve the Neural Machine Translation performance in the case of both high resource and low resources.

Currey et al. [6] integrate target side monolingual data into low resource Neural Machine translation by simply making the source sentences identical to the target sentences of a monolingual corpus. Thus, creating a copied corpus which is mixed with the parallel corpus to train the NMT. This method is similar to that proposed by Sennrich et al. However, instead of using a null source sentence, it uses the same source sentence as the target language. The authors observe that there is an improvement in translation performance when the copied corpus is mixed with the parallel data set. However, this improvement is only apparent in low resource language pairs and not on high resource language pairs. They also hypothesised that this improvement is as a result of the model learning to pass appropriate words through to the target output more successfully i.e, because of increased pass through accuracy. The author also

mentioned that source side monolingual data also improve translation performance. This collaborates the study performed by Zhang et al. [25] Surprisingly, adding more monolingual data consistently yields small improvements and using more monolingual data than parallel data do no affect the translation performance of the model. However, this technique of mixing a copied corpus to the parallel data set might add noise to the NMT system and this has not yet been studied by the authors [6].

5.4 Data Augmentation

Data Augmentation in Machine Translation is a technique used to generate more data for Low Resource Language from existing ones. Thus, improving the performance of low resource language translation [9].

Xia et al. [9] proposed methods for creating augmented or pseudo-parallel Low Resource Language using back translation from English (ENG) to Low Resource language (LRL) or High resource language (HRL) and converting HLR – ENG dataset to a pseudo LRL – ENG dataset. Instead of back translating from the target language to the source, which Currey et al.[6] showed that it is ineffective when data is sparse, they translated the target language to a highly related HRL which can then be simultaneously trained with the LRL-ENG dataset. The HRL-ENG dataset can then be easily converted to a pseudo-LRL-ENG dataset since the HLR and LRL are syntactically similar. They showed that under extreme low-resource settings, this technique can increase translation quality by up to 1.5 to 8 BLEU points compared to the supervised back-translation baselines.

5.5 Hard attention

This type of attention was proposed by Indurthi et al.[13] Unlike the soft attention based neural machine translation which computes the context vector by calculating a weighted sum of all the tokens in the input sequence which is not effective for longer sequences. This proposed model selects only a few relevant tokens across the whole input sequence thus handling longer sequences more effectively. The authors shows that soft attention NMT models are consistently outperformed by hard attention based NMT models and the gap between their performance become more apparent as the length of the sequence increases. They used a transformer architecture for the encoder-decoder where the second sub-layer of the transformer encoder-decoder was replaced with an Reinforcement Learning agent-based attention mechanism.

6 CONCLUSION

In this review, we have identified that although Neural Machine Translation models perform significantly better than Statistical Machine Translation when trained on a large amount of parallel data, its performance is worst than that of SMT when trained on low resource languages. However, there has been a large number of studies performed to increase the performance of NMT on low resource languages. These studies as mentioned in previous sections have shown promising results with considerable improvements in BLEU scores.

We will proceed with our research by implementing a Neural Machine Translation on low resource Bantu languages, more specifically, isiZulu and isiXhosa. From the above review, it seems reasonable to use monolingual data as additional parallel data to increase the performance of the NMT model.

REFERENCES

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

[2] Brown, P. F., Cocke, J., Della Pietra, S. A., Della Pietra, V. J., Jelinek, F., Lafferty, J., Mercer, R. L., and Roossin, P. S. A statistical approach to machine translation. *Computational linguistics 16*, 2 (1990), 79–85.

[3] Burlot, F., and Yvon, F. Using monolingual data in neural machine translation: a systematic study. *arXiv preprint arXiv:1903.11437* (2019).

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

[5] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).

[6] Currey, A., Miceli-Barone, A. V., and Heafield, K. Copied monolingual data improves low-resource neural machine translation. In *Proceedings of the Second Conference on Machine Translation* (2017), pp. 148–156.

[7] Dabre, R., Nakagawa, T., and Kazawa, H. An empirical study of language relatedness for transfer learning in neural machine translation. In *Proceedings of the 31st Pacific Asia Conference on Language, Information and Computation* (2017), pp. 282–286.

[8] Escribe, M. Human evaluation of neural machine translation: The case of deep learning. In *Proceedings of the Human-Informed Translation and Interpreting Technology Workshop (HiT-IT 2019)* (2019), pp. 36–46.

[9] Fadaee, M., Bisazza, A., and Monz, C. Data augmentation for low-resource neural machine translation. *arXiv preprint arXiv:1705.00440* (2017).

[10] Garg, A., and Agarwal, M. Machine translation: A literature review. *arXiv preprint arXiv:1901.01122* (2018).

[11] Hoang, V. C. D., Koehn, P., Haffari, G., and Cohn, T. Iterative back-translation for neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation* (2018), pp. 18–24.

[12] Hutchins, J. Machine translation: A concise history. *Computer aided translation: Theory and practice 13*, 29-70 (2007), 11.

[13] Indurthi, S. R., Chung, I., and Kim, S. Look harder: A neural machine translation model with hard attention. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (2019), pp. 3037–3043.

[14] Jurafsky, D., and H. Martin, J. Speech and language processing, 2021.

[15] Keet, C. M., and Khumalo, L. Grammar rules for the isizulu complex verb. *Southern African Linguistics and Applied Language Studies 35*, 2 (2017), 183–200.

[16] Koehn, P. *Statistical machine translation.* Cambridge University Press, 2009.

[17] Koehn, P., and Knowles, R. Six challenges for neural machine translation. *arXiv preprint arXiv:1706.03872* (2017).

[18] Lopez, A. Statistical machine translation. *ACM Computing Surveys (CSUR) 40*, 3 (2008), 1–49.

[19] Nguyen, T. Q., and Chiang, D. Transfer learning across low-resource, related languages for neural machine translation. *arXiv preprint arXiv:1708.09803* (2017).

[20] Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics* (2002), pp. 311–318.

[21] Sennrich, R., Haddow, B., and Birch, A. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709* (2015).

[22] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. *arXiv preprint arXiv:1706.03762* (2017).

[23] Weaver, W. Translation. *Machine translation of languages 14*, 15-23 (1955), 10.

[24] Zerbian, S. A first approach to information structuring in xitsonga/xichangana. *Research in African Languages and Linguistics 7*, 2005-2006 (2007), 1–22.

[25] Zhang, J., and Zong, C. Exploiting source-side monolingual data in neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* (2016), pp. 1535–1545.

[26] Zoph, B., Yuret, D., May, J., and Knight, K. Transfer learning for low-resource neural machine translation. *arXiv preprint arXiv:1604.02201* (2016).