Review of Machine Translation Techniques for Low Resource Languages with a focus on Nguni Languages

Fezeka Nzama
University of Cape Town
Cape Town, Western Cape, South Africa
NZMFEZ001@myuct.ac.za

Abstract

In this paper a survey of the state of machine translation (MT) is conducted with a focus on neural machine translation (NMT) and statistical machine translation (SMT) for the low resource language setting. As such an overview of both the Transformer variant of NMT and phrase-based SMT are detailed, as they are most powerful machine translation models for the respective machine translation methods. A case shall be presented for the use of SMT in the low resource setting, despite the gains derived from NMT in comparison with SMT in the high resource setting due to the significantly stronger results observed for SMT as opposed to NMT when trained on minial corpora. Additionally, various methods for improving low resource MT are presented, including modifying both NMT and SMT models, using pivot languages for translations and methods for increasing available parallel corpora. Additionally, this paper will provide an overview of the existing research regarding machine translation for the morphologically rich Nguni language group of Southern Africa in particular, and the difficulties that exist therein.

CCS Concepts

• **Computing methodologies** → Neural networks; **Neural networks**; **Machine translation**;

Keywords

neural machine translation, statistical machine translation, low resource languages, transformers, attention

1 Introduction

In a globalising world, interlingual communication has become of increasing importance to ensuring economic inclusion and allowing individuals from various parts of the world to work together efficiently. Thus, whilst machine translation is not new it has become increasingly more relevant as a means to facilitate this interlingual communication.

Machine translation refers to the use of a computer system to translate some corpora from a source language eg. English to a target language eg. IsiZulu [12]. Great strides have been made in this discipline using neural network architectures to make translation more accurate. However, much of the work in this field has been limited to high resource languages.[7, 10, 14, 17, 21]. High resource languages are those languages for which large amounts of text available to train models. The same cannot be said of low resource languages. This is the case when analysing machine translation for the Nguni language group of South Africa. The Nguni language group, including isiZulu and isiXhosa, constitutes the most widely spoken language group in South Africa, with 41.1% of South Africans speaking isiZulu or IsiXhosa as a home language [1]. However, despite their prevalence in everyday South Africa, Nguni language resources are still limited online, and as such parallel corpora for Nguni languages is minimal, limiting research into machine translation for this context and access to the majority of knowledge available on the web.

2 Evaluation Methods

Translations are evaluated along two main axes: adequacy and fluency. Adequacy refers to the level at which a translation is able to capture the meaning of a source sentence. This includes the ability to accurately capture tone. Fluency refers to readability of the translation in the target language. This includes being grammatically correct, clear and natural [22]. There are two main classes of evaluators used to assess translation quality, namely human evaluation, and automatic evaluation.

2.1 Human Evaluation

Human evaluation provides the most accurate evaluation of translations. This method makes use of people to evaluate translations along the dimensions of adequacy and fluency by either rating translations out of some range or ranking a set of translations in order from best to worst [20, 22]. This method whilst providing the most accurate evaluation, can be time consuming and expensive.

2.2 Automatic Evaluation

BLEU

BiLingual Evalution Understudy or BLEU is the most widely used evaluation metric for machine translation and makes use of a comparison between the machine translation system's output and, some human generated reference translation [9]. BLEU scores are evaluated along 3 main factors: (i) the translation length in comparison with the reference length, (ii) the words used in the translation in comparison with those used in the reference translation and (iii) The worder order in the translation in comparison with the reference translation. These comparisons are done by comparing the n-grams of the translations and the reference translations [9]. For a corpus, the BLEU algorithm counts the number of matching n-grams and returns as a score for the model a weighted average. Scores range from 0 for the lowest quality translation models, and 1 for highest quality or perfect translation models. Whilst BLEU is useful, it does fail to evaluate coherence in a document and does poorly when evaluating very different kinds of systems eg, Human-aided translation versus SMT [22].

NIST

The NIST metric, like the Bleu metric, makes use of n-gram comparisons between a reference translations and machine translations to determine the quality of a model. However, the NIST score differentiates itself from BLEU by scoring rarer segments higher weights. This aim here is to account for diversity in informational of translated texts [20].

Embedding-Based Methods

These methods try to improve on the BLEU metric by allowing translations that make use of synonyms [22]. Thus where the BLEU metric expects that a machine translation matches a reference translation exactly, embedding based methods allow for differences between the machine translation and reference translation arising from synonym use. These metrics do this by collecting reference translations and candidate machine translations which have been previously human rated for their quality in comparison to the reference translation and using these as a basis for new machine translations. Alternatively, where human labelling isn't available quality is determined based on the similarity of the reference and machine translation embeddings. Based on the earlier METEOR metric, some examples of embedding based methods include BLEURT and BERTScore [20, 22].

3 Training Data in the low resource setting

The data required to train a machine translation system is referred to as parallel corpora. This is text which can be found in two or more languages, whose sentences can be aligned as a source and translation pair. While online repositories of parallel corpora do exist, for low resource languages these databases either do not exist, offer small amounts of parallel corpora or have low quality translation pairs. [4, 9, 11, 18]. A number of methods have been proposed to overcome this challenge.

One method suggests scrubbing the internet for webpages which are translations of one another. Using the structural translation recognition acquiring natural data or STRAND architecture, a tool may be implemented to find pages that may be translations of each other. Thereafter generating candidate parallel sentences and filtering out non-translation pairs [19]

Another method suggests the use of a pivot language. This involves translating some source language S to an intermediary language I and from that language I to the target language T [10]. An obvious constraint of this method is that for the pivot language to be effective a large number of parallel corpora between language I and S as well as language I and T must exist. Moreover, a weak translation to the pivot language from the source or from the pivot language to the target language will negatively affect the translation as whole, whilst doing multiple translations increases the probability of translation errors arising [20].

Imankulova et. al suggest the use of a filtered pseudo-parallel data set as training data [9]. Creating this data involves using neural machine translation techniques to do a translation from a monolingual corpora in the target language to the source language, thus creating a synthetic source sentence. Thereafter, a translation from this source sentence into the target sentence is done to create a synthetic target sentence. The original target sentence and the synthetic target sentence are then compared. Where similarity is found to meet some threshold, the target-synthetic source sentence pair is retained, creating more parallel data [9]. This results in improved translation quality for low resource languages.

4 Neural Machine Translation(NMT)

Neural Machine Translation or NMT is the current state of the art in machine translation. This method makes use of the neural network architecture, taking in a set of inputs to predicts outputs [13, 24].

4.1 Encoder Decoder Approach

The most common approach to Neural Machine Translation is the encoder-decoder model. The vanilla architecture encoder takes in an input sequence in the source language and outputs a context vector, representing the essence of the original input. This context is then used as an input to the decoder, which generates an output sequence in the target language of translation taking into account history of the output sequence already seen by the decoder. [13, 24]. This is incapsulated by the conditional probability below:

$$p(y|x) = \prod_{t=1}^{T} p(y_t|y_{1:t-1}, x_{1:S}) \tag{1}$$

The goal of this encoder-decoder mechanism is to maximise the conditional log-likelihood

$$\max_{\theta} \frac{1}{|x|} \sum_{(x,y)\in D} \log p_\theta(y|x) \tag{2}$$

with D representing the set of paired training sentences and θ the set of parameters to be learned [3, 13, 23].

4.2 Attention

Modern architectures use an attention mechanism to calculate this context vector. This attention mechanism uses all the input word representations and previous decoder hidden state to generate a context vector for the next word to be decoded by the network. The attention mechanism aids in computing a context vector that is the weighted sum of all vectors representing the input words in the encoder. These weights assigned to each word differ depending on the relevance of an input word to the current token being generated. This leads to the generation of a dynamic context vector that seeks to exploit the strength of the association between the decoder state and each of the input words to make an accurate translation.[13, 23, 24]. This attention mechanism than can be expressed mathematically, using the scaled dot-product method, as follows:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\frac{1}{\sqrt{d_k}}}\right)V \tag{3}$$

where Q is a matrix of query parameters which correspond with the word for which the attention is being calculated, K is a matrix of key parameters which represent the various word embeddings, and V is a matrix of values also representing word embeddings.

The multiplication between Q and K yields an alignment score which is scaled down by a factor of $\frac{1}{\sqrt{d_k}}$. This value is than used as a weighting on V producing a weighted context vector. Whilst additive attention and dot product attention are both commonly used, dot product attention is faster and more space efficient and thus preferred [13, 23].
Taking into account a dynamic context vector, Bahdanau et al. [3] modifies the encoder-decoder conditional probability as follows:

$$p(y|x) = \prod_{t=1}^{T} p(y_t|y_{1:t-1}, x_{1:S}) \tag{4}$$

$$= \prod_{t=1}^{T} g(y_t|y_{1:t-1}, s_t, c(x)) \tag{5}$$

Where g is a nonlinear function resulting in a probability of y_t and s_t is the hidden state of the network. This model can be implemented using recurrent neural networks, long short-term memory networks and gated recurrent neural networks. However, transformer-based implementations are currently the preferred form of NMT [3, 23].

Self-Attention

Self-attention is a variant of the attention mechanism in which all the keys, queries and values come from the same input. The aim of the self-attention mechanism is to relate some input sequence with elements of itself. For example, given the following sentence "Jess likes her red hat", when calculating the self-attention for the query word "hat" this mechanism would seek to describe how strongly related with each word individually the word "hat" is. This contextualises the word within the sentence. Thus, an expected output might result in a higher weighting for the word "red" as relating to the hat [13, 20, 23].

Multi-Head Attention

Multi head attention uses multiple layers of attention with, projecting each query and key-value pair linearly multiple times. Once the independent attention outputs are found they concatenated and linearly transformed into some expected vector representation. This allows for the model to learn from different representational subspaces at different positions simultaneously.[3, 23].

4.3 Transformers

Transformers have become the state-of-the-art NMT method as they are quicker to train due to their parallelisability and outperform their predecessors due to their ability to offer an infinite window size where the model can draw on all the previous words in the model, as well as the input representations to derive the next output for the decoder. [13, 23]. The multi-head attention model is key to transformer functioning, whilst transformers also use attention in a number of ways [3] . Self-attention mechanisms are included in the encoder and decoder, allowing for self-contextualisation of words within each sub-network. Thereafter, a general attention mechanism is used to link each output position in the decoder to all input sequence positions, contextualising the output in terms of the input. [23]. This allows the model to learn from the various parts of the model individually, eliminates the need for recurrence within the model.

4.4 NMT for low resource and Nguni languages

Whilst NMT outperforms SMT when trained on large data sets, Koehn et al. found it to perform substantially worse than SMT when trained with corpus of a few million or less words using an attention-based encoder-decoder network [14]. Another alternative approach suggests the use of a universal neural machine, which is trained on a variety of languages, to avoid overfitting to limited data, whilst increasing the available parallel corpora for training purposes [21]. Where the languages a similar, such as Nguni languages, with shared surface forms for words and sentence structure, this method can yield substantial translation quality improvements. In addition to parallel corpora size, morphology also seems to play a part in NMT, with agglutinative languages, such as Nguni languages, showing less favourable results than Afrikaans, which is not agglutinative, when translated from English, despite have less training data [17].

5 Statistical Machine Translation(SMT)

Statistical machine translation or SMT grew from the 1990s as the internet facilitated access to large quantities of written text from which statistical methods could learn to perform translations [20]. The most widely used form of SMT is phrase based SMT that seeks to perform machine translations at a phrase level. This method outperforms all other SMT methods[**?**]The fundamental idea on which SMT is built is that given a list of parallel corpora and a particular source sentence S, then there exists in the list of translation sentences one translation that has the highest probability of having been derived from the source sentence S. Mathematically, this is expressed as the conditional probability Pr(T|S) [20]. Using Bayes Theorem this conditional probability can be expressed in terms of Pr(S|T) using the following formula:

$$Pr(T|S) = \frac{Pr(T)Pr(S|T)}{Pr(S)} \tag{6}$$

Where the Pr(T) is referred to as the language model for the target language and Pr(S|T) is the conditional probability that S is the correct source sentence given the translation sentence T, referred to as the translation model. The aim here is to maximise Pr(T|S)[6]. As the denominator is not dependent on S, the formula can then be simplified to the following:

$$T' = argmax_T[Pr(T) * Pr(S|T)] \tag{7}$$

This model combining the language and translation models as shown above is referred to as the noisy-channel model[12].

5.1 Language Model

The language model is expressed in the Pr(T) and works to capture information about word ordering within a target language sentence. Using substantially large monolingual corpora for language modelling leads to higher translation quality than a model trained with

a smaller data set of monolingual corpora, given both models are trained on an equal size parallel corpora dataset[12]. The major two major subgroups of language modelling techniques are outlined below.

Back-Off N-grams

The most commonly used method for deriving this language model is the Back-Off N-gram model [24]. This model uses n-grams, where the 'n' refers to the probability that a group of words appears as the last n words in a sequence. N-1 words are kept in the memory of the model and used as the basis on which the conditional probability is calculated. For example given the sequence English phrase sequence, "Children go", and two possible words "cat" and "home", Pr(home|Children go) would likely be higher than Pr(cat| Children go) based simply on the grammar rules and convention of the languages, and thus the likelihood that the sentence "Children go home" appear more often in text than "Children go cat". This is example of an N-gram with N = 3 [20]. N-grams have proven to be highly effective where large sets of training data are available, whilst also being advantageous due to their simplicity. However, this method does suffer where there is limited training data, as a grammatically and semantically correct ordering of words could have a low probability of appearing in a sequence due to it being unseen in the training data. This can lead to inconsistency in translations. Additionally, the N-gram model is very memory intensive as it stores the probability of every n-gram in the training data [2].

Continuous-Space Models

Continuous-Space Language Models (CSLMs) provides an alternative method for language modelling that is also shown to lead to better BLEU scores for SMT than BNLMs when trained on the same size corpus [24]. This alternative method is most commonly implemented using neural networks. First proposed by Bengio et. al. in 2003, this method seeks to represent words as points in a vector space according to their features, such that the model takes into account larger contexts ie. longer sequences of words or sentences, as well as the grammatic and semantic similarity of words in a vocabulary [5**?**] . The idea is that given a sentence, eg. "The dog is outside", the word "cat" could sensible be a substitute for dog as the two words perform similar semantic and grammatical functions. As such, these words would be located near each other in the vector space. Thus, using a neural network, the probability of seeing the word "cat" in the sequence would be similar to that of seeing the word "dog" in the sequence. This generalisation allows for unseen sequences to carry similar probabilities to seen sequences [5**?**] . However, due to the computationally heavy nature of training these neural networks, CSLMs continue to be underused in SMT.

5.2 Translation Model

The central idea used in the translation model for phrase based SMT is that of associating a sequence of n words from the source language to another sequence of m words from the target sentence. As such a phrase in the source language can be linked to a phrase in the target language[20].

Word-Level Alignments and Word-Based SMT

The simplest approach to the translation problem is at a word level. Given a source and target sentence, this approach seeks to find a translation in the target language for each word in the source sentence[6, 12]. This is done by analysing a body of parallel corpora and developing a probability distribution for the occurrence of a target word, in relation with a source word. Thus, given some target word T, if in the body of parallel corpora on which the model is trained a particular source word S appears most often in parallel sentences to those containing the word T, then using a frequency distribution table some conditional probability Pr(S|T) can be determined which is maximises the likelihood of the words S and T being translations of each other [6**?**] . This mapping of target words to source words is referred to as word level alignment. An extension of this concept then allows for the alignments of source words to 0 or more words in the target language, due to the vocabulary differences existing between languages. This idea lends itself to the IBM Model 1, which is word-based translation model defined using the following formula

$$p(T, a|S) = \frac{\epsilon}{(l_T + 1)^{l_S}} \prod_{j=1}^{l_S} t(S_j|T_{a(j)}) \tag{8}$$

$\frac{\epsilon}{(l_T+1)^{l_S}}$ works as a normalisation factor ensuring that the sum of all possible probabilities for translation TT and alignment a is one. The second half of the formula is the product of all the word-level translation probabilities. This simple model can then be optimised using expectation maximization (EM) algorithm to compensate for missing data to try various word alignments against one another and improving the model by allowing it to learn from itself as the EM algorithm approaches convergence[12]. Whilst further improvements to this model have ben suggested, and result in more powerful word-based statistical machine translation, this method pales in comparison to phrase-based statistical machine translation.

Phrase-Level Alignments and Phrase-Based SMT

Phrase based SMT makes use of phrases as the basic translation, thus alignments are done at a phrase level as opposed to at a word level. A phrase is sequence of words with consistent word alignment. This means for corresponding phrases there are no word alignments that fall outside of the sequences. This is referred to as being consistent with word alignment [12]. This method decomposes the translation model as follows:

$$p(\bar{T}^{I_1}|\bar{S}^{I_1}) = \prod_{i=1}^{I} \phi(\bar{T}_i|\bar{S}_i) d(a_i - b_{i-1}) \tag{9}$$

With the $\phi(\bar{T}_i|\bar{S}_i)$ probability distribution modelling the phrase translation. $d(a_i - b_{i-1})$ models the reordering of the target sentences using a distance-based reordering model that defines the differences in start position for related phrases within a source-targes sentence pair [12, 16]. The translation process in this case

begins by splitting the sentence into phrase, translating each phrase and thereafter permuting these phrases into an order consistent with the target language. [12, 15, 16]. Extracted phrase pairs are stored in a phrase translation table and the translation probability of phrase pair $\phi(\bar{T}_i|\bar{S}_i)$ is estimated using the relative frequency:

$$\phi(\bar{T}|\bar{S}) = \frac{count(\bar{S}, \bar{T})}{\sum_{\bar{T}_i} count(\bar{S}, \bar{T}_i)} \tag{10}$$

which allows for one phrase to be matched with multiple phrases in a sentence pair.

The advantages of this phase-based statistical machine learning approach over word-based SMT are that phrase based SMT allows for many-many-many alignments of words, overcoming issues around vocabulary length differences. Additionally, the use of phrases communicates context for words better than the word-based method and this method potentially allows for the learning of longer phrases given a large corpora. [12, 15]. However, this method is a memory intensive process.

5.3 SMT for low resource and Nguni languages

Whilst the clearest method for improving SMT for low resource languages is to increase the available corpora for these languages. An alternative method investigated by researchers is the use of pre- and post-processing rules to the machine translation process, which was found to increase BLEU and NIST scores for the Setswana, Sesotho and Arabic [7, 8, 18]. Another method suggests the use of linguistic modules in conjunction with SMT to improve performance[12]. As with NMT, another suggested approach makes use of a shared high resource pivot language. However, as in the case for NMT, this approach suffers from increased probability of errors arising from multiple translations, whilst also being dependent on the existence of a large parallel corpora between the pivot language and both the source and target language [12].

6 Conclusions

Machine translation in the low resource setting faces many challenges. These challenges are excerbated, in the case of the Nguni languages group, by the morphologically rich nature of these languages and the stark differences existing between them and more highly resourced European languages such as English. However, methods to overcome this challenge such as pre- and post-processing inputs or outputs using syntax rules, creating hybrid machine translation models have been suggested and have shown some promise.

Whilst neural machine translation, and transformers in particular, have led to great advancements in machine translation for high resource languages, where there is limited parallel corpora, this method's performance is significantly inferior to that of phrase-based SMT. This suggests that whilst NMT is the current state-of-the-art, SMT may still provide particular benefits and merit more research attention in the low resource setting.

Methods for improving NMT in the low resource setting are seeing more attention from researchers. This has led to the development of universal NMT models aimed at exploiting language similarity, and related neural network based models for creating synthetic parallel corpora. These methods have been further bolstered by tactics suggesting the use of pivot languages in the translation process. Thus, it is clear that machine translation for low resource languages warrants further exploration as a sub-field of machine translation.

References

[1] South Africa's people | South African Government.

[2] Arisoy, E., Chen, S. F., Ramabhadran, B., and Sethy, A. Converting Neural Network Language Models into Back-off Language Models for Efficient Decoding in Automatic Speech Recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing 22*, 1 (Jan. 2014), 184–192.

[3] Bahdanau, D., Cho, K., and Bengio, Y. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv:1409.0473 [cs, stat]* (May 2016). arXiv: 1409.0473.

[4] Baltescu, P., and Blunsom, P. Pragmatic Neural Language Modelling in Machine Translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (Denver, Colorado, 2015), Association for Computational Linguistics, pp. 820–829.

[5] Bengio, Y., Ducharme, R., Vincent, P., and Jauvin, C. A Neural Probabilistic Language Model. *Journal of machine learning research 3*, 6 (Aug. 2003), 1137–1155.

[6] Brown, P., Cocke, J., Della Pietra, S., Della Pietra, V., Jelinek, F., Lafferty, J., Mercer, R., and Roossin, P. A Statistical Approach to Machine Translation. *Computational Linguistics 16*, 2 (June 1990), 79–85.

[7] Griesel, M., McKellar, C., and Prinsloo, D. Syntactic Reordering as Preprocessing Step in Statistical Machine Translation of English to Sesotho sa Leboa and Afrikaans. pp. 205–211.

[8] Gu, J., Hassan, H., Devlin, J., and Li, V. O. K. Universal Neural Machine Translation for Extremely Low Resource Languages. *arXiv:1802.05368 [cs]* (Apr. 2018). arXiv: 1802.05368.

[9] Imankulova, A., Sato, T., and Komachi, M. Filtered Pseudo-parallel Corpus Improves Low-resource Neural Machine Translation. *ACM Transactions on Asian and Low-Resource Language Information Processing 19*, 2 (Mar. 2020), 1–16.

[10] Karakanta, A., Dehdari, J., and van Genabith, J. Neural machine translation for low-resource languages without parallel corpora. *Machine Translation 32*, 1-2 (June 2018), 167–189.

[11] Keet, C. M., and Khumalo, L. Toward a knowledge-to-text controlled natural language of isiZulu. *Language Resources and Evaluation 51*, 1 (Mar. 2017), 131–157.

[12] Koehn, P. *Statistical machine translation*. Cambridge University Press, Cambridge ; New York, 2010. OCLC: ocn316824008.

[13] Koehn, P. *Neural Machine Translation*, 1 ed. Cambridge University Press, June 2020.

[14] Koehn, P., and Knowles, R. Six Challenges for Neural Machine Translation. *arXiv:1706.03872 [cs]* (June 2017). arXiv: 1706.03872.

[15] Koehn, P., Och, F. J., and Marcu, D. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - NAACL '03* (Edmonton, Canada, 2003), vol. 1, Association for Computational Linguistics, pp. 48–54.

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

[17] Martinus, L., and Abbott, J. Z. Benchmarking Neural Machine Translation for Southern African Languages. *arXiv:1906.10511 [cs, stat]* (June 2019). arXiv: 1906.10511 version: 1.

[18] Maučec, M. S., and Brest, J. Slavic languages in phrase-based statistical machine translation: a survey. *Artificial Intelligence Review 51*, 1 (Jan. 2019), 77–117.

[19] Nyoni, E., and Bassett, B. A. Low-Resource Neural Machine Translation for Southern African Languages. *arXiv:2104.00366 [cs]* (Apr. 2021). arXiv: 2104.00366.

[20] Poibeau, T. *Machine translation*. The MIT Press essential knowledge series. The MIT Press, Cambridge, Massachusetts, 2017.

[21] Resnik, P., and Smith, N. A. The Web as a Parallel Corpus. *Computational Linguistics 29*, 3 (Sept. 2003), 349–380.

[22] Shterionov, D., Superbo, R., Nagle, P., Casanellas, L., O'Dowd, T., and Way, A. Human versus automatic quality evaluation of NMT and PBSMT. *Machine Translation 32*, 3 (Sept. 2018), 217–235.

[23] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention Is All You Need. *arXiv:1706.03762 [cs]* (Dec. 2017). arXiv: 1706.03762.

[24] Wang, R., Utiyama, M., Goto, I., Sumita, E., Zhao, H., and Lu, B.-L. Converting Continuous-Space Language Models into N -gram Language Models with Efficient Bilingual Pruning for Statistical Machine Translation. *ACM Transactions on Asian and Low-Resource Language Information Processing 15*, 3 (Mar. 2016), 1–26.