Improving Transfer Learning for Low-Resource Neural Machine Translation by Introducing Language Similarity

Okke van der Wal
Student number: 2006017

Thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Cognitive Science & Artificial Intelligence
Department of Cognitive Science & Artificial Intelligence
School of Humanities and Digital Sciences
Tilburg University

Thesis supervisor:

Dr. Menno van Zaanen

Tilburg University
School of Humanities and Digital Sciences
Department of Cognitive Science & Artificial Intelligence

Tilburg, The Netherlands
May 2019

Abstract

Neural machine translation systems succeed in data-intensive applications, but they lack the ability to learn from a limited number of training examples. While they often outperform traditional machine translation systems, this is not the case for *low-resource languages* for which adequate bilingual data is not available. To tackle this problem, transfer learning has been proven to be an effective approach, where one uses knowledge from a *source task* to improve the performance on a *target task*. Yet, one aspect that has largely remained unexplored is the similarity between these two tasks and the extent to which that similarity may influence the effectiveness of this approach. This is particularly remarkable since, intuitively, we may assume that tasks that are more similar to each other, are also more suited for utilizing a transfer learning method. Nonetheless, our research is, to our knowledge, the first to demonstrate this influence. In our task on neural machine translation for South-African languages, we demonstrate that by using the Levensthein distance as the similarity measure, language pairs in our *target task* that are closely related to the language pair in our *source task* yield an improvement in BLEU score of up to 4.9 BLEU.

Contents

List of Figures

List of Tables

1 Introduction

Language translation is a key aspect in understanding and keeping the world connected. While English is the most popular second language to learn in most countries (Hartshorne et al., 2018), most people who prefer to speak in their native languages, and some countries and regions do not have the ability or opportunity to speak English. This therefore increases the need for adequate translation tools. In any translation task, the meaning of a text in the source language must be transferred to its meaning in the target language's translation. Historically, the only way to translate languages was to use a human translator with sufficient knowledge of multiple languages. However, in recent years, there has been a surge in the use of machine translation (Hutchins and Somers, 1992). Even though human translators are still in use, the introduction of machine translation is increasingly replacing the need for human translators. In machine translation, one uses software to translate text or speech from one language to another. Historically in machine translation, two main approaches emerged: rule-based machine translation (RBMT) and statistical machine translation (SMT). In recent years, neural machine translation (NMT) emerged, an approach that utilizes deep learning techniques. Neural machine translation is demonstrating that it is a strong competitor to traditional machine translation methods, but the performance lags behind other statistical methods on languages that have little data available (Koehn and Knowles, 2017). Languages where little data is available are called *low-resource languages*, while languages where large amounts of data are available are called *high-resource languages*.

Currently, organizations are looking to replace rule-based machine translation (RBMT) systems and statistical machine translation (SMT) systems with neural machine translation (NMT) systems (Bentivogli et al., 2016). However, before an organization can make such a decision, it needs to be certain that enough data are available. The introduction of transfer learning in machine translation is an approach that may solve the issue of data scarcity. In this approach, one first trains a high- resource language pair (the parent model), then transfer some of the learned knowledge, or more concretely model weights or model parameters, to the low-resource pair (the child model). By doing this, less training data is required, possibly solving the issue for languages where limited amounts of data are available. It is expected that in the long-term, most organizations will replace their traditional machine translation systems with neural machine translation systems.

Nonetheless, the issue of data scarcity is a key issue in when that process of replacement should commence (Bentivogli et al., 2016).

1.1 Project motivation

The Centre for Text Technology (CTexT) is a South African research and development center at the Potchefstroom Campus of the North-West University. Their main interests are in research on human language technology and in developing language technology products for the South African languages. Their main focus is on resource-scarce languages, especially South African languages for which little data exists. One of their current projects is a feasibility study of neural machine translation for low-resource languages and they need to make a recommendation if their client should migrate the existing systems and services to NMT based systems or stay with statistical machine translation (SMT). The main task in our research, therefore, is to demonstrate whether a transfer learning approach may be a feasible approach to solve the issue of data scarcity for these languages. Moreover, if that is the case, the organization may be confident enough to replace their current system.

1.2 Research question

We stated that it is expected that in the long-term, most organizations will eventually replace their traditional machine translation systems with neural machine translation systems as the results in various experiments demonstrated that neural machine translation systems outperformed traditional machine translation systems (Bentivogli et al., 2016). Nonetheless, the issue of data scarcity is a key issue in the success of neural machine translation (Koehn and Knowles, 2017). We also observed that the introduction of transfer learning in machine translation is an approach that may solve this issue of data scarcity (Torrey and Shavlik, 2010). Besides, intuitively, we may assume that languages that are more similar to each other, are also more suited for transfer learning in machine translation.

Our hypothesis is therefore that languages that have a higher language similarity in comparison to the parent model, will also have a higher machine translation score. In this research we are therefore interested in answering the following main research question:

> To which extent does language similarity influence the performance in transfer learning for low-resource neural machine translation?

1.3 Thesis outline

In section 2, we provide an overview of the existing literature that is relevant in order to understand the recent developments in the domain of this thesis. We discuss the concept of machine translation and we explain how in recent years neural machine translation has made its introduction. We also introduce the idea behind transfer learning and discuss how this may influence machine translation.

We subsequently explain how we performed our experiments in section 3, in order to ensure that the results of this research may be replicated. We then describe our results in section 4, where we use visualizations to better portray our results. We then discuss these results in section 5, where we explain the implications of our results and indicate what this may signify within the domain of machine translation. Finally, we provide an overview of our conclusions in section 6.

2 Literature review

In this section we review the currently existing literature and provide an overview of the recent developments in the domain of this thesis. We first describe the idea behind machine translation in subsection 2.1 after which we describe the recent developments in neural machine translation in subsection 2.2. We subsequently explain the idea behind transfer learning in subsection 2.3 and how it may improve the results of neural machine translation. Finally we describe how language similarity can be measured and how it can be applied to transfer learning in subsection 2.4.

2.1 Machine translation

Machine translation (MT) is a field of research within the natural language processing domain and can be simply defined as the automated translation of natural languages (Sennrich et al., 2016). Moreover, one uses computer software to automatically translate a text from one natural language (such as English) to another (such as Spanish). In general, to process any translation, human or automated, the meaning of a text in the original (source) language must be restored in the target language. Machine translation is originated in the 1950s and has since developed a number of different approaches. In general, two dominant approaches can be distinguished: rule-based machine translation (RBMT) and statistical machine translation (SMT) (Brown et al., 1990).

Rule-based machine translation (RBMT) relies on built-in linguistic rules and a bilingual dictionary for each language pair. These sets of rules then transfer the grammatical structure of the source language to the target language. Statistical machine translation, on the other hand, utilizes statistical translation models whose parameters are derived from the analysis of monolingual and bilingual data sets. Moreover, statistical MT implements predictive algorithms to teach a machine how to translate text. These models are created, or learned, from parallel bilingual text data and are used to create the most likely output, based on different bilingual examples. Statistical MT has different subgroups, including word-based (Koehn et al., 2003), phrase-based (Zens et al., 2002), syntax-based and hierarchical phrase-based (Brown et al., 1990). The advantage of using SMT is that it has demonstrate to provide reliable translation quality when large corpora are available. One limitation is that this system needs bilingual material to work from, and it

can be hard to find these data for low-resource languages, especially since it is estimated that approximately two million words are required to train a statistical MT model (Jurasfsky and Martin, 2000).

In general, the quality of machine translation systems is often measured using the BLEU score. The Bilingual Evaluation Understudy Score (BLEU) is an evaluation metric for evaluating a generated sentence to a reference sentence. The BLEU score was first proposed by Papineni et al. (2002) and is often used for evaluating machine translation systems. Formally, the BLEU score P_n can be computed with the formula below by comparing n-grams of the candidate translation S with the n-grams of the source translation C and count the number of matches.

$$p_n = \frac{\sum_{S \in C} \sum_{ngram \in S} Countmatched(ngram)}{\sum_{S \in C} \sum_{ngram \in S} Count(ngram)} \tag{1}$$

2.2 Neural machine translation

In recent years, neural networks have gained increasing popularity in the machine learning domain. This is based on the model of neural networks in the human brain, where information is sent to different layers in the neural network to be processed before the network produces an output. In machine translation, neural machine translation (NMT) systems utilize these deep learning techniques to translate text based on large data sets. Moreover, a neural machine translation system is a neural network that directly models the conditional probability of translating a source sentence to a target sentence, similar to statistical machine translation systems.

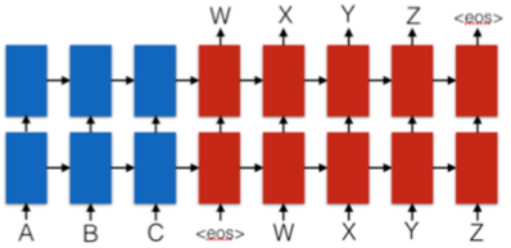
Written formally, it aims to directly model the conditional probability $p(y|x)$ of translating a source sentence, $x_1, \ldots, x_n$ to a target sentence, $y_1, \ldots, y_n$.

Yet, one key difference is that neural machine translation systems require only a fraction of the memory needed by traditional statistical machine translation (SMT) models. This results in faster translations than the statistical method and has the ability to create higher quality output as consistently demonstrated (Sutskever et al., 2014) (Srivastava et al., 2014).

One of the most popular systems used in NMT is the sequence-to-sequence (seq2seq) approach, also known as the encoder-decoder architecture. This system was developed by Ilya Sutskever (Sutskever et al., 2014) and was one of the first neural machine translation systems to outperform a baseline

statistical machine learning model on a large translation task. This encoder-decoder framework is visualized in Figure 1. We can see here that, in this framework, a source sentence C B A (presented in reverse order as A B C) is translated into a target sentence W X Y Z.

Figure 1: The encoder-decoder framework for neural machine translation (NMT) translating a source sequence A B C into a target sequence W X Y Z. Here,< eos > marks the end of a sentence (Sutskever et al., 2014)



The framework consists of two systems, indicated by the two different colors: the encoder and the decoder, hence the name of the framework. The basic idea is that the encoder takes the sequence of input words, converts it to an intermediate representation, then passes that representation to the decoder which produces the output sequence. The goal is to maximize the likelihood of generating the correct output sequence, thus, at each step, the decoder is rewarded for predicting the next word correctly and penalized for predicting the next word incorrectly.

Formally, the encoder computes a representation S of all the input words in source sentence $x_1, \ldots, x_n$. Based on that source representation of source sentence, the decoder generates a translation $y_1, \ldots, y_n$, one target word at a time, and hence, decomposes the log conditional probability $\log p(y|x)$ of generating this translation as:

$$\log p(y|x) = \sum_{t=1}^{m} \log p\left(y_t|y_{<t}, s\right) \tag{2}$$

What is also interesting to note in Figure 1 is that an end-of-sequence $<$ eos $>$ token was added to the end of sequences during training. This to signal the end of the sentence. This allows the model to translate variable length output sequences, one of the advantages of the framework. Moreover,

the encoder first reads an input sequence in entirety until it reaches this < eos > token, after which it encodes the sequence to a fixed-length vector representation. This encoder includes a combination of several recurrent units, where each unit takes a single element of the input sequence, collects information for that element and propagates it forward, as demonstrated in the figure. The decoder network then uses this final vector representation to output words for the translated sentence until the < eos > token is reached.

Now, we describe the architecture of the framework in more detail, specifically which machine learning algorithms are often used for both the encoder and the decoder. In general, the machine learning model of choice in the decoder is a recurrent neural network (RNN) architecture. RNNs work by processing language sequentially in a left-to-right or right-to-left fashion, which makes it most suitable in this task. As the RNN reads only one word at a time, it is required to perform multiple steps to make decisions that depend on words far away from each other, also called long-term dependencies. Even though this architecture has since become an effective and standard approach for neural machine translation, it has also been proven to be difficult to train the RNN networks due to these resulting long-term dependencies (Chorowski et al., 2015). Moreover, while one of the advantages of RNNs is that they are able to connect information from a previous position in a sequence to the current position in the sequence, the further away this previous position, the more difficult it is for the RNN to draw the right conclusions. To solve this, both Sutskever et al. (2014) and Luong et al. (2015) combined multiple layers of an RNN with a Long Short-Term Memory (LSTM) hidden unit for both the encoder and the decoder. The Long Short-Term Memory (LSTM) is an algorithm known to learn problems with long range temporal dependencies and has proven to solve the aforementioned issue. Essentially, Long Short Term Memory networks (LSTMs) are a type of RNN, that was specifically designed by Hochreiter and Schmidhuber (1997) to better learn these long-term dependencies

Overall, the main advantages of the encoder-decoder framework are the ability to train a single end-to-end model directly on source and target sentences as we described before. This makes for faster translations than statistical machine translation and has demonstrated that it frequently outperforms statistical machine translation results(Sutskever et al., 2014). However, deep neural networks still require fixed-sized input and output vectors, and this is one of the reasons they were not always able to outperform statistical machine translation models. Moreover, previous studies (Cho et al., 2014)

(Kalchbrenner et al., 2014) (Sutskever et al., 2014) find that translation quality drops significantly when a NMT system translates long sentences. The main reason is that, for longer sentences, the fixed-size vector representations of source sentences by cannot identify all the cues for the decoder to generate appropriate translations, as it needs to compress all the information of a source sentence into a fixed-length vector representation.

The concept of an attention mechanism is a method that might solve this. Recently, this concept has gained popularity in the machine learning domain as this allows machine learning models to learn alignments between different data types, for example between speech frames and text in speech recognition tasks (Chorowski et al., 2015). In applying this method in an NMT system, such an attention mechanism only focuses on the relevant parts of the source sentence while translating, instead of only relying on a fixed vector representation of the full sentence. In other words, it only focuses on the position in a sentence where the most relevant information is concentrated. Therefore, in contrast to the standard encoder–decoder framework, an attention mechanism does not attempt to encode the input sentence into a single fixed-length vector. Instead, it encodes the input sentence into a sequence of vectors and chooses a subset of these vectors continuously while decoding the translation sentence. The key difference in the attention mechanism is the attention layer. It works as follows. An input sequence of the attention layer is called a query and for a query, the attention layer returns the output based on its memory, which is a set of key-value pairs. Formally, given a query $\mathbf{q} \in \mathbb{R}^{d_q}$ and the memory which contains n key-value pairs, $(\mathbf{k}_1, \mathbf{v}_1), \ldots, (\mathbf{k}_n, \mathbf{v}_n)$ with $\mathbf{k}_i \in \mathbb{R}^{d_k}, \mathbf{v}_i \in \mathbb{R}^{d_v}$ and the goal is that the attention layer finally returns an output $\mathbf{o} \in \mathbb{R}^{d_v}$.

To compute this output, we need a scoring function α, which measures the similarity between the query and a key. Then we may compute all n scores $a_1, \ldots, a_n$ by

$$a_i = \alpha(\mathbf{q}, \mathbf{k}_i) \tag{3}$$

Next we use a softmax function to obtain the attention weights $b_1, \ldots, b_n$

$$b_1, \ldots, b_n = softmax(a_1, \ldots, a_n) \tag{4}$$

The output o of an attention layer is finally a weight sum of these values

$$\mathbf{o} = \sum_{i=1}^{n} b_i \mathbf{v}_i \tag{5}$$

Concretely, an attention mechanism provides the decoder access to all the hidden states of the encoder that were stored in memory. The attention mechanism then asks the decoder to choose which hidden states to use and which to ignore by weighting the hidden states. The decoder is then passed a weighted sum of hidden states to use to predict the next word.in most cases attention mechanisms were used in conjunction with a recurrent neural network (RNN). This class models of is often called *sequence-to-sequence with attention*, but it has limitations.

Moreover, if only a single attention weighted sum of the values was computed, it would be more difficult to capture various different aspects of the input. To solve this problem, the *Transformer* architecture, first introduced by Vaswani et al. (2017) extends this idea of an attention mechanism by using so-called *multi-head attention blocks*. One block, also referred to as *head*, computes multiple attention weighted sums instead of a single weighted sum vector as in traditional attention mechanisms. Concretely, in their research, Vaswani et al. (2017) replaced the recurrent layers in their encoder-decoder architecture with these*multi-head attention blocks*. To better understand the differences between this approach and the *seq2seq with attention* approach, Figure 2 below provides a visual comparison.

Figure 2 demonstrates that these two models are rather similar, overall. Yet, there are three key differences. First, a recurrent layer in *seq2seq with attention* is replaced with a *head* or an *attention block*. This block contains a self-attention layer and a network with two dense layers for the encoder. Second, the state of the encoder is forwarded to each transformer layer in the decoder, instead of using as an additional input of the first recurrent layer. Finally, a positional encoding layer is implemented in the *Transformer* to add sequential information into each sequence item, since the self-attention layer cannot distinguish the input order of a particular sequence.

The *Transformer* approach has since its introduction demonstrated to deliver a better performance in neural machine translation tasks, compared to the *seq2seq with attention* approach and is now the default choice of architecture in neural machine translation and has been shown effective in large data scenarios (Lakew et al., 2018) (Ott et al., 2018).

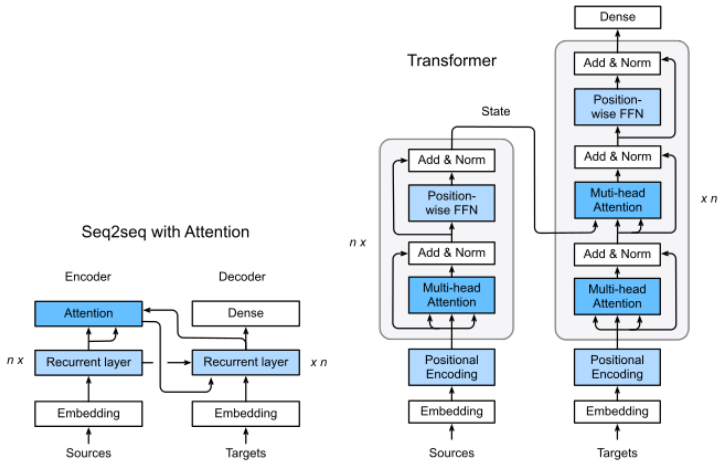


Figure 2: The *Transformer* architecture (Vaswani et al., 2017) on the right, in comparison to the *seq2seq model with attention* architecture on the left (Luong et al., 2015).

However, it is often less effective for low-resource languages (Kocmi and Bojar, 2018). Moreover, NMT systems with attention are outperformed by standard string-to-tree statistical MT (SBMT) when translating low-resource languages into English as demonstrated by Luong, Pham and Manning (Luong et al., 2015). Especially for NMT models with attention, or *Transformer* architectures that have a large number of parameters, as these are particularly sensitive to the quality of data (Vaswani et al., 2017).

To solve the issue of data scarcity in low-resource neural machine translation, Zoph et al. (2016) was one of the first to implement a transfer learning method. Their main idea was to first train a high- resource language pair *(the parent model)*, then transfer some of the learned parameters to the low-resource pair *(the child model)*. They demonstrated that they could improve baseline NMT models by an average of 5.6 BLEU on four low-resource language pairs. This idea of implementing transfer learning is an approach that has since received little attention, but it is an idea that we would like to explore further. We therefore first present an explanation of the transfer learning approach.

2.3 Transfer learning

The idea behind transfer learning is that one uses knowledge from a a *source task* in a certain *source domain* to improve the performance on a *target task* in a *target domain*. In transfer learning, essentially, one stores the knowledge that is gained in the *source task* and applies it to the *target task*, typically reducing the amount of required training data. This approach is visualized in Figure 3. Typically, the *domain* refers to the nature of the task and specifically the type of input data (Torrey and Shavlik, 2010). If both tasks are part of the same domain as the tasks are similar, one may simply refer to a *source task* and a *target task* and assume that they are part of the same domain.

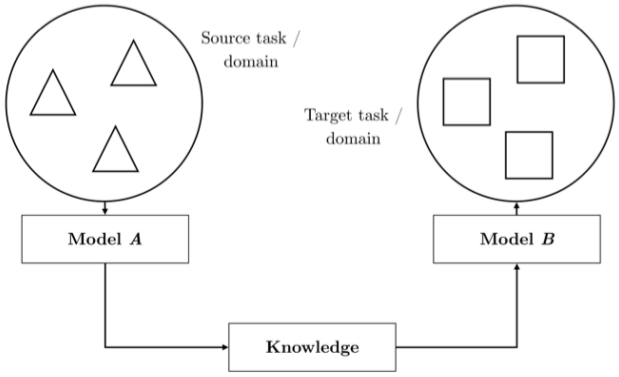


Figure 3: The transfer learning framework (Ruder, 2019)

We already observed this transfer learning approach in subsection 2.2 in the research by Zoph et al. (2016), in which they identified a *parent model* which may be considered to be the *source task* and they transferred the knowledge learned to the *child model*, which was their *target task*. As both tasks were similar, they were thus part of the same *domain*.

Before we provide an explanation of how a transfer learning approach can be implemented within the machine translation domain, we first provide a formal definition of transfer learning following the notation of Pan and Yang (2010), to mathematically explain the transfer learning approach. As aforementioned, transfer learning revolves around the concepts of a *domain* and a *task*. A domain D consists of a feature space $\mathcal{X}$ and a marginal probability distribution $P(X)$ over the feature space, where $X = \{x_1, \cdots, x_n\} \in \mathcal{X}$.

Given a domain $\mathcal{D} = \{X, P(X)\}$, a task T consists of a label space Y, a prior distribution $P(Y)$, and a conditional probability distribution $P(Y|X)$ that is typically learned from the training data consisting of pairs $x_i \in X$ and $y_i \in \mathcal{Y}$.

Then, given a source domain $\mathcal{D}_S$, a corresponding source task $\mathcal{T}_S$, as well as a target domain $\mathcal{D}_T$ and a target task $\mathcal{T}_T$, the objective of transfer learning now is to learn the target conditional probability distribution $P_\eta\left(Y_T|X_T\right)$ in D_T with the information gained from $\mathcal{D}_S$ and $\mathcal{T}_S$ where $\mathcal{D}_S \neq \mathcal{D}_T$ or $\mathcal{T}_S \neq \mathcal{T}_T$. This mathematical definition demonstrates that the knowledge of the *source task* that is transferred to the *target task* is essentially the prior distribution of the *source task*.

Recently, transfer learning has become a popular approach in the machine learning domain, since large data sets are often required to successfully train machine learning models (Torrey and Shavlik, 2010). Especially in tasks where not enough data is available, transfer learning has demonstrated that it can overcome this issue of data scarcity. For instance, in natural language processing, transfer learning methods have been successfully applied to speech recognition and sentiment analysis (Wang and Zheng, 2015).

There are a various approaches to transfer learning, but the main approach is the 'develop model approach'. In this approach, one trains a machine learning model on a different but related task, where there is an abundance of data. One then uses the pre-trained model, more specifically the parameters or weights of the model, to train the model for the task where few data is available. The relatedness between these different tasks if crucial in successfully applying transfer learning. One could also use an already pre-trained model, often called the 'pre-trained model approach'. In this approach, one uses an external pre-trained model, which is often publicly available. However, the first approach is often preferred, as one has more control over the relatedness between the different tasks (Torrey and Shavlik, 2010).

In the task of NMT, research by Zoph et al. (2016) implemented this transfer learning method and improved baseline NMT models by an average of 5.6 BLEU for four low-resource language pairs. They demonstrated the success of using transfer learning in NMT, but also suggested that there is still room for improvement in selecting parent languages that are more similar to child languages. Moreover, they stated that taking into account language similarity may strongly improve the generalization ability of a neural machine translation system. To illustrate this, in their research, they devised

a synthetic language to use as a child model, based on the original parent model and obtained a 4.3 BLEU improvement with an unrelated parent and obtained a 6.7 BLEU improvement with a 'closely related' parent. Based on these results, they concluded that the choice of parent model can have a strong impact on the performance of transfer learning models, and stated that choosing more suitable parents for the low- resource languages could significantly improve the results of a transfer learning approach within the neural machine translation domain.

Since, this idea has, to our knowledge, not been explored further and we agree with Zoph et al. (2016) and hypothesize that the language similarity between the source language and the target language indeed significantly influences the performance of this method, based on the results of their experiments. However, the issue arises how to measure this language similarity. In the research by Zoph et al. (2016), they devised a synthetic language, ensuring there was a certain degree of similarity with the parent language. Yet in this task, we focus on natural languages. In the literature we see that language similarity between official languages can be measured using a number of different methods. We therefore first present and compare different language similarity measures to see which metric is most suitable to our research.

2.4 Language similarity

How to measure language similarity is one of the main questions in the computational linguistics domain (Thompson et al., 2018). Various comparative methods have been implemented to identify relationships between languages, to determine language similarity (Eska et al., 1997). The main issue is that languages are complex since they differ in vocabulary, grammar, syntax and many other characteristics. This introduces a certain difficulty in establishing objective comparative measures between languages. Even if one intuitively knows for example, that English is closer to French than it is to Chinese, it may difficult to quantify this difference.

Often, subjective measures have generally been employed to determine the degree of similarity between different languages (Heeringa et al., 2013), due to the lack of available objective measures. These subjective measures often classify languages based on the region where the language is spoken. In our task we specifically focus on South African languages and an example of such a subjective measure applied to these languages is the Guthrie zone classification method. Most languages that are spoken on the African continent are namely classified as Bantu languages. These Bantu languages are a large family of languages spoken by the Bantu people throughout Sub-Saharan Africa. These are conventionally divided up into geographic zones first proposed by Guthrie (1971). These were assigned letters A–S and divided into decades (groups A10, A20, etc.); individual languages were assigned unit numbers (A11, A12, etc.), and dialects further divided (e.g. A11a). This coding system has since become the standard for identifying Bantu languages and it still is the only practical way to distinguish many ambiguously named languages on the African continent.

However, this subjective method is not well suited to determine how similar languages are to each other and is therefore not well suited to measure language similarity in our research. As we previously mentioned, with using this method, it may be difficult to quantify the difference between these languages and their respective Guthrie zones. To better quantify the distance between languages, more objective or quantifiable measures should be used instead. An example of such a measure is the Levensthein distance, as employed in the research by Heeringa et al. (2013). The Levenshtein distance is a metric for measuring the difference between two sequences.

In essence, the Levenshtein distance between two words is the minimum number of edits of characters in a word (insertions, deletions or substitutions)

necessary to change one word into the other. It was first introduced in 1995 as a tool for measuring linguistic distances between dialects (Kessler, 1995). In using the Levenshtein distance measure, the distance between two languages is equal to the average of a sample of Levenshtein distances of corresponding word pairs. It is estimated that a sample of approximately 50 words per language is sufficient to measure the distance.

Formally, the Levensthein distance d_{ij} between string $a = a_1 \dots a_n$ and string $b = b_1 \dots b_m$ is given by

$$d_{i0} = \sum_{k=1}^{i} w_{\text{del}}\left(b_k\right), \qquad for 1 \leq i \leq m \tag{6}$$

$$d_{0j} = \sum_{k=1}^{j} w_{\text{ins}}\left(a_k\right), \qquad for 1 \leq j \leq n \tag{7}$$

$$d_{ij} = \begin{cases} d_{i-1,j-1} \\ \\ \min \begin{cases} d_{i-1,j} + w_{\text{del}}\left(b_i\right) \\ d_{i,j-1} + w_{\text{ins}}\left(a_j\right) \\ d_{i-1,j-1} + w_{\text{sub}}\left(a_j, b_i\right) \end{cases} \end{cases} \tag{8}$$

For South African languages, Zulu et al. (2007) calculated this Levensthein distance measure, by comparing official textual documents, issued by the South African government in all the major languages spoken in South Africa. Moreover, Levenshtein distances were calculated using existing parallel orthographic word transcriptions of sets of 144 words from the official languages of South Africa. Their results are portrayed in Table 1.

From their results, we may observe interesting conclusions. For example, English and Afrikaans are closely related with a similarity score of 157, while Xitsonga and Sesotho are not closely related with a smilarity score of 448. We previously hypothesized that language similarity may strongly influence the generalization ability of a neural machine translation system. Moreover, based on the Levensthein distances, we expect that child language pairs with a lower distance from the parent language pair will outperform the results of the child language pairs with a higher distance from the parent language pair. In our task, we for example expect that translation quality for isiZulu will be worse than the translation quality for Sepedi, as the language similarity

Table 1: Language similarity score matrix calculated from Levenshtein distances for South African languages.

	Afrikaans	English	isiZulu	Sepedi	Setswana	Sesotho	Xitsonga
Afrikaans	0	157	451	279	390	452	390
English	157	0	444	276	382	438	389
isiZulu	451	444	0	384	426	430	399
Sepedi	279	276	384	0	186	271	363
Setswana	390	382	426	186	0	292	382
Sesotho	452	438	430	271	292	0	448
Xitsonga	390	389	399	363	382	448	0

score between Afrikaans and isiZulu is 451, while the language similarity score between Afrikaans and Sepedi is 279.

2.5 Summary

Overall, we have observed that transfer learning is a popular approach in the Natural Language Processing domain. Yet, the application of transfer learning in machine translation remains an unexplored task. Specifically, in neural machine translation (NMT), the application of transfer learning has the ability to solve the main issue of NMT that they require large data sets to yield promising results. We hypothesize that the language similarity between the source language and the target language significantly influences the performance of this method, as we observed that the relatedness between the source task and the target task significantly influences the results of a transfer learning approach. We distinguished two approaches to measuring language similarity and concluded that a quantifiable measure is most suited for our research. Based on previous research (Zulu et al., 2007), we established a metric for measuring language similarity between South African languages and based on this similarity, we may form our hypotheses. We namely expect that target languages that have a lower language similarity to the source language will perform worse than the target languages that have higher similarity to the source language.

3 Experimental setup

This section describes how our experiments were performed. More specifically, we describe which data were used to perform our experiments and the configuration details of our training methods. The goal of this section is to ensure that our experimental results can be replicated and to explain the decisions that were made in the choice of architecture for example, or in the choice of the various training parameters. We also present how we evaluate our experimental results and what we use as our baseline method to be able to answer the main research question we introduced in section 1.

3.1 Data

Our experiments were conducted using parallel textual data, provided by CTexT. They provided us with parallel training data for English and several South African languages. Basic statistics of the tokenized training corpus can be found in Table 2 below. Also, following Open NMT guidelines (Klein et al., 2017), we split our training data sets into a training set and a validation set. This to to evaluate the convergence of the training phase. All validation sets contain no more than 5000 sentences. To evaluate the results of our experiments, separate test sets were also provided by CTexT. These evaluation sets also consisted of different documents from the government domain. There were four different files per language, each translated by a different professional translator. These four different documents were combined in one file, which were subsequently used as our reference translations. All files consisted of 500 sentences per file.

Table 2: Basic statistics of the training data sets. Please note that ASL stands for average sentence length.

Language	Sentence pairs	Total words	Unique words.	ASL
English - Afrikaans				
English	421,318	2,052,928	48,765	24.3
Afrikaans		2,147,730	53,581	25.4
English - isiZulu				
English	35,489	249,666	39,886	25.7
isiZulu		215,539	38,095	19.7
English – Setswana				
English	31,376	219,834	36,786	21.6
Setswana		281,426	18,017	27.0
English - Sesotho				
English	42,386	223,634	39,050	21.1
Sesotho		197,879	28,283	28.6
English – Sepedi				
English	44,980	266,913	36,617	22.5
Sepedi		184,796	36,233	29.2
English - Xitsonga				
English	44,719	268,465	40,814	20.1
Xitsonga		257,331	30,751	25.4

3.2 NMT implementation

Our neural machine translation system is based on the OpenNMT (Klein et al., 2017) implementation of the *Transformer* architecture. In our review of the existing literature, we concluded that using an attention mechanism has significantly improved the performance of machine translation systems. Previous research (Luong et al., 2015) (Bahdanau et al., 2014) for example demonstrated that by ensuring that the decoder only attends to relevant parts of the source input, translation quality improved significantly. Moreover, we compared the *seq2seq with attention approach* with the *Transformer* approach and concluded that the *Transformer* approach has demonstrated to often outperform the *seq2seq with attention approach*. We therefore used the OpenNMT implementation of the *multi-head Transformer* architecture as our neural machine translation system.

We also noted that various different algorithms can be implemented for the encoder and the decoder. In our system, we opted that both the encoder and the decoder are long short-term memory (LSTM) recurrent neural nets (Hochreiter and Schmidhuber, 1997) with multiple hidden layers. We favored LSTM's over RNNs, due to the demonstrated ability to better handle long-term dependencies. Again, the encoder LSTM reads each input sequence one input token at a time updating the vector representation of the sequence read. Those representations are essentially the LSTM hidden states. The final LSTM hidden state (after seeing the end of sequence $<$ eos $>$ token in the source) is then used to initialize the decoder LSTM whose task is to generate the output sentence, again one token at a time.

3.3 Transfer learning approach

In our review of the existing literature we distinguished two main approaches to transfer learning and deemed the 'develop model approach' most suitable to our research. We also introduced the concepts of *domains* and *tasks*, where we distinguish between *source (domain, task)* and *target (domain, task)*. As the tasks in our research are part of the same *domain*, we from now on refer to a *source task* and a *target task*, as we also stated in subsection 2.3.

Concretely, in our research, we first trained an NMT model on a data set where there is a large amount of bilingual data, which we call the parent model, or our *source task*. In our research the English - Afrikaans language pair is our parent model, as there is an abundance of data available for this

language pair. Next, we initialize an NMT model with the already-trained parent model. This model is then trained on a data set with very little bilingual data, which we call the child model, or our *target task*. In essence, this means that the new NMT model did not start with random weights, but with the weights from the parent model. Training data sets for English to isiZulu, Sepedi, Setswana, Sesotho and Xitsonga are significantly smaller as we observed in Table 2 and thus, these language pairs are therefore considered to be our child models: English - isiZulu, English - Setswana, English - Sesotho, English - Sepedi and finally English - Xitsonga.

3.4 Training details

The Transformer model we used in this research is sensitive to setting the correct hyper parameters (Klein et al., 2017). We therefore replicated the *multi-head Transformer* architecture of Vaswani et al. (2017). Concretely, we used eight *attention heads* and 6-layered LSTMs for both the encoder and the decoder Luong et al. (2015). Similar to Sutskever et al. (2014), all out-of-vocabulary words were encoded with a special < UNK > character. Both the dimensionality of word embeddings and the LSTM hidden layer are of size 1000. Dropout (Srivastava et al., 2014) between the LSTM layers was set to 0.1. Finally, we used the Adam optimizer with $\beta_1 = 0.9, \beta_2 = 0.98$ and $\epsilon = 10^{-9}$.

3.5 Hardware

Our experiments were conducted using the Tesla P100 from Nvidia, using the NVIDIA Pascal architecture.

3.6 Baseline method

The main task in this project is a feasibility study of neural machine translation for resource scarce languages, more specifically the languages that are most frequently spoken in South Africa. Again, we use a transfer learning approach where we use train a model on a high-resource language and transfer their weights to a new model to train a model on a low-resource language. In order to evaluate this approach, we therefore compared the results of our transfer learning approach to the results of the experiments without transfer learning. In other words, we performed the experiments on

the South African languages without using the parameters from our parent model and use this as our main baseline for comparison. If the results of the experiments with our transfer learning approach outperform the results of the experiments without our transfer learning approach, we may state that our transfer learning approach is indeed a promising approach to neural machine translation for low-resource languages.

3.7 Evaluation metrics

In our review of the existing literature, we presented the Bilingual Evaluation Understudy Score (BLEU) as a method for evaluating the performance of machine translation experiments. This score is an evaluation metric for evaluating a generated sentence to a reference sentence. Again, the BLEU score P_n can be computed with the formula below by comparing n-grams of the candidate translation S with the n-grams of the source translation C and count the number of matches.

$$p_n = \frac{\sum_{S \in C} \sum_{ngram \in S} Countmatched(ngram)}{\sum_{S \in C} \sum_{ngram \in S} Count(ngram)} \tag{9}$$

However, in machine translation tasks, it is common to report multiple metrics. We therefore also report the Perplexity. Perplexity is a measure of the prediction error. In general, machine learning models with lower perplexity are more varied, and thus the model is able to better perform on unseen test data. It can be calculated using the formula below where *H(p)* is the entropy (in bits) of the distribution and x ranges over the events.

$$H(p) = 2^{-\sum_x p(x) \log_2 p(x)} \tag{10}$$

4 Experimental results

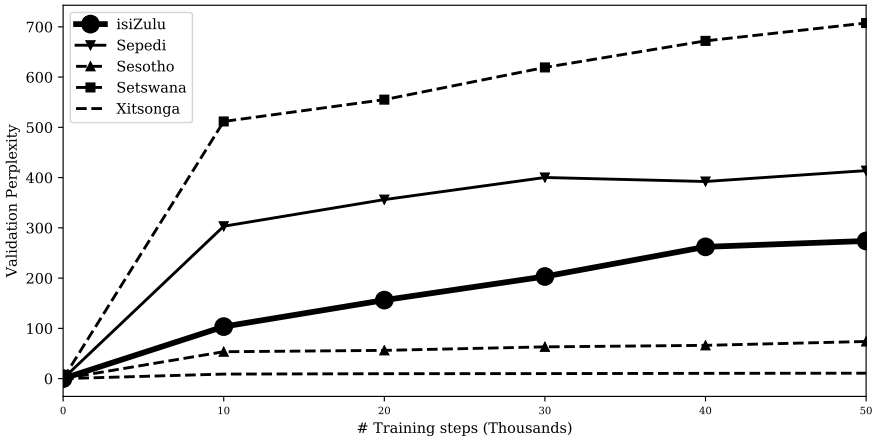
In order to evaluate the experimental setup we discussed in section 3, in this section, we compared the results of our transfer learning approach to the results of the experiments without transfer learning, using the BLEU score as our main evaluation metric. This research had two distinct goals. We first attempted to demonstrate the effectiveness of transfer learning for neural machine translation, specifically for low-resource languages, and second whether introducing a language similarity metric may influence this. In this section, we therefore first report the results of the transfer learning approach, where we compare the results with our baseline method. Second, we report the results in combination with the language similarity metric we introduced.

4.1 Transfer learning approach

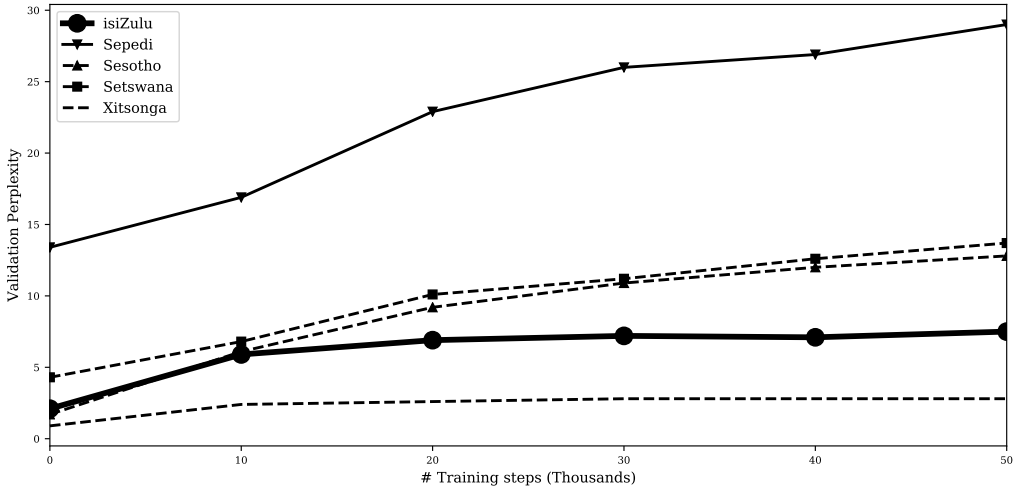
As we concluded in our review of the existing literature in section 2, the objective of the transfer learning approach is to learn the target conditional probability distribution $P_\eta\left(Y_T|X_T\right)$ of the *source task* D_T. One then utilizes the knowledge that is learned in the *source task* to the *target task* and transfer this knowledge to our *target task* by transferring the weights of the pre-trained model. You may therefore hypothesize that this transfer learning approach ensures that the model in the new task does not start from zero. This hypothesis is indeed confirmed by the figures that are portrayed in Figure 4. Here we use *perplexity* as a metric to measure the development of a model. We observed in section 3 that *Perplexity* is a measure of how variable a prediction model is, meaning that is capable of correctly handling different input data. Moreover, the lower the *Perplexity* the better the performance of a machine learning model on new unseen data.

We may observe from the two figures that *Perplexity* is high for our baseline method, for all our language pairs. Yet, for our transfer learning method, Perplexity is low, indicating that indeed our transfer learning approach improves how our models learn. However, what is striking is that *Perplexity* increases as the number of the training steps increases, while we expect it to decrease. There is no clear explanation for this. Therefore, we also present the figures for the validation accuracy as a function of the number of training steps for our language models. This to support the conclusion we have drawn from the figures where we presented the *Perplexity* as a function of the number of training steps for our language models. We again observe in Figure 5 that our transfer learning indicates that our models have gained knowledge from the *parent model* as we clearly observe that validation accuracy does not start from zero for all language pairs. Moreover, we observe a steep increase in validation accuracy at the beginning of the training phase. This may indicate that our language models quickly learn the distributions of the target language pair, even though it may be dissimilar to the parent language.

Overall, we may state that, indeed, knowledge is transferred from our *parent model* to our *child models* as we see that these *child models* demonstrate that they already have knowledge about the task. Yet, as previous studies (Zoph et al., 2016) have already demonstrated similar results, the novelty in our approach was taking into account the similarity between the *parent model* and the *child models*. These results are discussed next.

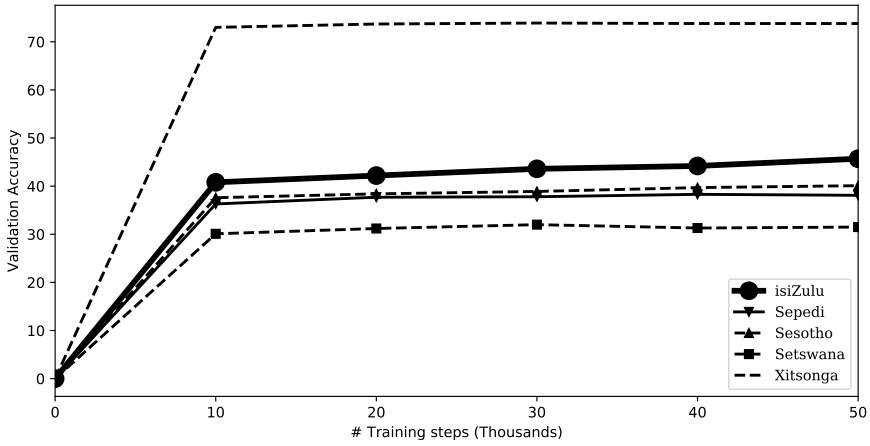


(a)

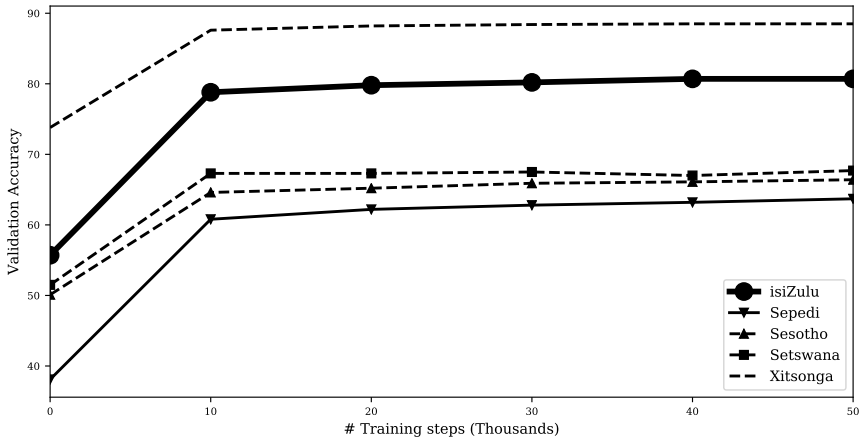


(b)

Figure 4: Validation perplexity as a function of the number of training steps (in thousands). Figure (a) portrays the baseline method, while figure (b) portrays the transfer learning method.



(a)



(b)

Figure 5: Validation accuracy as a function of the number of training steps (in thousands). Figure (a) portrays the baseline method, while figure (b) portrays the transfer learning method.

4.2 Language similarity

The main research question in this research was defined as follows:

> To which extent does language similarity influence the performance in transfer learning for low-resource neural machine translation?

Using the BLEU score as our main evaluation metric we present our experimental results in Table 3. Here, we report the BLEU score of our baseline method and compare this with the BLEU score of our transfer learning method. To be able to compare these two approaches, we also report the Δ BLEU to draw comparisons between the different language pairs. Finally, to be able to answer our research question, we report the language similarity to the parent language, for each language pair. We used the language similarity from the research by Zulu et al. (2007), where the lower the language similarity score, the more closely related the language pair is to the parent language.

Table 3: For each of our *child models* we present the BLEU score of our baseline method, the BLEU score of our transfer learning method, the Δ BLEU which represents the change in BLEU score and the language similarity to the parent language.

Language pair	**Baseline BLEU**	**Transfer Learning BLEU**	**Δ BLEU**	**Language similarity**
English - isiZulu	3.3	4.5	+1.2	451
English - Sepedi	14.8	19.7	+4.9	279
English - Sesotho	11.8	13.7	+1.9	390
English - Setswana	20.1	21.3	+1.2	452
English - Xitsonga	21.2	22.6	+1.4	390

We observe that in general, our transfer learning approach improves translation quality for each language pair. Yet, the improvements in BLEU are relatively small. What is striking is the improvement for the English-Sepedi language pair. As the this language pair is closest related to the parent language, Afrikaans, this improvement indicates that indeed language similarity does influence the performance in transfer learning for low-resource neural machine translation. Moreover, we observe the pattern that the higher the

language similarity score, the lower the improvement in BLEU. However, these results need to be interpreted with caution as the improvements are relatively low and do not differ significantly, except for the results of the English-Sepedi language pair.

What is also striking is the poor translation quality for the English - Zulu language pair, relative to the other language pairs. For this pair, we observed that translations often contained more information than in original sentence, leading to poor BLEU scores as also concluded in the research by Abbott and Martinus (2018), who conducted their experiments using the same data set.

4.3 Sample translations

In Table 4 we show sample translations generated by the transfer learning model and we compare these with the human translations and the source sentences. These translations demonstrate to which extent the various translations differ from each other, even though the source sentence was the same.

We may observe several interesting trends from these comparisons, which are highlighted in *Italic* in Table 4. First, in the English - Xitsonga sample translation, we observe that our transfer learning model was not able to correctly translate the Xitsonga translation of *South African Social Security Agency*. Instead it provides us with the English term. We see the same pattern in the English - isiZulu language pair, where again our transfer learning model was not able to correctly translate *South African Social Security Agency*, but it instead provided us with the English term. This indicates that neural machine translation systems have difficulty in correctly predicting certain terms, especially if these do not frequently appear in the training data.

Second, interestingly, we observe different spellings for certain words. For example, for the English - Sepedi language pair, our Transfer Learning system translated the word *South Africa* to *Africa Boroa*, while the human translator had translated it to *Africa Borwa*. These differences in spelling are not necessarily incorrect, but demonstrate the difficulty for neural machine translation systems to correctly predict certain words with multiple spellings, especially for languages that may have a lot of different spellings for the same word.

Second, it is interesting to observe how the system is able to provide synonyms for certain translations. In the English-Sesotho language pair, we observe that *cost effectively* is translated to the Sesotho equivalent of *affordable*, while the human translator has indeed used the direct translation of cost effectively. This is an interesting pattern since languages frequently have many synonyms for the same word. Yet, our model demonstrates that it can still capture the meaning reliably.

Finally, we see that for the English-Setswana language pair, our transfer learning model is not able to provide a reliable translation of the source sentence, as the translated sentence is not even closely related to the human translated sentence. This is an example sentence that demonstrates that neural machine translation systems do make mistakes, where they are unable to capture the meaning of the source sentence.

Table 4: Sample translations – for each language pair, we show an example of the source sentence (src), the human translation (ref) and the translation generated by our transfer learning model (transfer).

English - isiZulu	
src	This charter tells you about the standard of service you can expect from *the South African Social Security Agency*
ref	Le shatha ikwazisa ngezinga lesevisi ongalilindela *eHhovisi elibhekele ezenhlalakahle eNingizimu Afrika*
transfer	Leli khadi likutshela ngemigomo yesevisi ongayilindela *ku-South African Social Security Agency*
English - Sepedi	
src	SASSA is an extension of a government delivery arm that administers the delivery of social grants to the citizens of *South Africa*.
ref	SASSA ke koketšo ya letsogo la kabo la mmušo leo le sepetšago kabo ya ditšhelete tša mphiwafela go badudi ba *Afrika Borwa*.
transfer	SASSA ke ts'ebetso ea letsoho la puso ea 'muso le tsamaisang litšebeletso tsa sechaba ho baahi ba *Afrika Boroa*.
English - Sesotho	
src	To administer quality social security services, *cost effectively* and efficiently using the best practices.
ref	*Go* sepetša ditirelo tša tšhireletšo ya leago ya boleng, *ka ditheko tša phethagalo* le kgonagalo go šomišwa ditlwaetšo tše kaonekaone.
transfer	*Ho* fana ka litšebeletso tsa ts'ireletso ea sechaba, *ho lefella ka katleho* le ka katleho ho sebelisa mekhoa e metle.
English - Setswana	
src	We will use the information you give us to improve our services.
ref	Tšhata e go bolelela ka ga maemo a tirelo a o ka a solofelang go tswa go Bodiredi
transfer	Re tla dirisa tshedimosetso e e oketsegileng go tokafatsa tsenngo tirisong e.o
English - Xitsonga	
src	This charter tells you about the standard of service you can expect from the *South African Social Security Agency*
ref	Chata leyi yi ku byela hi mpimo wa vukorhokeri lebyi u nga byi langutelaka ku suka eka *Ejensi ya Nsirhelelo wa Vanhu ya Afrika-Dzonga*
transfer	Tsalwa leri ri vula hi nkarhi wa vukorhokeri bya nkarhi lowu u nga langutela ku suka eka *South African Social Security Agency*

5 Discussion

In this section we discuss our experimental results that were presented in section 4. We also attempt to answer the main research question in this research, which was defined as follows in section 1:

> To which extent does language similarity influence the performance in transfer learning for low-resource neural machine translation?

To answer this, we used a transfer learning approach where we trained a model on a high-resource language and transferred their weights to a new model to train a model on a low-resource language. We portrayed our experimental results in section 4 and in this section we review these and discuss their implications within the neural machine translation domain.

5.1 Transfer learning approach

In order to evaluate our transfer learning approach, we compared the results of our transfer learning approach to the results of the experiments without transfer learning. This was our baseline method as it may indicate whether a transfer learning approach is indeed successful for low-resource languages, as we observed that for all our language pairs, our transfer learning approach yielded an improvement in BLEU score.

Previous studies (Zoph et al., 2016) (Wang and Zheng, 2015) have already demonstrated similar results. Thus, our results therefore confirm the hypothesis that transfer learning is indeed a successful approach for low-resource neural machine translation. These results were best observed in Figure 4 and in Figure 5 where we visualized the training process by portraying the validation perplexity and the validation accuracy as a function of the number of training steps. These figures demonstrated that our models had gained knowledge from the training process of the English-Afrikaans language pair which was our *parent model* or our *source task*. Moreover, neural machine translation systems, or any machine learning model, typically start training from scratch, as they obtain knowledge of the task as the number of training steps increases. As it is exposed to more training instances, the model gets better at recognizing patterns, thus increasing the performance of the model. What is therefore interesting to induce from the results of our approach, is that our transfer learning method results in the model already having some

knowledge of the task prior to the start of the training phase. Moreover, we observed in the figures that there is a steep increase in the beginning of the training phase, indicating that the model adjusts to and familiarizes with the structure of the new training instances. Of course, even though the task is similar, the type of language data is still very different, especially given that languages are very complex. It is therefore impressive that our systems demonstrate the ability to handle these variations in languages. We for instance provided sample translations in section 4 that demonstrated how our system generated these translations. Even though the ability was impressive, as the system is able to for instance provide synonyms for certain translations. Yet we also observed the mistakes that our system made, where it was not able to generate any meaningful translations, indicating the complexity of the task of machine translation in general.

Summarizing, even though we demonstrated that a transfer learning approach may be effective for low-resource neural machine translation, this was not the main goal of this research. Our goal was to evaluate a transfer learning approach, in conjunction with introducing a language similarity metric. The results of this novel approach are further discussed in the next section.

5.2 Language similarity

Again, the novelty in our approach was the introduction of a language similarity metric. In our task on machine translation for South-African languages, we used the Levensthein distance measure. Previous research (Zulu et al., 2007) calculated this Levensthein distance by comparing official textual documents, issued by the South African government in all the major languages spoken in South Africa. We subsequently used this distance measure to objectively quantify the differences between the major languages spoken in South Africa. Our hypothesis was that the more similar *the child language pair* was to the *parent language pair*, the better the translation quality was of our transfer learning approach. Using the BLEU score as our main evaluation method, the results indeed indicate that language similarity influences the performance in transfer learning for low-resource neural machine translation. This was most clearly demonstrated in the results for our English-Sepedi language pair. As this language pair is most closely related to the parent language pair, English-Afrikaans, it also yielded the strongest improvement in BLEU score. This indicates that language similarity may indeed influence the performance. Yet, we should also note that the improvements in BLEU

score for our other language pairs were not as strong as hypothesized. One could argue that this may be since these language pairs are more dissimilar to the parent language pair. On the other hand, it may also be that this indicates that language similarity may not influence the performance. Nonetheless, we may state that our results are promising and should be demonstrating that more future research is needed. Yet, one main question still is what exactly is transferred from the *source task* to the *target task*. An understanding of this may be helpful in improving the results of transfer learning for neural machine translation. We therefore discuss this in the next section.

5.3 What knowledge is transferred?

In our review of the existing literature, we observed that the idea behind transfer learning is that one uses knowledge from a a *source task* in a certain *source domain* to improve the performance on a *target task* in a *target domain*. However, we also observed that the relationship between these has remarkably remained largely unexplored. Intuitively, one can expect that the more similar the *source task* and the *target task* are, the better the results of a transfer learning approach are expected to be. Therefore, the novelty in this research was that we introduced a similarity metric where we measure the similarity between languages. In our research the English - Afrikaans language pair was our parent model, or our *source task*. In addition we had several language pairs which were considered to be our *child models*, or our *target tasks*.

Our results indicate that our *child models* indeed benefit from the transfer of the weights of the pre-trained *parent model* as the training phase progress demonstrates that these *child models* already have gained knowledge about the task prior the start of the training phase.

Even though our results indicate this, it remains uncertain what exactly the model learns. In our review of the existing literature, we concluded that the objective of transfer learning is to learn the target conditional probability distribution $P_\eta(Y_T|X_T)$ of the *source task* D_T. With this, we demonstrated that the knowledge of the *source task* that is transferred to the *target task* is essentially the prior distribution of the *source task*. In our research we used the English-Afrikaans language pair as our *parent model* or our *source task* as there was an abundance of data available for this language pair. This is especially important given that large amounts of data are required for neural machine translations to be effective. For our *child models*, we therefore selected language pairs where little bilingual data was available. Subsequently, our hypothesis was that the prior distribution of the English-Afrikaans language pair would be similar to our *child models*, possibly removing the need for large amounts of bilingual data to effectively implement neural machine translation systems. We indeed demonstrated that a transfer learning approach is a promising method for low-resource neural machine translation, yet our results do not provide a clear explanation, thus we may only assume what the knowledge that is transferred between the two tasks entails. Therefore, in the next section we discuss how future work may benefit from the results in our research.

5.4 Future work

Neural machine translation systems typically require vast amounts of bilingual data yield effective and reliable results. For instance, previous studies (Sutskever et al., 2014) (Luong et al., 2015) (Kalchbrenner et al., 2014) have demonstrated that neural machine translation systems often outperform traditional machine translation systems, when there is plentiful bilingual data available. Yet, for language pairs where this is not available, traditional machine translation systems still outperform neural machine translation systems as they require less training data. This task is often referred to as *low-resource neural machine translation* and various approaches have been proposed to solve this issue of data scarcity. We observed in our review of the existing literature that one such an approach is transfer learning, where one uses knowledge from a *source task* to improve the performance on a *target task*. We also observed how the similarity between these tasks has been remained unexplored. Our work is a first step into this direction as our results indicate that similarity indeed may influence the success of such a transfer learning approach. Yet, more future work is needed before we are able to draw firm conclusions. In general, recently we have seen a surge in the research that is published where the common shared goal is to improve model performance with limited amount of training data. Not only in the neural machine translation domaine, but also in other domains including for instance image recognition (Han et al., 2018). Another recent approach is the *Few-Shot Learning (FSL)* approach, where the goal is to rapidly generalize from limited amounts of training data (Garcia and Bruna, 2017). This also demonstrates the importance of the work in this area as the main goal is to develop reliable machine learning models, using limited amounts of data. The results of our work may therefore be considered to be useful for future work in this domain.

6 Conclusions

In this research, we proposed to introduce a metric for measuring language similarity in order to improve the results of transfer learning for neural machine translation systems. In general, the key motivation for applying a transfer learning approach is to reduce the need for vast amounts of labeled data, as this is often required for successful supervised machine learning models. This is especially true within the domain of neural machine translation, as vast amounts of bilingual data are required to develop reliable and effective machine translation systems. Especially for models with a large number of parameters that are particularly sensitive to the quality of data, including the *Transformer* architecture implemented in this research (Vaswani et al., 2017). In section 2, we stated that the idea behind transfer learning is that one uses knowledge from a *source task* to improve the performance of a *target task* and one essentially stores the knowledge that is gained in the *source task* and applies it to the *target task*, reducing the amount of required training data. Our novel approach was the introduction of a language similarity measure, which we used to determine the similarity between the *source task* and the *target task*. We were interested to see whether that influences machine translation quality as we hypothesized that the more similar these tasks would be, the better the results of a transfer learning approach.

In our task on neural machine translation for South-African languages, we conducted experiments using bilingual data provided by CTexT where we used the English-Afrikaans language pair as our *source task* or our *parent model* and other language pairs as our *child models* or our *target task*. Our results indicated an improvement, measured in BLEU, for all of our *child models*. We observed the strongest improvement in our English-Sepedi language pair, which was closest related to the *parent model*. We also observed in the visualization of the training phase, that knowledge of the task is transferred from the *source task* to the *target task* as our *child models* demonstrated this by not starting the training phase from scratch as typical neural machine translation systems or machine learning models.

Even though these results are promising, we should note that more works needs to be done before we may state that indeed language similarity strongly improves the results of low-resource neural machine translation. Yet, our results are an indication of the influence and should be a starting point for future work in the area of neural machine translation and more importantly the area of transfer learning in general.

Bibliography

Abbott, J. Z. and Martinus, L. (2018), 'Towards neural machine translation for african languages', *arXiv preprint arXiv:1811.05467* .

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

Bentivogli, L., Bisazza, A., Cettolo, M. and Federico, M. (2016), 'Neural versus phrase-based machine translation quality: a case study', *arXiv preprint arXiv:1608.04631* .

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

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

Chorowski, J. K., Bahdanau, D., Serdyuk, D., Cho, K. and Bengio, Y. (2015), Attention-based models for speech recognition, *in* 'Advances in neural information processing systems', pp. 577–585.

Eska, J., Durie, M. and Ross, M. (1997), 'The comparative method reviewed: Regularity and irregularity in language change', *Language* **73**, 893.

Garcia, V. and Bruna, J. (2017), 'Few-shot learning with graph neural networks', *arXiv preprint arXiv:1711.04043* .

Guthrie, M. (1971), *Comparative Bantu: An Introduction to the Comparative Linguistics and Prehistory of the Bantu Languages. Bantu Prehistory, Inventory and Indexes*, Vol. 2, Gregg.

Han, D., Liu, Q. and Fan, W. (2018), 'A new image classification method using cnn transfer learning and web data augmentation', *Expert Systems with Applications* **95**, 43–56.

Hartshorne, J. K., Tenenbaum, J. B. and Pinker, S. (2018), 'A critical period for second language acquisition: Evidence from 2/3 million english speakers', *Cognition* **177**, 263–277.

Heeringa, W., Golubovic, J., Gooskens, C., Schüppert, A., Swarte, F. and Voigt, S. (2013), 'Lexical and orthographic distances between germanic, romance and slavic languages and their relationship to geographic distance', *Phonetics in Europe: Perception and Production* pp. 99–137.

Hochreiter, S. and Schmidhuber, J. (1997), 'Long short-term memory', *Neural computation* **9**(8), 1735–1780.

Hutchins, W. J. and Somers, H. L. (1992), *An introduction to machine translation*, Vol. 362, Academic Press London.

Jurasfsky, D. and Martin, J. H. (2000), 'An introduction to natural language processing', *Computational Linguistics, and Speech Recognition, Prentice Hall, New Jersey* .

Kalchbrenner, N., Grefenstette, E. and Blunsom, P. (2014), 'A convolutional neural network for modelling sentences', *arXiv preprint arXiv:1404.2188* .

Kessler, B. (1995), Computational dialectology in irish gaelic, *in* 'Proceedings of the seventh conference on European chapter of the Association for Computational Linguistics', Morgan Kaufmann Publishers Inc., pp. 60–66.

Klein, G., Kim, Y., Deng, Y., Senellart, J. and Rush, A. M. (2017), 'Opennmt: Open-source toolkit for neural machine translation', *arXiv preprint arXiv:1701.02810* .

Kocmi, T. and Bojar, O. (2018), 'Trivial transfer learning for low-resource neural machine translation', *arXiv preprint arXiv:1809.00357* .

Koehn, P. and Knowles, R. (2017), Six challenges for neural machine translation, *in* 'Proceedings of the First Workshop on Neural Machine Translation', Association for Computational Linguistics, pp. 28–39.
URL: *http://aclweb.org/anthology/W17-3204*

Koehn, P., Och, F. J. and Marcu, D. (2003), 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-Volume 1', Association for Computational Linguistics, pp. 48–54.

Lakew, S. M., Cettolo, M. and Federico, M. (2018), 'A comparison of transformer and recurrent neural networks on multilingual neural machine translation', *arXiv preprint arXiv:1806.06957* .

Luong, M.-T., Pham, H. and Manning, C. D. (2015), 'Effective approaches to attention-based neural machine translation', *arXiv preprint arXiv:1508.04025* .

Ott, M., Edunov, S., Grangier, D. and Auli, M. (2018), 'Scaling neural machine translation', *arXiv preprint arXiv:1806.00187* .

Pan, S. J. and Yang, Q. (2010), 'A survey on transfer learning', *IEEE Transactions on knowledge and data engineering* **22**(10), 1345–1359.

Papineni, K., Roukos, S., Ward, T. and 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', Association for Computational Linguistics, pp. 311–318.

Ruder, S. (2019), Neural Transfer Learning for Natural Language Processing, PhD thesis, National University of Ireland, Galway.

Sennrich, R., Haddow, B. and Birch, A. (2016), Improving neural machine translation models with monolingual data, *in* 'Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)', Association for Computational Linguistics, pp. 86–96. **URL:** *http://aclweb.org/anthology/P16-1009*

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R. (2014), 'Dropout: a simple way to prevent neural networks from overfitting', *The Journal of Machine Learning Research* **15**(1), 1929–1958.

Sutskever, I., Vinyals, O. and Le, Q. V. (2014), Sequence to sequence learning with neural networks, *in* 'Advances in neural information processing systems', pp. 3104–3112.

Thompson, B., Roberts, S. and Lupyan, G. (2018), Quantifying semantic similarity across languages, *in* 'Proceedings of the 40th Annual Conference of the Cognitive Science Society (CogSci 2018)'.

Torrey, L. and Shavlik, J. (2010), Transfer learning, *in* 'Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques', IGI Global, pp. 242–264.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. and Polosukhin, I. (2017), Attention is all you need, *in* 'Advances in neural information processing systems', pp. 5998–6008.

Wang, D. and Zheng, T. F. (2015), Transfer learning for speech and language processing, *in* '2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)', IEEE, pp. 1225–1237.

Zens, R., Och, F. J. and Ney, H. (2002), Phrase-based statistical machine translation, *in* 'Annual Conference on Artificial Intelligence', Springer, pp. 18–32.

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

Zulu, P., Botha, G. and Barnard, E. (2007), 'Orthographic measures of language distances between the official south african languages', *Literator* **29**.