SACJ 57 December 2015
Research Article

Syllabification and parameter optimisation in Zulu to English machine translation

Gideon Kotzé, Friedel Wolff

University of South Africa

ABSTRACT

We present a series of experiments involving the machine translation of Zulu to English using a well-known statistical software system. Due to morphological complexity and relative scarcity of resources, the case of Zulu is challenging. Against a selection of baseline models, we show that a relatively naive approach of dividing Zulu words into syllables leads to a surprising improvement. We further improve on this model through manual configuration changes. Our best model significantly outperforms the baseline models (BLEU measure, at $p < 0.001$) even when they are optimised to a similar degree, only falling short of the well-known Morfessor morphological analyser that makes use of relatively sophisticated algorithms. These experiments suggest that even a simple optimisation procedure can improve the quality of this approach to a significant degree. This is promising particularly because it improves on a mostly language independent approach—at least within the same language family. Our work also drives the point home that sub-lexical alignment for Zulu is crucial for improved translation quality.

Keywords: machine translation, word segmentation, alignment, Zulu, English

Categories: **Computing methodologies ~ Machine translation**, *Applied computing ~ Language translation*

Email:
Gideon Kotzé kotzegj@unisa.ac.za (CORRESPONDING),
Friedel Wolff wolfff@unisa.ac.za

Article history:
Received: 5 May 2015
Accepted: 6 Nov 2015
Available online: 10 Dec 2015

1 INTRODUCTION

Statistical machine translation (SMT) is an approach where the successful translation of a given text is regarded as a mathematical problem to be solved by computational means. Although its limitations are well known and it is by no means at the level of replacing the hard work done every day by professional translators across the globe, the rise of SMT in the last decade or so has been remarkable.

Machine translation (MT) has found great application for many end-users in products such as Google Translate, providing a way of getting the "gist" of something in another language. Apart from improving accuracy, there has in recent years also been much focus on ease of use, integration in existing products and websites, and broad coverage of the languages involved.

Kotzé, G., and Wolff, F. (2015). Syllabification and parameter optimisation in Zulu to English machine translation. *South African Computer Journal 57*, 1–23. http://dx.doi.org/10.18489/sacj.v0i57.323

SACJ is a publication of the South African Institute of Computer Scientists and Information Technologists. ISSN 1015-7999 (print) ISSN 2313-7835 (online).

In the translation industry, MT is used as a tool for both improving the speed and accuracy of the human translation process. Moreover, in the field of natural language processing (NLP), MT can also be applied to solve other problems, such as cross-lingual information retrieval or for the extrinsic evaluation of more basic tasks.

Training requires a relatively vast selection of digital text to serve as examples for the chosen MT system. For languages with less data available, this is a limiting factor. Most of the official South African languages suffer from this drawback. This is one of the important issues that we need to consider in our research.

In this paper, we look at the case of translating from Zulu (the source text) to English (the target text). As the most widely spoken mother-tongue in South Africa, we believe that any positive outcomes of our research on translation experiments using Zulu may have a wide impact.

One of the most challenging aspects of the computational modeling of Zulu is its morphological structure. The *morphology* refers to the various grammar rules specifying how the different parts of the words—so-called *morphemes*—such as prefixes, suffixes and stems, are written and combined with each other. In the case of Zulu, many of these morphemes correspond to syntactic categories such as pronouns and certain types of function words. This phenomenon is referred to as *agglutination*.

Zulu also has a so-called *conjunctive* writing system, meaning that many single words contain elements from multiple syntactic categories. Therefore, the abovementioned complex morphological constructions resulting from agglutination tend to be written as single words. Such a word may even correspond to a whole sentence. One example is the word “ngisamthanda”, which means “I still love him/her”.

Note that it is not possible to automatically determine, for example, that “love” corresponds to “thanda” unless we analyse the text in some way below the word level. Ideally, this requires a morphological analysis, where the morphemes would more or less correspond semantically to their similarly analysed counterparts in the other language, making them easier to align and learn the translations, improving accuracy. Because morphological analysis for Zulu is complex and a separate analyser for each new language has to be developed, we decided to build on the research presented in Wolff and Kotzé (2014) where a simple syllable-based approach is followed, and which we hope is a more generalisable alternative to the manual construction of analysers in the context of MT.

For the purposes of this work, a syllable is simply defined as a substring of a word consisting of zero or more consonants and ending on a vowel, with hyphens removed. The method entails the splitting of Zulu words into syllables so that each one is treated as a *token*. A token is the basic unit of processing which traditionally is either a word or punctuation mark.

In this way, a simple approximation to Zulu morphological analysis is attempted. The choice of translating to English, rather than the other way around, has the advantage that the output does not need any postprocessing and can be readily inspected by anyone with a knowledge of English, as well as the fact that target language models (see next section) can be trained based on huge amounts of freely available English text.

In Figure 1, we show a small example of the kind of preprocessing that we describe here. On the left side (a), no word segmentation has been done. The words “of the Cabinet” are simply aligned to the single Zulu word “eKhabhinethi”. The fact that, for example, “of the” corresponds to “e-”, is

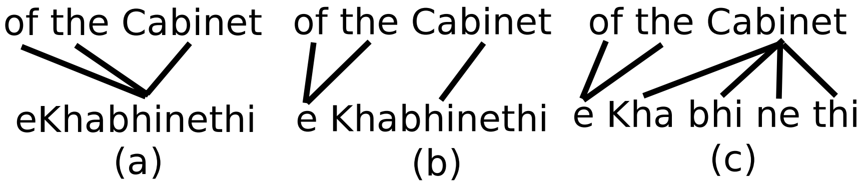

Figure 1: Different alignments with Zulu text being represented as (a) a single word, (b) morphemes, and (c) syllables.

undiscoverable.

At (b), "eKhabhinethi" has been morphologically segmented into its constituent parts: the prefix "e" and the stem "Khabhinethi". Here, "of the" can be aligned to "e", and "Cabinet" to "Khabhinethi", indicating equivalence on a finer scale.

At (c), the Zulu word has been segmented on a syllabic level. Note that the stem "Khabhinethi" has been oversegmented—there is no real need to align any part of the string "Khabhinethi" below this level. However, the "e" is still correctly segmented, and perhaps even more importantly, the distinction between the stem and the prefix is preserved. This suggests that syllabification, although not perfect, may serve to be a useful approximation to morphological analysis in the sense that the boundaries between important morphemes tend to be preserved for alignment purposes.

The main focus of this work is on the optimisation of the syllable-based approach. We achieve this by investigating a number of parameters: the token alignment[1] approach, the phrase length used by the translation as well as the target language model, and the word aligner used. In addition, we compare the syllable-based models against a number of baselines. The parameters of all models have been tuned using the Minimum Error Rate Training algorithm (Och, 2003).

In the next section, we present a brief background of related research. This is followed by a description of the data that we used in our experiments and how they were preprocessed (Section 3). We present the design, implementation and results of our experiments in Section 4. This is followed by a statistical analysis of significance (Section 5). Next, in Section 6, we present a qualitative evaluation. This is followed by a discussion of our results, with reference to future work (Section 7). Finally, we present our conclusion in Section 8.

2 BACKGROUND

Statistical machine translation is based on the idea of viewing the text in the source language as a variant of the target language that was transmitted through a noisy channel. The search for the best

[1] Note that in the context of this article, the term *token alignment* is used as a more suitable term than the more well-known *word alignment*.

English translation $\hat{E}$ from all possible English strings e^* is often formulated according to Bayes' rule as follows:

$$\hat{E} = \underset{e \in e^*}{argmax}\; p(z|e)p(e) \tag{1}$$

where z is the Zulu (input) text and where we attempt to choose or construct the optimal e from the training data to optimise the probabilities. The first factor $p(z|e)$ refers to decoding using the translation model, and the second factor $p(e)$ to language modelling to ensure fluent output in the target language.

The SMT paradigm within which we perform our current experiments is called *phrase-based statistical machine translation* (PBSMT). It has its origins in word-based SMT, which was an IBM initiative (Brown et al., 1990) and has been around since the late 90s (Koehn, 2010b). The text is processed in phrases—non-linguistic contiguous tokens that co-occur. Although this limits the abilities of PBSMT to a degree, it still compares favourably to state-of-the-art systems.

The second factor in equation 1 refers to target language modeling. This is used to ensure fluency in MT output. Chen and Goodman (1999) define a language model as "a probability distribution over strings $p(s)$ that attempts to reflect the frequency with which each string s occurs as a sentence in natural text," whereas Stolcke (2002) defines statistical language modeling as "the science (and often art) of building models that estimate the prior probabilities of word strings".

For phrase-based SMT based on Zulu words, if the phrase table (translation model) does not contain a certain phrase, it would have to fall back to smaller and smaller segments, until it reaches the word level.

If a source language word is not present in the phrase table, it can not be translated, and normally might simply be duplicated into the target language, or dropped entirely. With Zulu as source language, this can happen quite frequently, since a great number of words is possible because of the complex morphology. The Zulu *lexicon*—the list of all possible words—is therefore very large.

A very important process in SMT is the alignment of words in order to help construct the phrase table. In the introduction, we presented the main problems of the alignment of English with Zulu words, as well as word segmentation as a possible way to mitigate this. A very common form of word segmentation occur with morphological analysis, which is a well-studied problem in the field of natural language processing. This entails the segmentation of a word into its constituent morphemes, possibly also labeling the morphemes in the process. An introduction can be found in Jurafsky and Martin (2009). Software to perform this task has been developed for Zulu (Pretorius & Bosch, 2003). This analyser is not freely available, and although the basic rules of Zulu morphological analysis is well understood, the work required to assemble a big enough root lexicon for a high-accuracy morphological analyser is considerable. On the other hand, the syllable-based method is much more language independent and should be applicable to a variety of languages within the Bantu family.

The intuition for using syllables as cheap substitutes for morphemes stems from the fact that many Zulu prefixes are indeed single syllables and often have obvious alignments with a parallel English text. For example, verb prefixes indicating subject (*si-*) and future tense (*-zo-*) are single syllables in *si+zo+hamb+a* (English: "we will walk"). Although a multi-syllabic stem will be oversegmented, our hope is that it will be transferred to the target language semantically intact due to frequent

http://dx.doi.org/10.18489/sacj.v0i57.323

co-occurrence as a multi-syllabic phrase.

Zulu and most languages in the Bantu family have a preference for open syllables (Spinner, 2011). This means that syllables occur in a very regular way, making syllabification easy to perform, even though this is only a rough substitute for proper morphological analysis. This also gives rise to a unique and unambiguous analysis in each case—thereby removing the need to address aspects such as disambiguation at this stage.

Token alignment between English and syllabified Zulu (Kotzé & Wolff, 2014) showed that semantically significant syllables could be identified in automatic token alignment. This supports our hypothesis that syllabification is not an arbitrary division, but that it can isolate semantically meaningful units, at least to some degree.

The automatic induction of a morphological analyser is possible with supervised, semi-supervised and unsupervised methods (Spiegler, Golénia, Shalonova, Flach & Tucker, 2008; Quasthoff, Bosch & Goldhahn, 2014). We do not provide a thorough comparison here, but note some differences compared to the simpler syllable-based approach:

- Supervised systems require an extra step in the form of the construction of training data. This can be expensive in the case of morphological analysis and is not an ideal situation for us. We realise that resource scarceness as applied to Zulu and its related languages does not only apply to corpora and NLP technologies but also to economic resources being spent on the required research and development. By the application of our approach, we therefore hope to alleviate this requirement as well.

- Being based on machine learning, an automatically induced morphological analyser would have some level of dependence on the domain and style of the training data. The syllable-based approach is inherently domain and genre independent. Such dependence of course exists also for a machine translation system trained on the same data.

- Depending on the exact approach of the induced morphological analyser, the matter of disambiguating between analyses might remain. Although all analyses can be added to a lattice in the SMT engine, it is not clear how ambiguous analyses for tokens in a sentence would be handled in token alignment. Disambiguation is not required with the syllable-based approach, since only a single output is produced.

Such unsupervised morphological analysers have been used to generate the training data for morpheme-based machine translation engines which resulted in slightly lower evaluation scores (according to the BLEU metric) within the context of the complex morphology of the Nordic languages (Virpioja, Väyrynen, Creutz & Sadeniemi, 2007) as well as for Czech to English (Virpioja, Väyrynen, Mansikkaniemi & Kurimo, 2010). In contrast to these results, an automatically induced morphological analyser for Zulu was used to segment the Zulu data in an English to Zulu MT system which improved results, admittedly over very low baseline scores (Van Niekerk, 2014). Morphological analysis on Swahili text (a related language) has been shown to improve token alignment (De Pauw, Wagacha & Schryver, 2011)—the first step in training an SMT system.

Translating on the level of characters is another sub-lexical approach to SMT that has been attempted before. For example, Tiedemann (2009) and Nakov and Tiedemann (2012) combine word-based and character-based approaches to improve translation scores between closely related languages (Norwegian to Swedish, Macedonian to Bulgarian). What is of interest here is the fact that they only apply a small data set for the task with some success.

Before one can build and test models, the text to be used in training, tuning and evaluation must be prepared properly. This mostly consists of tokenisation, sentence splitting and sentence alignment.

Tokenisation consists of separating words from punctuation, such that each can be processed as a separate unit during the training and translation process. Words and punctuation marks are treated as the same type of unit: the *token*.

Sentence splitting is the practise of separating sentences from each other. The reason for this is to limit the search space for algorithms to one sentence at a time. In practise, this occurs by placing each new sentence on a separate line. Although sentence splitting excludes the possibility of learning information relating to surface structures which may span over sentence boundaries, such as the words "although", "yet" and some types of references such as anaphora, this is generally regarded as a fair trade-off by the SMT community.

Once they are in the proper machine readable format, the texts are to be aligned on various levels in order to facilitate extracting translational equivalents for building the translation model. The first alignment step takes place on the sentence level. In many cases, SMT is applied to one-to-one pairs of sentences only, in order to limit the search space for applying the statistical algorithms. Although this excludes the possibility of the accurate modeling of patterns where the sentence boundary is exceeded—such as is the case with so-called discourse units—this is considered mostly adequate, especially since this vastly reduces the hypothesis space for token alignment and thus the time and memory required for the task. Sentence alignment itself often works with the assumption that the sentences on both sides are in the same order.

The second alignment step is token alignment, more generally known as *word alignment*. This is a much harder problem, since parallel texts are seldom word-for-word translations of each other. A common approach is to align a bitext twice: a single token is allowed to be aligned with multiple tokens on the other side, and then vice versa. The two alignment sets can then be combined in various ways to find a good balance between high precision (intersection) and high recall (union) alignments. The two tools that we applied (Section 4) both implement this method.

The output of token alignment is used for extracting phrases and estimating translation probabilities in PBSMT systems. As mentioned before, by using Minimum Error Rate Training (MERT), certain learned system parameters can be optimised for the final model according to some specified metric and using a held-out data set.

For our quantitative evaluation step, we apply the BLEU, METEOR and TER metrics. The basic idea that they have in common is that they measure some kind of distance between the *hypothesis* (system output) and the *reference* translation (gold standard). The "closer" the hypothesis, the better the score. The way that this is determined differs between the metrics.

BLEU[2] (Papineni, Roukos, Ward & Zhu, 2002), one of the most widely used metrics, is an *n-gram*

[2] Bilingual Evaluation Understudy

based metric that does not make use of any linguistic information. The term *n-gram* refers to any substring of a length of *n* units, such as tokens. For example, if one applies a token-based method that makes use of *bigrams*, all substrings consisting of two tokens are considered. Similarly, *trigrams* refer to substrings of three tokens. The unit is not necessarily limited to tokens. For example, in Section 4 we apply models that are based on character *n*-grams.

METEOR (Banerjee & Lavie, 2005)[3] is in many aspects similar to BLEU, but attempts to address some of its perceived weaknesses. It also supports the use of language specific methods such as the detection of synonyms, reducing the chances that highly related translations are unfairly penalised. It is very well supported for English, making this a good choice for our experiments.

Translation Edit Rate (TER), also called Translation Error Rate (Snover, Dorr, Schwartz, Micciulla & Makhoul, 2006),[4] measures the amount of editing necessary for an automatic output to equal the reference translation. Generally, the more edits a system output requires, the worse the translation is perceived to be, while conversely, fewer edits suggest a better translation. Hence, lower scores are better.

In the next section, we describe the data we used for our experiments and how they were preprocessed.

3 DATA AND PREPROCESSING

We used the same data sets and tools as described in Wolff and Kotzé (2014), which we describe below for completeness. The results reported in this paper is therefore comparable.

One way to categorise text data is according to their organisation or how they are used. The first distinction is between so-called monolingual and parallel text. The former is only written in a single language. The latter—also called a *bitext*—is a collection of text in two languages, usually translations of each other.[5] Our monolingual text consists of English only, the reason being that it is reserved for the target language modeling component to ensure greater fluency in the English output. The bitext consists of parallel Zulu/English texts for creating the translation model.

The bitext is divided up into different sets: a training data set, a development test (so-called devtest) set and a tuning set. The former is used for training, the devtest for continuous evaluation and the tuning set is used for applying a selected algorithm to optimise the weights of the translation model. A final held-out test set may also be used as a stricter objective evaluation measure. This usually includes text from other domains, or at least other documents, than those included in the training data, in order to test the robustness of the models. In our case, such a test set exists, but it was extracted from the same documents that we used for for the devtest set, because of the relative lack of available resources. We did not include the final test set at this point in time, as we might still have some use for it in the future.

For all these phases involved—training, tuning and evaluation—we needed to build a bitext from existing resources. We used the Bible, the Autshumato English/Zulu corpus (McKellar & Groenewald,

[3] Metric for Evaluation of Translation with Explicit ORdering

[4] http://www.cs.umd.edu/~snover/tercom

[5] In the case of three or more languages, this fact is usually made explicit by using such terms as *multilingual corpus*.

Table 1: Corpus statistics after sentence alignment

	Bible	Autshumato	Constitution	Total
Sentence pairs	39 916	36 292	2 788	78 996
Zulu words	380 432	415 976	36 190	832 598
Zulu syllables	1 116 900	1 428 983	109 974	2 655 857
English words	626 187	554 212	47 602	1 228 001

2012) and the South African constitution of 1996.[6]

The Zulu Bible is the 1959 version,[7] and for English, we used the World English Bible British English Edition.[8] The latter has a few advantages in the sense that it has a more modern lexicon than, for example, the King James, is in the public domain, is published in an easily readable XML format, uses British spelling which is more appropriate in the South African context, and its use of punctuation, such as quotation marks, is similar to its Zulu equivalent. The translation followed a so-called formal equivalence approach, similar to the Zulu Bible (Hermanson, 2002), which means that both are more literal translations of the original texts, thereby hopefully improving alignment.

Table 1 shows some count statistics from the corpus. In the composition of the SAWA corpus for English/Swahili (De Pauw et al., 2011), a similar majority of religious text was also reported, even though a greater attempt was made there to include texts of a wider variety.

For any text to be used for training in tasks such as SMT, it needs to be preprocessed. We applied the following steps:

- removal of undesirable elements, such as markup and erroneously encoded segments
- tokenisation, for which we applied TreeTagger (Schmid, 1994)[9] for both the English and the Zulu text. As a post-processing step, we isolated the em-dash (Unicode: U+2014) as a token of its own.
- sentence splitting, for which we used the split-sentences.perl script[10] that is included with Moses (Koehn et al., 2007), the SMT system that we adopted for our experimental work. Additionally, a script was applied to properly handle embedded quotes and also to split sentences at semicolons and colons.
- sentence alignment, for which we used Hunalign (Varga et al., 2007).[11] This was necessary since the bitexts were not perfectly aligned: The Bible and constitution texts have been extracted separately from their sources, whereas we also found some misaligned sentences in Autshumato.

[6] http://www.polity.org.za/polity/ss/constitution/
[7] http://wordproject.org/
[8] http://ebible.org/
[9] http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/
[10] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/ems/support/split-sentences.perl
[11] http://mokk.bme.hu/en/resources/hunalign/

- removal of duplicate sentence pairs. This is partly since it seems that the Autshumato data set already contains parts of the constitution.

Proper tokenisation usually requires a list of abbreviations so that periods are not mistaken for full stops, keeping the abbreviations together as single tokens and making them easier to learn as single units of meaning. An example would be "e.g." which, under normal tokenisation rules, should become "e . g .". However, as an abbreviation, the form "e. g." seems to be preferred. A list of abbreviations containing this example would ensure that it is correctly written.

Apart from the list of English abbreviations bundled with TreeTagger, we used the same small list of Zulu abbreviations as in Wolff and Kotzé (2014).

Some experiments described in the next section make use of data sets where the Zulu words have been segmented in some way. To perform syllabification, a script was implemented where each vowel was regarded as signifying the end of a syllable. This was based on the open syllable assumption mentioned in section 2.

Also described in the next section is the creation of a set of baselines where Zulu words are split after a specified set of characters from the left. For example, if the number of characters was set at 3, the Zulu word "sawubona" would be written as "saw ubo na". Three separate data sets were created: for the number set at 3, 4 and 5.

New text to be translated would then be processed first in the same way as explained above. For example, if we wanted to apply the model that was trained on text where the Zulu text was split at every fourth character, the new Zulu text would first have to be split in the same way.

Our data set for target language modeling includes the target side of the bilingual data mentioned above (including unaligned segments, but excluding tuning and test data), as well as a large selection of English text that is freely available on the Web. We used a selection of three books from Project Gutenberg[12] relating to South Africa, as well as a 1% random sample of the English side of the English/French EU bookshop corpus (Skadiņš, Tiedemann, Rozis & Deksne, 2014)[13] and also a 1% random sample from the English part of the WMT13 2012 news corpus.[14] The text was tokenised and duplicates were removed, just as with the bitexts. The final size of this monolingual corpus is 7 843 797 tokens.

We now have a monolingual corpus which can be used for training the target language model. In addition, we also have a bitext that we can apply in the training, tuning and evalution stages of our translation models.

4 EXPERIMENTS

The purpose of our experiments is twofold. First, we compare the syllable-based approach against a set of baseline models. Secondly, we focus on optimising our models according to the data and tools that we have.

[12] http://www.gutenberg.org/

[13] http://opus.lingfil.uu.se/EUbookshop.php

[14] http://www.statmt.org/wmt13/translation-task.html

As mentioned in Section 2, an experiment consists of the training, possible tuning and evaluation of an MT system. Apart from target language modeling, all these steps are applied to the bitext. Typically, such a text is divided into a large training data set and smaller tuning and evaluation sets. Hence, our corpus composition is as follows: 90% for training, 5% for tuning, 5% for testing. The test set was divided into two, one of which we regard as a development test set and the other as a final test set, only to be used once.

As mentioned before, we use the SMT system Moses for our experiments. Based on the closed sourced Pharaoh system (Koehn, 2004a), it features a *decoder*—essentially a translation module, an optimised implementation meant to support effective use of the toolkit on less powerful systems and many other features. It also facilitates the use of external tools for essential tasks such as token alignment and target language modeling.

Moses' Experiment Management System (EMS) (Koehn, 2010a) is a tool which facilitates and simplifies setting up MT experiments. Among other things, it makes use of a large configuration file where one can specify most parameters. For our baseline models, we used the default values of most.

There are several language modeling tools available, such as SRILM (Stolcke, 2002), IRSTLM (Federico, Bertoldi & Cettolo, 2008), RandLM (Talbot & Osborne, 2007) and KenLM (Heafield, 2011; Heafield, Pouzyrevsky, Clark & Koehn, 2013). We opted to use KenLM because of its speed and low memory requirements. Some informal experiments suggested that it positively impacts translation quality.

We henceforth made use of the following configuration:

- For token alignment, we used two tools for comparison. The first is MGIZA++ (Gao & Vogel, 2008), which is a multi-threaded implementation of GIZA++ (Och & Ney, 2003), an unsupervised aligner that apply a number of different models developed by IBM, as well as a Hidden Markov Model, in a bidirectional way. The second is Fast Align (Dyer, Chahuneau & Smith, 2013), based on IBM Model 2 and also applied bidirectionally, which we found an attractive option because of its speed and simplicity.

- The maximum length of extracted phrases was 5 (the default) or 7 (in the last experiment below).

- For KenLM, we set the *order* parameter to either 5 (the default) or 6 (in the last experiment below). We used the same data and software for training the language model in each experiment.

- Ten iterations were used for MERT tuning (the default).

- All our evaluation scores are based on truecased (non-lowercased) versions of the text. Although the text in MT experiments are often lowercased in order to reduce the size of the lexicon (which may be an issue for very large corpora), this is in our case not a problem. Although this may lead to data sparsity problems for tokens that occur in both upper and lower case, evaluating on lowercased text also artificially improves evaluation results, which is something we attempted to avoid.

As mentioned in the background, both token alignment tools have an asymmetric bidirectional alignment approach. Each direction therefore produces a one-to-many alignment set that differs from the other. If both sets are simply combined, this results in many-to-many alignments. This is the *union* of the two alignment sets. The *intersection* is, of course, the set of one-to-one alignments that appear in both sets.

Additionally, there exists a selection of heuristics to combine the two alignment sets, which are implemented by the Moses toolkit. Each one starts out with the high-precision intersection of the bidirectional alignments, then adds links to neighbouring tokens in an iterative manner based on specific considerations. For example, *grow-diag* aligns immediate neighbours if both of them are unaligned and both of them appear in the union set.

In all of our configurations, we applied all of the possible token alignment techniques: the directional alignments and their combinations (*srctotgt*, *tgttosrc*, *intersect* and *union*) as well as the four heuristics (*grow*, *grow-diag*, *grow-diag-final* and *grow-diag-final-and*).

Our main baseline is the word-based MT model. This would be the default approach in PBSMT. One of our goals is to determine whether or not the syllable-based approach is significantly better than the normal, word-based approach. We also devised another set of baseline models, where each word is segmented at every n-th character, counting characters from left to right, where n ranges from 3 to 5 for each model. The idea behind this is to prove that syllables are better constituents of meaning than just any selection of strings of a specified length, lending strength to our hypothesis that it is a useful approximation of morphological analysis, at least for the task of applying these constituents in SMT. Finally, we apply Morfessor 2.0 (Virpioja, Smit, Grönroos & Kurimo, 2013) as an additional baseline, using all default settings. The model was trained on the Zulu side of the same training data set as for our MT experiments.

Using MGIZA++, our baseline model configuration and the aforementioned selection of text data, we trained models for each of the eight token alignment methods and tuned them with MERT. We then repeated the exercise on the same selection of data but where Zulu words have been segmented into syllables. The English output was evaluated using the BLEU, METEOR and TER metrics (Table 2). For the sake of brevity, we limit our results to the models with the best BLEU scores. We include evaluation results using Google Translate[15] as a point of reference, although the comparison is not entirely useful, as Google might have used some or even all of the development test data for training, and almost definitely uses a substantially larger language model of English.

Figure 2 shows a bar chart for all of the BLEU scores. It is striking that the syllable-based model outperforms the baseline for all alignment approaches but for the *grow-diag-final* heuristic. For the METEOR metric, the difference is even more pronounced, where the syllable-based approach wins on every occasion.

We also experimented with the aforementioned alignment tool, Fast Align. Its main attraction is its great speed, and we also found the fact that the EMS facilitates its seamless integration quite convenient. Our next step was to test its performance against MGIZA++ using the syllable-based model. Figure 3 shows the BLEU results against all token alignment approaches. Here, MGIZA++ only does better on lower performing models, while on the rest, Fast Align is the clear winner. Again,

[15] Evaluated on 29 April 2015 using translate.google.com.

Table 2: BLEU / METEOR / TER scores for best syllable-based (*intersection*) and best word-based translation systems (*tgttosrc*) using MGIZA++ and our baseline configuration (order 5, maximum phrase length 5). Note that as an error-rate measure, lower TER scores are better.

Model	**BLEU**	**METEOR**	**TER**
Syllables (intersect)	**32.82**	0.30	0.59
Words (tgttosrc)	28.54	0.27	0.64
Google Translate	31.49	**0.31**	**0.55**

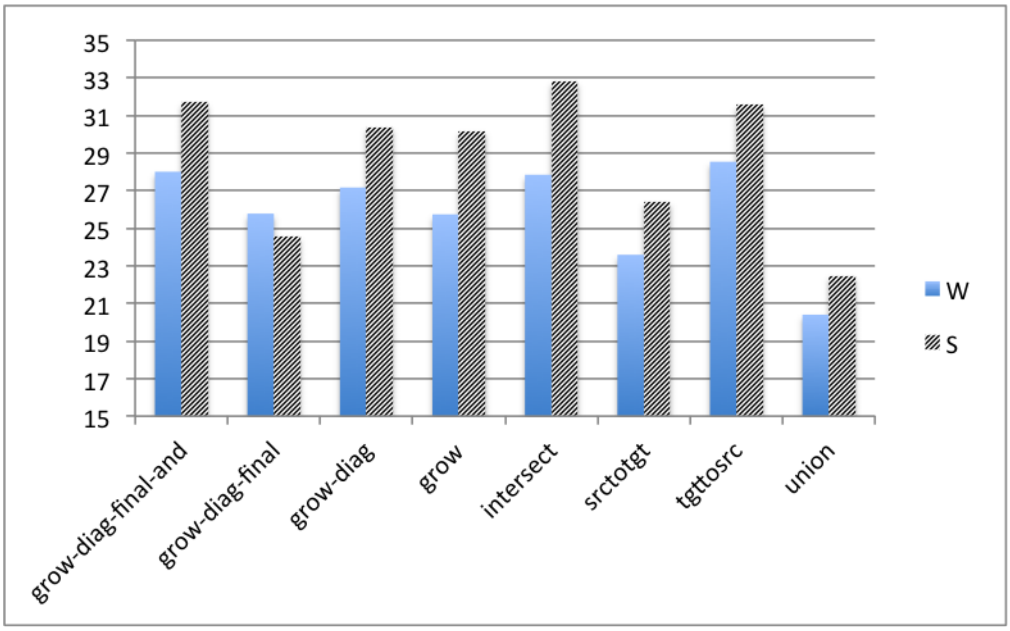

Figure 2: Comparison of BLEU scores for different token alignment approaches using MGIZA++ on words (W) vs. text where the Zulu words are split into syllables (S).

METEOR displays very similar results, while Fast Align also does better on all but *srctotgt* (source to target) on the TER metric. We found that Fast Align also outperformed MGIZA++ on the word-based models. We therefore decided to use the latter as a new baseline and to only use Fast Align for the rest of our experiments.

Since Zulu has been split into syllables, the maximum phrase length of 5 is probably insufficient to properly model many Zulu phrases, and even longer words. Therefore, we experimented with specifying an increased maximum phrase length of the translation model. Similarly, we also increased the phrase length that can be handled by the target language model (the *order* of the model). With each increase, scores clearly improved. We eventually settled on a maximum phrase length of 7 and an order of 6.[16] BLEU, METEOR and TER scores all improved, as Table 3 suggests.

[16] With its default build configuration, KenLM does not allow for an order of more than 6 tokens.

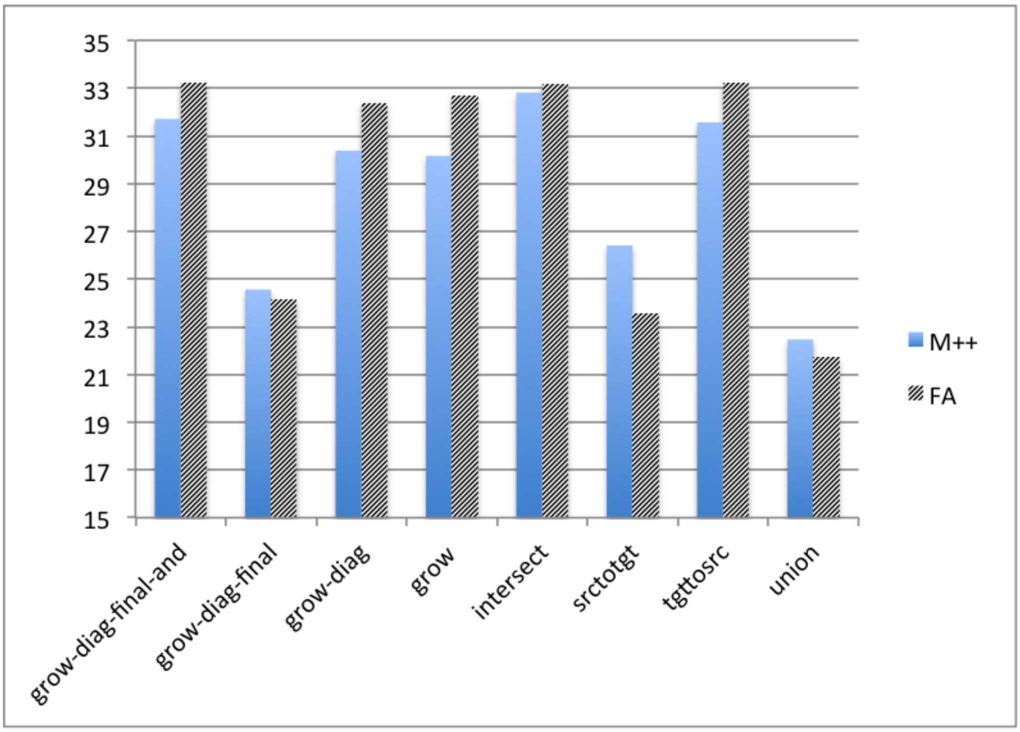

Figure 3: Comparison of BLEU scores for different alignment approaches using MGIZA++ (M++) and Fast Align (FA) on syllabified text.

Table 3: Demonstrating the improvement brought about by the best syllable-based model according to the standard configuration (*gdfa*, which refers to the heuristic *grow-diag-final-and*) when changing the maximum phrase length from 5 to 7 and the language model order from 5 to 6 (5–5 to 7–6). We also show the improvement leading to the best 7–6 model (*intersect*).

Model	**BLEU**	**METEOR**	**TER**
gdfa (5-5)	33.24	0.30	0.58
gdfa (7-6)	**34.96**	**0.31**	**0.56**
intersect (5-5)	33.20	0.30	0.59
intersect (7-6)	**35.32**	**0.31**	**0.56**

Figure 4 shows the best BLEU scores from each of our configurations. On the left side are the word-based models, followed by the character n-gram baselines, where each Zulu word has been divided after the stated number of characters, from left to right. This is followed by the syllable-based models and finally by the Morfessor models. All models have been tuned by MERT.

Here we can clearly see the increase in performance with each configuration, apart from the 5 character model which does slightly worse than the 4-character model. The syllable-based approach outperforms all but the Morfessor models, which are clearly better. An interesting observation is that

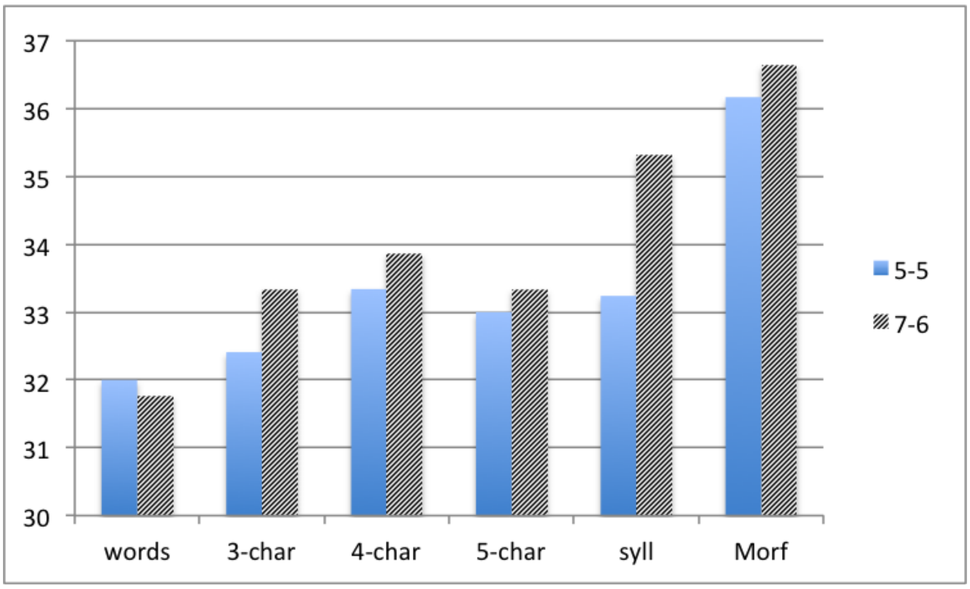

Figure 4: Comparison of best BLEU scores for all baselines as well as the syllable-based approach, with all models tuned. Note that different word alignment methods may have been used to train the models, depending on which was better for that particular configuration. "char" refers to the character-based approach, "syll" to the syllable-based approach and "Morf" refers to Morfessor.

the syllable-based models also clearly benefit the most from the optimisation. It is not clear why the "improved" word-based model fares worse.

Again, the METEOR results are very similar, with the optimised syllable-based model improving as much over the rest of the baselines apart from Morfessor. The 7–6 word-based model is now slightly better, although when inspecting the score differences for all heuristics, differences remain negligible. The strongest baseline (bar Morfessor) remains 4-char, but only by a slight margin. 3-char is slightly better than 4-char according to the TER metric, but other than that there is nothing interesting to report, and so we omit displaying the results here.

5 ANALYSIS OF SIGNIFICANCE

In this section, we perform analyses of significance with respect to our best models and how they compare against optimised versions of the baseline models. We determine whether or not we can reject the following null hypotheses:

- that the best syllable-based model (syll7–6) is not significantly better than the best word-based model (word5–5) (note that word5–5 has a better BLEU score than word7–6)
- that syll7–6 is not significantly better than the syllable-based model using the standard configuration (syll5–5)

- that syll7–6 is not significantly better than the best 4-gram character based model (4-char7–6)
- that syll7–6 is not significantly better than the best 5-gram character based model (5-char7–6)
- that the best Morfessor model (Morf7–6) is not significantly better than syll7–6
- that the standard Morfessor model (Morf5–5) is not significantly better than syll7–6

For each approach, we only consider the best performing model according to the BLEU metric and using Fast Align. Thus, the models that we compare are:

- syll5–5: *grow-diag-final-and*
- syll7–6: *intersect*
- word5–5: *intersect*
- 4-char7–6: *intersect*
- 5-char7–6: *intersect*
- Morf5–5: *intersect*
- Morf7-6: *grow*

We apply the paired bootstrap resampling method (Koehn, 2004b) to investigate the statistical significance of the improvements. We use the scripts published by ARK[17] for this purpose.

The method creates 1000 samples with replacement from the evaluation set, and compares the BLEU score for competing systems on each sample, keeping count of the winning system in each case. Apart from providing a better view on the statistical validity of the results, it also offers a way to address issues arising from small evaluation sets—not a real concern in our case with more than 1 500 segments used in evaluation.

The syll7–6 model is a significant improvement over word5–5, 4-char7–6 and 5-char7–6 at $p < 0.001$, $p < 0.05$ and $p < 0.001$ respectively. Optimising the syllable-based model from syll5–5 to syll7–6 brings about a great improvement at a significance of $p < 0.001$. Finally, Morf7–6 is significantly better than syll7–6 at $p < 0.001$, and even Morf5–5 is better than syll7–6 at $p < 0.05$. Improving Morf5–5 to Morf7–6 is also significant at $p < 0.05$.

We can therefore reject all of the abovementioned null hypotheses at the p values stated.

[17] http://www.ark.cs.cmu.edu/MT/

6 QUALITATIVE EVALUATION

One of the reasons for decomposition of the complex source text words in Zulu is that we hope to reduce the size of the lexicon, and thereby reduce the occurrence of *out-of-vocabulary words*—words that are completely unrecognised by the system and therefore have to be omitted or passed through untranslated. The baseline using character *n*-grams should achieve this as well as the syllable-based approach. A quantitative evaluation as performed above alone does not tell us if this goal was achieved. A qualitative evaluation also offers insights into the strong and weak points of each approach. In this section, all examples are from the best configuration of each approach, with the exception of the best 5-5 syllable model which is also included.

Here we see an example where words, as well as character 4-grams and 5-grams, all suffer from out-of-vocabulary tokens.

Source text	...ngaphambi kwenyathelo olihlosile .
Words	...before the *kwenyathelo olihlosile* .
Character 4-grams	...before *olih* steps .
Character 5-grams	...before another *olihl* steps .
Syllables 5-5	...before the action .
Syllables 7-6	...before you have measures .
Reference	...before your intended action .

Here is an example where the word model completely fails and many of the models transfer the number incorrectly (18 instead of 15). While the syllable models transfer the number correctly, the resulting translations are still not ideal.

Source text	...uma uneminyaka engaphezu kwengu-15 .
Words	...if *uneminyaka* than *kwengu-15* .
Character 3-grams	...once held over 18 years of age .
Character 4-grams	...if you are over 18 years of age .
Character 5-grams	...if behold 18 years of age .
Syllables 5-5	...if years was higher than 15 .
Syllables 7-6	...if years on 15 .
Morfessor	...if over 18 years of age 15 .
Reference	...if you are over the age of 15 .

The fine segmentation resulting from syllabification causes problems with certain longer phrases, such as numerals. This was noted as a weakness of the syllable-based approach (Wolff & Kotzé, 2014).

Here is an example where the 5-syllable model performs quite badly, but the 7-syllable model performs better, even though it is still not correct. (None of the systems translated this correctly.)

Source text	ayengamakhulu amabili namashumi ayisishiyagalombili nane .
Words	two hundred and eighty years .
Syllables (5-5)	three hundred thousand and two hundred and fifty eight years.
Syllables (7-6)	two hundred and fifty eight years .
Morfessor	two hundred and eighty .
Reference	two hundred and eighty-four .

We close this section with an example containing a long term *isivumelwano esenziwa phambi komendo* (English: antenuptial contract).

Source text	Ikhophi yesivumelwano esenziwa phambi komendo ...
Words	A copy of an ...
Character 4-grams	A copy of an antenuptial contract ...
Character 5-grams	A copy of an antenuptial ...
Syllables (5-5) and (7-6)	A copy of an agreement that was made before the marriage ...
Morfessor	A copy of contract ...
Reference	Copy of antenuptial contract ...

While the output from the syllable models is not a perfect translation of the term, the complexity involved here can be understood considering that it is a four-word term in Zulu, and contains 15 syllables for the syllable-based approaches—far exceeding the phrase length considered here (5 and 7). The very literal translation provides good semantic transfer for the purpose of getting a gist of the Zulu source text. The 4-gram and 5-gram models happened to generate the word "antenuptial" (in one case without "contract"), which indicates that the correct translation is possible from the training data in principle, but that the current models are not guaranteed to generate them due to the stochastic nature of the software and the long phrase length required.

7 DISCUSSION AND FUTURE WORK

Some interesting differences between the word-based and syllable-based approaches were discussed in Wolff and Kotzé (2014) and highlighted with some examples in the previous section. The additional baselines bring a few additional issues to the table. Firstly, we notice that all the models using sub-word tokens perform much better than the word-based approach. This confirms the importance of some form of sub-word handling when using Zulu as the source language in statistical machine translation. Specifically, we note that even the models based on character n-grams perform surprisingly well. With a phrase length of 5, 4-grams are competitive with the syllable-based approach.

With the phrase length increased to 7, the syllable-based approach becomes significantly better than the n-gram models. This confirms our hypothesis that syllabification is not merely successful because it reduces the size of the lexicon and the complexity of token alignment, but also because it models the language more accurately than the "blind" division into character n-grams. The fact that syllables divide a word at consistent points is crucial here, since the n-gram method could result in stems being segmented inconsistently, depending on which morphemes exist at the start of a word. For example, in “ngomile” (“I am thirsty”) and “somile” (“we are thirsty”), we expect to obtain two opportunities to learn about the translation of “-omile”. With trigrams, “-omile” is not divided consistently, since the first trigram will comprise “ngo” and “som” respectively, resulting in different tokens following this first trigram (“mil e” vs. “ile”). With the syllable-based approach, both these words will be segmented to end with “mi le”.

With the syllable-based method, the problem of the large lexicon is solved. All of the other methods also reduce the size of the lexicon substantially (Table 4). Reducing the lexicon provides more training opportunities per type (unique token) to learn the appropriate translations in different contexts. However, the method relies heavily on the phrase-based mechanics of the machine translation engine for correct lexical and semantic transfer. It might be that there is too much ambiguity in some cases with such a small vocabulary. Investigating the average token length gives us an idea of why the increased phrase length benefits the syllable approach as much: a phrase of length 5 only spans $2.131 \times 5 = 10.657$ non-whitespace tokens on average, while the other methods all span substantially longer (in the case of Morfessor, around 19.85 characters). It therefore seems as if some balance has to be sought between reducing the lexicon size, while not reducing token length too much.

Table 4: The lexicon size and average token length of each approach. The data is taken from the training data, after the Moses cleaning script has been applied. For reference, we include the same statistics for English.

Data set	Vocabulary	Average length
Character 3-grams	11905	2.557
Character 4-grams	33158	3.179
Character 5-grams	61536	3.765
Syllables	4623	2.131
Morfessor	19052	3.970
Words	128577	6.810
English (*Words*)	35018	4.194

The handling of proper names is particularly problematic, since the segmentation of a proper name into syllables is likely to cause problems unless all the relevant syllables can be transferred intact. It necessarily means that proper names that were not encountered during training will probably not be translated correctly, since ever smaller parts of the name will be considered, until (possibly) only a single syllable at a time will be translated—very likely resulting in an incorrect translation in the case of proper names. A similar matter was discussed in Wolff and Kotzé (2014) with regards to English loan words in the Zulu source text.

The issue of proper names, or more generally named entities of different kinds, is a known problem in machine translation. The nature of the failures is just different in the syllable-based method. It is possible that detection of named entities and loan words could help in avoiding these pitfalls by simply letting them pass through without syllabification. This would in effect enlarge the vocabulary by adding these detected entries as tokens of their own without finer segmentation.

The small lexicon and the very fine division do mean that multi-word terms comprising of many syllables are less likely to be handled correctly in a system that considers a maximum phrase length that is shorter than the multi-word term. It is not simply a matter of just enlarging the phrase length in the MT system—it severely impacts training and processing time, as well as system requirements in terms of memory and storage space. While the language pair considered here will usually have less training data available than for mainstream languages in MT research, the system requirements do not scale linearly in terms of the phrase length, so this provides only a little bit of extra room (as we did here in this study by increasing the default length of 5 to 7). It might therefore be meaningful to look for a middle ground where the lexicon is allowed to become a bit bigger so that segmentation is not quite as fine-grained as described here, thereby hopefully allowing the engine to model longer-distance phenomena more successfully.

Combining the different approaches should be investigated in the future, as this could hold promise of a way to combine their respective strong points.

8 CONCLUSION

We set out to find a way of handling the complexity of the Zulu writing system in machine translation from Zulu to English. The syllable-based approach performs significantly better than a simple word-based baseline, as do other baselines based on character *n*-grams and automatic morphological segmentation. The improvement of the syllable-based approach occurs despite having a small window on the text at a time in the phrase-based approach used here.

We also found that there was still quite some room for optimisation of such a system by different choices of alignment software, the size of the phrase table, as well as the order of the language model. The syllable-based approach seems to benefit more from the increase in phrase length and the order of the language model compared to the approaches based on *n*-grams.

The syllabification changes the way that the language is represented in its textual form, and this had consequences for which alignment method is most suitable. We found the *tgttosrc*, *intersect* and *grow-diag-final-and* alignment methods to be particularly effective when using the syllable-based approach—with either of the two alignment tools that we applied. To a large extent the other approaches also performed well with the *intersect* alignment method.

Our best syllable-based model (*grow-diag-final-and*) obtains a BLEU score of 33.24, an improvement of 2 BLEU points (3.8% improvement) over the word-based baseline with the highest BLEU score (*intersect*, 32). With increases in phrase length and language model order, we could further improve the BLEU score to 35.32, a further 2.08 BLEU points (6.3%). METEOR and TER scores show a similar improvement over the word-based baseline. The performance of models based on *n*-grams approaches that of syllables with the shorter phrase length (5), but with a longer phrase length and a higher language model order, the syllable-based approach is significantly better. The model based on segmentation by Morfessor outperformed all other models. This is in contrast to previous experiments in the Nordic languages (Virpioja et al., 2007) and Czech to English (Virpioja et al., 2010) where an approach with Morfessor did not improve BLEU scores. However, from inspection we also found far less out-of-vocabulary words; not only with Morfessor, but with all the sub-lexical approaches.

The good results of Morfessor suggest that morphological segmentation of Zulu is preferable. This makes intuitive sense as morphemes by definition approximate semantically distinct units within the word. The fact that syllabification is prone to oversegmentation but seemingly preserves morphological boundaries for the most part seems to correlate with the fact that it fares worse than Morfessor but is still significantly better than sub-lexical segmentation at fixed character intervals without regard for its internal structure.

We have presented an approach for improving statistical machine translation to English that has the potential to be applied to a large selection of languages within Africa with a similar syllabic structure. Importantly, most of these languages are under-resourced and may benefit from such an approach. In our case, analysis suggests that the scores are significantly better than a word-based model, but doesn't compare favourably to an approach using a state-of-the-art automatically induced morphological analyser. Additionally, we have also shown that a simple optimisation procedure can lead to significant improvements. We hope that the results of this paper will stimulate further research into the machine translation of under-resourced languages.

ACKNOWLEDGMENTS

We thank our colleague Dr Loïc Dugast and the anonymous reviewers for their valuable feedback. This research was supported in part by funding from the Academy of African Languages and Science Strategic Project of the University of South Africa. We are grateful for the UNISA High Performance Computing (HPC) facilities that were used in this work.

References

Banerjee, S. & Lavie, A. (2005, June). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization* (pp. 65–72). Ann Arbor, Michigan: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/W05-0909

Brown, P. F., Cocke, J., Pietra, S. A. D., Pietra, V. J. D., Jelinek, F., Lafferty, J. D., ... Roossin, P. S. (1990, June). A statistical approach to machine translation. *Computational Linguistics*, *16*(2), 79–85. Retrieved from http://dl.acm.org/citation.cfm?id=92858.92860

Chen, S. F. & Goodman, J. (1999). An empirical study of smoothing techniques for language modeling. *Computer Speech & Language*, *13*(4), 359–393. http://dx.doi.org/10.1006/csla.1999.0128

De Pauw, G., Wagacha, P. W. & Schryver, G.-M. (2011). Exploring the SAWA corpus: Collection and deployment of a parallel corpus English-Swahili. *Language Resources and Evaluation*, *45*(3), 331–344. http://dx.doi.org/10.1007/s10579-011-9159-7

Dyer, C., Chahuneau, V. & Smith, N. A. (2013, June). A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 644–648). Atlanta, Georgia: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/N13-1073

Federico, M., Bertoldi, N. & Cettolo, M. (2008). IRSTLM: An open source toolkit for handling large scale language models. In *INTERSPEECH* (pp. 1618–1621). ISCA.

Gao, Q. & Vogel, S. (2008). Parallel implementations of word alignment tool. In *Software engineering, testing, and quality assurance for natural language processing* (pp. 49–57). SETQA-NLP '08. Columbus, Ohio: Association for Computational Linguistics. http://dx.doi.org/10.3115/1622110.1622119

Heafield, K. (2011, July). KenLM: Faster and smaller language model queries. In *Proceedings of the EMNLP 2011 Sixth Workshop on Statistical Machine Translation* (pp. 187–197). Edinburgh, Scotland, United Kingdom. Retrieved from http://kheafield.com/professional/avenue/kenlm.pdf

Heafield, K., Pouzyrevsky, I., Clark, J. H. & Koehn, P. (2013, August). Scalable modified Kneser-Ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics* (pp. 690–696). Retrieved from http://kheafield.com/professional/edinburgh/estimate_paper.pdf

Hermanson, E. A. (2002). A brief overview of Bible translation in South Africa. *Acta Theologica, Supplementum 2*, *22*(1), 6–18. http://dx.doi.org/10.4314/actat.v22i1.5451

Jurafsky, D. & Martin, J. (2009). *Speech and language processing (2nd edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc.

Koehn, P. (2004a). Pharaoh: A beam search decoder for phrase-based statistical machine translation models. In *Machine Translation: From Real Users to Research, 6th Conference of the Association for Machine Translation in the Americas, AMTA 2004, Washington, DC, USA, September 28-October 2, 2004, Proceedings* (pp. 115–124). http://dx.doi.org/10.1007/978-3-540-30194-3_13

Koehn, P. (2004b, July). Statistical significance tests for machine translation evaluation. In D. Lin & D. Wu (Eds.), *Proceedings of EMNLP 2004* (pp. 388–395). Barcelona, Spain: Association for Computational Linguistics.

Koehn, P. (2010a). An experimental management system. *Prague Bull. Math. Linguistics*, *94*, 87–96. http://dx.doi.org/10.2478/v10108-010-0023-5

Koehn, P. (2010b). *Statistical machine translation* (1st). New York, NY, USA: Cambridge University Press.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., ... Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions* (pp. 177–180). ACL '07. Prague, Czech Republic: Association for Computational Linguistics. http://dx.doi.org/10.3115/1557769.1557821

Kotzé, G. & Wolff, F. (2014). Experiments with syllable-based English-Zulu alignment. In *Proceedings of the SaLTMiL Workshop on Free/open-source Language Resources for the Machine Translation of Less-resourced Languages (at LREC 2014)* (pp. 7–11).

McKellar, C. A. & Groenewald, H. J. (2012). Frequency-based data selection for statistical machine translation with scarce resources. In H. S. Ndinga-Koumba-Binza & S. E. Bosch (Eds.), *Language science and language technology in Africa: Festschrift for Justus C Roux* (pp. 271–290). Stellenbosch: Sun Media.

Nakov, P. & Tiedemann, J. (2012). Combining word-level and character-level models for machine translation between closely-related languages. In *The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the conference, July 8-14, 2012, Jeju Island, Korea - Volume 2: Short Papers* (pp. 301–305). Retrieved from http://www.aclweb.org/anthology/P12-2059

Och, F. J. (2003). Minimum Error Rate Training in statistical machine translation. In *Proceedings of the 41st annual meeting on Association for Computational Linguistics - volume 1* (pp. 160–167). ACL '03. Sapporo, Japan: Association for Computational Linguistics. http://dx.doi.org/10.3115/1075096.1075117

Och, F. J. & Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, *29*(1), 19–51. http://dx.doi.org/10.1162/089120103321337421

Papineni, K., Roukos, S., Ward, T. & Zhu, W.-J. (2002). BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (pp. 311–318). ACL '02. Philadelphia, Pennsylvania: Association for Computational Linguistics. http://dx.doi.org/10.3115/1073083.1073135

Pretorius, L. & Bosch, S. E. (2003). Finite-state computational morphology: An analyzer prototype for Zulu. *Machine Translation*, *18*(3), 195–216. http://dx.doi.org/10.1007/s10590-004-2477-4

Quasthoff, U., Bosch, S. & Goldhahn, D. (2014). Morphological analysis for less-resourced languages: Maximum affix overlap applied to Zulu. In *Workshop on Collaboration and Computing for Under-resourced Languages in the Linked Open Data Era (LREC), Reykjavik*.

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*. Manchester, UK.

Skadiņš, R., Tiedemann, J., Rozis, R. & Deksne, D. (2014, May). Billions of parallel words for free: Building and using the EU bookshop corpus. In N. Calzolari, K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, ... S. Piperidis (Eds.), *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC '14)*. Reykjavik, Iceland: European Language Resources Association (ELRA).

Snover, M., Dorr, B., Schwartz, R., Micciulla, L. & Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. In *Proceedings of Association for Machine Translation in the Americas* (pp. 223–231).

Spiegler, S., Golénia, B., Shalonova, K., Flach, P. A. & Tucker, R. C. F. (2008). Learning the morphology of Zulu with different degrees of supervision. In A. Das & S. Bangalore (Eds.), *SLT* (pp. 9–12). IEEE. http://dx.doi.org/10.1109/slt.2008.4777827

Spinner, P. (2011). Review article: Second language acquisition of Bantu languages: A (mostly) untapped research opportunity. *Second Language Research*, *27*(3), 418–430. http://dx.doi.org/10.1177/0267658310376217. eprint: http://slr.sagepub.com/content/27/3/418.full.pdf+html

Stolcke, A. (2002, November). SRILM — An extensible language modeling toolkit. In *Proceedings of International Conference on Spoken Language Processing* (pp. 257–286).

Talbot, D. & Osborne, M. (2007). Randomised language modelling for statistical machine translation. In J. A. Carroll, A. van den Bosch & A. Zaenen (Eds.), *ACL*. The Association for Computational Linguistics.

Tiedemann, J. (2009). Character-based PSMT for closely related languages. In *Proceedings of 13th Annual Conference of the European Association for Machine Translation EAMT09* (pp. 12–19).

Van Niekerk, D. (2014). Exploring unsupervised word segmentation for machine translation in the South African context. In M. Puttkammer & R. Eiselen (Eds.), *Proceedings of the 2014 PRASA, RobMech and AfLaT International Joint Symposium* (pp. 202–206). Cape Town, South Africa: PRASA. Retrieved from http://www.prasa.org/proceedings/2014/prasa2014-35.pdf

Varga, D., Németh, L., Halácsy, P., Kornai, A., Trón, V. & Nagy, V. (2007). Parallel corpora for medium density languages. In N. Nicolov, K. Bontcheva, G. Angelova & R. Mitkov (Eds.), *Recent Advances in Natural Language Processing IV. Selected papers from RANLP 2005* (Vol. 292, pp. 247–258). 'Current Issues in Linguistic Theory'. John Benjamins. http://dx.doi.org/10.1075/cilt.292.32var

Virpioja, S., Smit, P., Grönroos, S.-A. & Kurimo, M. (2013). *Morfessor 2.0: Python implementation and extensions for Morfessor baseline*. Aalto University publication series Science + Technology, 25/2013. Aalto University, Helsinki. Retrieved from https://aaltodoc.aalto.fi/handle/123456789/11836

Virpioja, S., Väyrynen, J. J., Creutz, M. & Sadeniemi, M. (2007). Morphology-aware statistical machine translation based on morphs induced in an unsupervised manner. In *Proceedings of the MT Summit XI* (pp. 491–498).

Virpioja, S., Väyrynen, J., Mansikkaniemi, A. & Kurimo, M. (2010). Applying morphological decomposition to statistical machine translation. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR* (pp. 195–200). WMT '10. Uppsala, Sweden: Association for Computational Linguistics. Retrieved from http://dl.acm.org/citation.cfm?id=1868850.1868879

Wolff, F. & Kotzé, G. (2014). Experiments with syllable-based Zulu-English machine translation. In M. Puttkammer & R. Eiselen (Eds.), *Proceedings of the 2014 PRASA, RobMech and AfLaT International Joint Symposium* (pp. 217–222). Cape Town, South Africa: PRASA.

http://dx.doi.org/10.18489/sacj.v0i57.323