Optimal Word Segmentation for Neural Machine Translation into Dravidian Languages

Dhar, Prajit; Bisazza, Arianna; van Noord, Gertjan

Published in:
Proceedings of the 8th Workshop on Asian Translation (WAT2021)

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2021

Link to publication in University of Groningen/UMCG research database

Citation for published version (APA):

Copyright
Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license. More information can be found on the University of Groningen website: https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment.

Take-down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): http://www.rug.nl/research/portal. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

Download date: 18-10-2023
Optimal Word Segmentation for Neural Machine Translation into Dravidian Languages

Prajit Dhar Arianna Bisazza Gertjan van Noord
University of Groningen
{p.dhar, a.bisazza, g.j.m.van.noord}@rug.nl

Abstract

Dravidian languages, such as Kannada and Tamil, are notoriously difficult to translate by state-of-the-art neural models. This stems from the fact that these languages are morphologically very rich as well as being low-resourced. In this paper, we focus on subword segmentation and evaluate Linguistically Motivated Vocabulary Reduction (LMVR) against the more commonly used SentencePiece (SP) for the task of translating from English into four different Dravidian languages. Additionally we investigate the optimal subword vocabulary size for each language. We find that SP is the overall best choice for segmentation, and that larger subword vocabulary sizes lead to higher translation quality.

1 Introduction

Dravidian languages are an important family of languages spoken by about 250 million of people primarily located in Southern India and Sri Lanka (Steever, 2019). Kannada (KN), Malayalam (MA), Tamil (TA) and Telugu (TE) are the four most spoken Dravidian languages with approximately 47, 34, 71 and 79 million native speakers, respectively. Together, they account for 93% of all Dravidian language speakers. While Kannada, Malayalam and Tamil are classified as South Dravidian languages, Telugu is a part of South-Central Dravidian languages. All four languages are SOV (Subject-Object-Verb) languages with free word order. They are highly agglutinative and inflectionally rich languages. Additionally, each language has a different writing system. Table 1 presents an English sentence example and its Dravidian-language translations.

The highly complex morphology of the Dravidian languages under study is illustrated if we compare translated sentence pairs. The analysis of our parallel datasets (section 4.1, Table 3) shows for instance that an average English sentence contains almost ten times as many words as its Kannada equivalent. For the other three languages, the ratio is a bit smaller but the difference with English remains considerable. This indicates why it is important to consider word segmentation algorithms as part of the translation system.

In this paper we describe our work on Neural Machine Translation (NMT) from English into the Dravidian languages Kannada, Malayalam, Tamil and Telugu. We investigated the optimal translation settings for the pairs and in particular looked at the effect of word segmentation. The aim of the paper is to answer the following research questions:

- Does LMVR, a linguistically motivated word segmentation algorithm, outperform the purely data-driven SentencePiece?
- What is the optimal subword dictionary size for translating from English into these Dravidian languages?

In what follows, we review the relevant previous work (Sect. 2), introduce the two segmenters (Sect. 3), describe the experimental setup (Sect. 4), and present our answers to the above research questions (Sect. 5).

2 Previous Work

2.1 Translation Systems

Statistical Machine Translation One of the earliest automatic translation systems for English into a Dravidian language was the English→Tamil system by Germann (2001). They trained a hybrid rule-based/statistical machine translation system that was trained on only 5k English-Tamil parallel sentences. Ramasamy et al. (2012) created SMT systems (phrase-based and hierarchical) which were trained on a dataset of 190k parallel
He was born in Thirukkuvalai village in Nagapattinam District on 3rd June, 1924.

Table 1: Example sentence in English along with its translation and transliteration in the four Dravidian languages.

<table>
<thead>
<tr>
<th>Language</th>
<th>Sentence</th>
<th>Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>EN</td>
<td>He was born in Thirukkuvalai village in Nagapattinam District on 3rd June, 1924.</td>
<td>ஆவர் 1924-ஆம் ஆண்டில் நாகப்பட்டினம் மாவட்டத்தில் திருக்குவளையில் 3-ம் மாதம் ஆண்டு ஜனித் தொண்டியார்.</td>
</tr>
<tr>
<td>KN</td>
<td>அவர் 1924ஆம் ஆண்டில் நாகப்பட்டினம் மாவட்டத்தில் திருக்குவளை கிராமத்தில் துணை ஜனவரியின் தினம் ஜூன் 3ஆம் தேதி பிறந்தார்.</td>
<td></td>
</tr>
<tr>
<td>ML</td>
<td>1924-ஆம் ஆண்டு மாதம் திருக்குவளை கிராமத்தில் துணை ஜனவரியின் தினம் ஜூன் 3ஆம் தேதி பிறந்தார்.</td>
<td></td>
</tr>
<tr>
<td>TA</td>
<td>ஆயனா 1924ஆம் ஆண்டில் திருக்குவளை கிராமத்தில் ஜனவரி 3ஆம் தேதி பிறந்தார்.</td>
<td></td>
</tr>
<tr>
<td>TE</td>
<td>1924ஆம் ஆண்டு ஜூன் 3ஆம் வாகனம் இவர் திருக்குவளை கிராமத்தில் துணை ஜனவரியின் தினம் பிறந்தார்.</td>
<td></td>
</tr>
</tbody>
</table>

Neural Machine Translation On the neural machine translation (NMT) side, there have been a handful of NMT systems trained on English → Tamil. On the aforementioned Indic languages multilingual tasks of WAT-2018, the Phrasal-based SMT system of Ojha et al. (2018) with a BLEU score of 30.53.

Multilingual NMT Since 2018 several studies have presented multilingual NMT systems that can handle English → Malayalam, Tamil and Telugu translation (Dabre et al., 2018; Choudhary et al., 2020; Ojha et al., 2018; Sen et al., 2018; Yu et al., 2020; Dabre and Chakrabarty, 2020). In particular, Sen et al. (2018) presented results where the BLEU score improved when comparing monolingual and multilingual models. Conversely, Yu et al. (2020) found that NMT systems that were multi-way (Indic ↔ Indic) performed worse than English ↔ Indic systems.

3 Subword Segmentation Techniques

Prior to the emergence of subword segmenters, translation systems were plagued with the issue of
out-of-vocabulary (OOV) tokens. This was particularly an issue for translations involving agglutinative languages such as Turkish (Ataman and Federico, 2018) or Malayalam (Manohar et al., 2020). Various segmentation algorithms were brought forward to circumvent this issue and in turn, improve translation quality.

Perhaps the most widely used algorithm in NMT to date is the language-agnostic Byte Pair Encoding (BPE) by Sennrich et al. (2016). Initially proposed by Gage (1994), BPE was repurposed by Sennrich et al. (2016) for the task of subword segmentation, and is based on a simple principle whereby pairs of character sequences that are frequently observed in a corpus get merged iteratively until a predetermined dictionary size is attained. In this paper we use a popular implementation of BPE, called SentencePiece (SP) (Kudo and Richardson, 2018).

While purely statistical algorithms are able to segment any token into smaller segments, there is no guarantee that the generated tokens will be linguistically sensible. Unsupervised morphological induction is a rich area of research that also aims at learning a segmentation from data, but in a linguistically motivated way. The most well-known example is Morfessor with its different variants (Creutz and Lagus, 2002; Kohonen et al., 2010; Grönroos et al., 2014). An important obstacle to applying Morfessor to the task of NMT is the lack of a mechanism to determine the dictionary size. To address this, Ataman et al. (2017) proposed a modification of Morfessor FlatCat (Grönroos et al., 2014), called Linguistically Motivated Vocabulary Reduction (LMVR). Specifically, LMVR imposes an extra condition on the cost function of Morfessor Flatcat so as to favour vocabularies of the desired size. In a comparison of LMVR to BPE, Ataman et al. (2017) reported a +2.3 BLEU improvement on the English-Turkish translation task of WMT18.

Given the encouraging results reported on the agglutinative Turkish language, we hypothesise that translation into Dravidian languages may also benefit from a linguistically motivated segmenter, and evaluate LMVR against SP across varying vocabulary sizes.

4 Experimental Setup

4.1 Training Corpora

The parallel training data is mostly taken from the datasets available for the MultiIndicMT task from WAT 2021. If a certain dataset is not available from the MultiIndicMT training repository, we resorted to extract that dataset from OPUS (Tiedemann, 2012) or WMT20. Table 2 reports on the datasets that we used along with their domain and their source.

After extracting and cleaning the data (see below), approximately 8 million English tokens and their corresponding target language tokens are selected as our training corpora. We fixed the number of source tokens across language pairs in or-

<table>
<thead>
<tr>
<th>Name</th>
<th>Domain</th>
<th>Available in:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Kannada</td>
<td>Malayalam</td>
</tr>
<tr>
<td>Bible</td>
<td>Religion</td>
<td>18</td>
</tr>
<tr>
<td>ELRC</td>
<td>COVID-19</td>
<td>&lt;1</td>
</tr>
<tr>
<td>GNOME</td>
<td>Technical</td>
<td>&lt;1</td>
</tr>
<tr>
<td>JW300</td>
<td>Religion</td>
<td>70</td>
</tr>
<tr>
<td>KDE</td>
<td>Technical</td>
<td>1</td>
</tr>
<tr>
<td>NLPC</td>
<td>General</td>
<td>&lt;1</td>
</tr>
<tr>
<td>OpenSubtitles</td>
<td>Cinema</td>
<td>26</td>
</tr>
<tr>
<td>CVIT-PIB</td>
<td>Press</td>
<td>5</td>
</tr>
<tr>
<td>PMIndia</td>
<td>Politics</td>
<td>10</td>
</tr>
<tr>
<td>Tanzil</td>
<td>Religion</td>
<td>18</td>
</tr>
<tr>
<td>Tatoeba</td>
<td>General</td>
<td>&lt;1</td>
</tr>
<tr>
<td>Ted2020</td>
<td>General</td>
<td>&lt;1</td>
</tr>
<tr>
<td>TICO-19</td>
<td>COVID-19</td>
<td>&lt;1</td>
</tr>
<tr>
<td>Ubuntu</td>
<td>Technical</td>
<td>&lt;1</td>
</tr>
<tr>
<td>UFAL</td>
<td>Mixed</td>
<td>11</td>
</tr>
<tr>
<td>Wikimatrix</td>
<td>General</td>
<td>&lt;1</td>
</tr>
<tr>
<td>Wikititles</td>
<td>General</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 2: Composition of training corpora. The numbers indicate the relative size (in percentages) of the corresponding part for that language.
order to compare the efficacy of a segmentation technique across the languages without a size bias. Table 3 presents the statistics on the corpora for all language pairs. One takeaway from the table is that there is a very large difference in the token sizes between English and the Dravidian languages. On average, there are 6 to 9 times more tokens on the English side of a corpus than on its Dravidian language translation. This shows that all our Dravidian languages are morphologically very complex, but there are also important differences among them, with Kannada having the highest source/target ratio, considerably higher than the more widely studied Tamil language.

### 4.2 Pre-Processing

Sentence pairs with identical source and target sides, or with more than 150 tokens are removed. The target language texts are then normalized using the Indic NLP Library\(^1\). Afterwards, either SP\(^2\) or LMVR\(^3\) is used to segment both source and target sentences. To further reduce noise in the datasets, we discard sentence pairs with either (i) a target to source length ratio above 0.7 or (ii) a language match threshold below 85% according to the lang-id tool (Lui and Baldwin, 2011), and (iii) duplicate sentence pairs.

### 4.3 NMT Training

We developed our NMT systems using Fairseq (Ott et al., 2019). We adopt the Transformer-Base implementation (BASE) with a few modifications following the architecture setup of Philip et al. (2019) and Dhar et al. (2020). These modifications include: setting both encoder and decoder layers to 6, embedding dimensions to size 1024 and number of attention heads to 8. Training is performed using batches of 4k tokens, using a label-smoothed cross entropy loss. The hidden layers are of 1024 dimensions and layer normalization is applied before each encoder and decoder layer. Dropout is set to 0.001 and weight decay to 0.2. Our loss function is cross-entropy with label smoothing of 0.3. The models are trained for a maximum of 100 epochs with early stopping criterion set to 5.

### 4.4 Dictionary Size

The segmentation algorithms are trained on the training data described in Section 4.1. We experiment with the following subword dictionary sizes: 1k, 5k, 10k, 15k, 20k, 30k, 40k and 50k. In all experiments, we learn separate subword dictionaries for the source and target languages, for two reasons: (i) LMVR is a linguistically motivated morphology learning algorithm that models the composition of a word based on the transitions between different morphemes and their categories. Therefore, training jointly on two languages would not be a principled choice. (ii) Prior studies such as (Dhar et al., 2020) have reported better translation scores for English-Tamil using SP models that were separately trained on the source and target sides.

### 5 Results

The NMT systems are evaluated and tested on the official development and test sets, respectively from WAT21. These evaluation sets are sourced from the PMIndia dataset (Haddow and Kirefu, 2020). During validation, models are evaluated by BLEU on the segmented data, whereas final test scores are computed on the un-segmented and de-tokenized sentences (de-tokenization is performed with the Indic NLP library tool). In addition to BLEU (Papineni et al., 2002), we also report on CHRF score (Popović, 2015), which is based on character n-grams and is therefore more suitable to assess translation quality in morphologically complex languages.\(^4\) We report the macro-averaged

### Table 3: Approximate sizes (in thousands) of the parallel training corpora

<table>
<thead>
<tr>
<th>Target Language</th>
<th>Tokens(k)</th>
<th>EN Tokens(k)</th>
<th>Sentences(k)</th>
<th>Source/Target Token Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kannada</td>
<td>817</td>
<td>7791</td>
<td>361</td>
<td>9.53</td>
</tr>
<tr>
<td>Malayalam</td>
<td>1153</td>
<td>7973</td>
<td>458</td>
<td>6.91</td>
</tr>
<tr>
<td>Tamil</td>
<td>1171</td>
<td>7854</td>
<td>345</td>
<td>6.71</td>
</tr>
<tr>
<td>Telugu</td>
<td>1027</td>
<td>7872</td>
<td>385</td>
<td>7.67</td>
</tr>
</tbody>
</table>

\(^1\)http://anoopkunchukuttan.github.io/indic_nlp_library/

\(^2\)https://github.com/google/sentencepiece

\(^3\)https://github.com/d-ataman/lmvr

\(^4\)We compute BLEU scores with SacreBLEU (Post, 2018), and CHRF scores with chrF++.py https://github.com/
Table 4: BLEU and CHRF scores for English-to-X NMT, using different segmenters and varying subword vocabulary size. SP refers to the purely statistical SentencePiece segmenter, LMVR to Linguistically Motivated Vocabulary Reduction. Dictionary size refers to the size of both the source and target subword dictionaries. Rightmost columns show the Jaccard similarity (percentage) for the types and tokens from the segmenter outputs.

<table>
<thead>
<tr>
<th>Target Language</th>
<th>Dictionary Size</th>
<th>BLEU SP</th>
<th>CHRF SP</th>
<th>Jaccard Similarity (%) Types</th>
<th>Tokens</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kannada</td>
<td>1k</td>
<td>10.4</td>
<td>6.2</td>
<td>48.3</td>
<td>40.6</td>
</tr>
<tr>
<td></td>
<td>5k</td>
<td>13.0</td>
<td>5.9</td>
<td>50.2</td>
<td>40.7</td>
</tr>
<tr>
<td></td>
<td>10k</td>
<td>13.9</td>
<td>6.8</td>
<td>49.6</td>
<td>42.8</td>
</tr>
<tr>
<td>Malayalam</td>
<td>1k</td>
<td>8.1</td>
<td>8.8</td>
<td>47.4</td>
<td>46.1</td>
</tr>
<tr>
<td></td>
<td>5k</td>
<td>11.2</td>
<td>12.6</td>
<td>52.3</td>
<td>50.5</td>
</tr>
<tr>
<td></td>
<td>10k</td>
<td>14.6</td>
<td>15.9</td>
<td>55.3</td>
<td>50.5</td>
</tr>
<tr>
<td>Tamil</td>
<td>1k</td>
<td>10.4</td>
<td>8.1</td>
<td>48.3</td>
<td>45.7</td>
</tr>
<tr>
<td></td>
<td>5k</td>
<td>13.2</td>
<td>8.2</td>
<td>50.6</td>
<td>46.2</td>
</tr>
<tr>
<td></td>
<td>10k</td>
<td>15.6</td>
<td>10.0</td>
<td>51.8</td>
<td>48.7</td>
</tr>
<tr>
<td>Telugu</td>
<td>1k</td>
<td>5.3</td>
<td>11.8</td>
<td>40.7</td>
<td>45.9</td>
</tr>
<tr>
<td></td>
<td>5k</td>
<td>5.6</td>
<td>10.8</td>
<td>44.6</td>
<td>43.5</td>
</tr>
<tr>
<td></td>
<td>10k</td>
<td>6.2</td>
<td>12.8</td>
<td>45.4</td>
<td>45.6</td>
</tr>
</tbody>
</table>

SP clear winner for Kannada and Tamil: SP presented the highest BLEU and CHRF scores for Kannada and Tamil. When we compare the best systems for both SP and LMVR, large differences are observed. For Kannada differences of +6 BLEU and +7.4 are observed and for Tamil the differences are +14.9 for BLEU and +5.9 for CHRF.

Mixed results for Telugu and Malayalam: However, we find no clear winner for the other two languages. When observing only BLEU scores, LMVR appears to have the upper hand, with an improvement of +2.8 BLEU and +4.5 BLEU for Malayalam and Telugu, respectively. However the results are flipped when we look at the CHRF scores. SP systems here report higher scores, with
+3.5 improvement in Malayalam and +1.1 for Telugu. Given the morphological richness of our target languages, we take CHRF as the more reliable score, and conclude that the purely statistical segmenter SP is a better choice for translation into Dravidian languages in our setup.

**Larger dictionary sizes better:** When observing the effect of the dictionary size, we find that the size 50k gives the highest BLEU scores for Malayalam, Tamil and Telugu. This is in contrast with studies such as (Philip et al., 2019; Sennrich and Zhang, 2019) who suggest to use a smaller dictionary size for low-resource settings. For these language pairs, we see a steady increase in BLEU and CHRF as we increase the dictionary size. For Kannada, the best results are obtained for much smaller dictionary sizes, but in contrast with the other three languages, the differences between the scores for other dictionary sizes is much smaller. For instance, looking at the CHRF scores of SP, the numbers decrease from 48.3 to 46.0, whereas for instance for Malayalam, these numbers range from 47.4 to 63.6.

**Kannada hardest to translate:** When comparing more in general translation difficulty across target languages, Kannada appears to be the most challenging language by far. A possible explanation for this difference is the genre distribution of our datasets (cf. Table 2): While the test sets are from PMIndia (a mixture of background information, news and speeches), the majority of our Kannada training data consists of religion related texts. Another possible confounding factor is that we based our NMT configuration on prior work that focused only on English-Tamil (Philip et al., 2019; Dhar et al., 2020), and this may be sub-optimal for the other Dravidian languages despite the similar training data size.

## 6 Analysis

### 6.1 Different Subtokens generated

Table 4 presents the Jaccard similarity (JS) between the segmenter outputs between LMVR and SP. The outputs are either the types (dictionaries) or the tokens in the training sentences. A JS of 0 denotes that none of the subwords were the same in the sentences being compared, while a score of 100 denotes a complete match (i.e., they are identical). As visible from the scores, though there is some sharing of types between the segmenters (ranging from 9-17%), there is no such sharing of subwords in the training data, with a maximum JS score of only around 4% for the smallest dictionary sizes. In fact, these values reduce even further as the dictionary size are increased. For the largest dictionary size (50k), almost no subtoken sharing occurs.

### 6.2 Effect of Unknown Subwords

We carried out an analysis on the effect of unknown subwords found in the development set after the application of a given segmentation algorithm. We present these statistics in Figure 1. Few details stand out:

---

**High percentage of unknown subwords in Kannada with LMVR** While development sets encoded with SP reported the lowest percentage of unknowns, it is the complete opposite for the ones encoded with LMVR (0.2% vs 15% on average). This could have played a role in the lowest CHRF scores achieved by the LMVR systems on Kannada.

**LMVR sensitive to dictionary size** This is observed in particular for Kannada and Malayalam, where the increase in dictionary size leads to higher numbers of unknown subwords. Conversely for SP, increasing the dictionary size causes no major change in the number of unknowns found for these two languages. On the other hand, SP is more susceptible to the dictionary size for Tamil while Telugu, in general, does not present any such trends.

Overall we find no strong correlation between system performance and percentage of unknown subwords. By contrast, and quite surprisingly so, our best NMT systems for Malayalam, Tamil and Telugu are those with larger dictionary sizes and higher percentage of unknowns in the development set.

### 6.3 Effect of subword lengths

We also looked at the effect of the segmenter on the subword length. Given a language and segmenter, we calculate the average length of a subword (in characters) for the training sets. In Figure 2 we plot the distribution of the average subword lengths for all our settings. Few observations are apparent,

- For every language and dictionary size, LMVR results in shorter subwords. Taking dictionary size of 50k as an example, the dif-
Figure 1: Number of unknown tokens (in percentages) in the development set vs Dictionary size for each language and segmentation type. Also systems that reported the lowest and highest CHRF scores (on the development set) for each language and segmentation are marked.

7 Conclusion

We presented our work on Neural Machine Translation from English into four Dravidian languages (Kannada, Malayalam, Tamil and Telugu). Several experiments were carried out to find out whether a linguistically motivated subword segmenter (LMVR) is more suitable than a purely statistical one (SentencePiece) for translating into the morphologically complex Dravidian languages, while using a Transformer architecture. While BLEU results were mixed on Malayalam and Telugu, CHRF scores clearly suggest that SentencePiece remains the best option for all of our tested language pairs.

We also found interesting differences among the four target languages. Though they all belong to the same language family and share various linguistic phenomena, they are different with respect to source/target token ratio (Table 3), and the rate of unknown subwords in the development set (Figure 1). Whether this is due to linguistic characteristics or to genre differences in the training corpora remains hard to gauge.

Finally, we invite future researchers to carry out research on Dravidian languages, especially Kannada. Compared to the plethora of work found for other languages, the work on Dravidian languages is lagging behind. As our results show, there remains a large space for improvements, particularly
Figure 2: The Probability density function plot showing the distribution of the average subword length for a given segmenter and language on the training sets. The colored boxes denote the mean of the respective distributions. Also included are the differences in the CHRF scores between SP and LMVR.

when translating into these languages.

References


5th Workshop on Asian Translation, Hong Kong. Association for Computational Linguistics.


