LLMSegm: Surface-level Morphological Segmentation Using Large Language Model

Marko Pranjić[1,2], **Marko Robnik Šikonja**[3], **Senja Pollak**[1]
[1]Jožef Stefan Institute, Ljubljana, Slovenia
[2]Jožef Stefan International Postgraduate School, Ljubljana, Slovenia
[3]University of Ljubljana, Faculty of Computer and Information Science, Ljubljana, Slovenia
{marko.pranjic, senja.pollak}@ijs.si marko.robnik@fri.uni-lj.si

Abstract

Morphological word segmentation splits a given word into its morphemes (roots and affixes), the smallest meaning-bearing units of language. We introduce a novel approach, called LLMSegm, to surface-level morphological segmentation leveraging large language models (LLMs). The proposed approach is applicable in low-data settings as well as for low-resourced languages. We show how to transform the surface-level morphological segmentation task to a binary classification problem and train LLMs to solve it efficiently. For input, we leverage the information from the default LLM subword tokenisation, and a custom morphological segmentation using novel encoding. The evaluation of LLMSegm across seven morphologically diverse languages demonstrates substantial gains in minimally-supervised settings as well as for low-resourced languages, compared to several existing competitive approaches. In terms of F_1-scores and accuracy, we achieve improved results compared to the competing methods in six out of seven datasets.

Keywords: morphological segmentation, surface-level segmentation, large language models, low-resource settings

1. Introduction

Morphological word segmentation splits a given word into its morphemes (roots and affixes), the smallest meaning-bearing units of language. Computational morphological analysis has the potential to improve many natural language processing (NLP) tasks, especially for low-resource languages (Wiemerslage et al., 2022; Eskander et al., 2020) and morphologically-rich languages, which contain a lot of information in morphemes. As part of the morphological analysis, morphological segmentation has been specifically addressed because morpheme segments provide explicit information about word semantics, even for previously unseen words, and provide an intuitive solution to the problem of out-of-vocabulary words (Creutz et al., 2007a). The importance of morphological segmentation has not only been demonstrated in various downstream tasks such as machine translation (Dyer et al., 2008; Huck et al., 2017), dependency parsing (Seeker and Çetinoğlu, 2015), and speech recognition (Creutz et al., 2007b), but it also improves the fundamental ability of language models to produce high-quality word representations (Knigawka, 2022; Hofmann et al., 2021; El-Kishky et al., 2019).

Research distinguishes between two types of morphological segmentation – *canonical segmentation* and *surface-level segmentation*. In canonical segmentation, the target word is treated as a sequence of canonical morphemes that may differ from its written (surface) manifestation. This type of segmentation is very interesting from a linguistic point of view. The second type of segmentation treats the target word as a string of morpheme forms that have already deviated from their canonical representation. Let us take the word "biologist" as an example: the canonical segmentation would yield the result "bio+logy+ist", while the surface-level segmentation would yield "bio-log-ist". The difference lies in the second morpheme, whose canonical representation is "logy", while its (surface) manifestation in the target word is "log". In this paper, we focus on surface-level morphological segmentation, also known as morph boundary prediction or concatenative morphology modeling.

For a morphological segmentation task, labeled training datasets are usually small and research has mainly focused on unsupervised, semi-supervised, and minimally-supervised learning approaches. The supervised learning approaches exist and produce superior results if sufficient data is available, which excludes all mid- and low-resourced languages. Although pretrained large language models (LLMs) have been successfully used in most NLP tasks, to the best of our knowledge, they have not yet been applied to the surface-level morphological segmentation task. One reason for this is the subword tokenization of the input to LLMs. A single token spans from a single character to an entire word. This poses a problem for supervised tagging approaches that work with characters. A fundamental problem is that token boundaries do not match morpheme boundaries, so an LLM approach shall seemingly focus on character-level input, which is a serious limitation.

In this work, we overcome the above limitation of LLM tokenization and provide a novel approach

LREC-COLING 2024, pages 10665–10674

that views morphological segmentation as a binary classification problem suitable for LLMs. For each word in a labeled dataset, we create the training examples by adding a morpheme boundary token after every character in a word. Words augmented with a morpheme boundary that is present in the ground truth segmentation are taken as the positive examples, while the words with boundaries not present in the ground truth segmentation are considered as the negative ones. During inference, we predict for each word the presence of a morpheme boundary in every possible position and collect the predictions to form the morphological segmentation. We evaluate our approach in two different settings that are challenging for existing methods. The first setting is relevant for all languages with a small amount of available annotated data, where supervised approaches are inappropriate. The second setting addresses low-resourced languages where even unsupervised learning approaches are limited by the total amount of available language resources. The experiments show high performance of our approach and improvement over baseline models.

The paper is organized into five sections. Section 2 presents related work on surface-level segmentation, and is followed by Section 3 introducing the datasets used in our experiments. Section 4 presents the proposed methodology, while we describe the experimental results in Section 5. We draw the conclusions and present ideas for further work in Section 6.

2. Related work

There are many approaches to surface-level segmentation, as well as canonical segmentation. The introduction of morphological segmentation challenges enabled systematic comparison of different approaches. The MorphoChallenge (Kurimo et al., 2010b) competition (2005-2010) helped to focus research interest on surface-level morphological segmentation and increase the visibility of systems such as Morfessor Baseline (Creutz and Lagus, 2002), Morfessor CatMAP (Creutz and Lagus, 2005), and ParaMor (Monson et al., 2008). On the other side, Sigmorphon shared task (Batsuren et al., 2022) enabled comparison of canonical segmentation approaches.

Most of the research in surface-level morphological segmentation has focused on the development and application of unsupervised learning systems. This is not surprising, since the creation of datasets with labeled morphological segmentations is expensive and usually involves several hundred to several thousand annotated examples. One of the early systems of this type, Morfessor(Creutz and Lagus, 2002), still serves as a strong baseline and is often used for comparison with newer methods. The Morfessor systems received several extensions in the form of Morfessor CatMAP (Creutz and Lagus, 2005) and Morfessor FlatCat (Grönroos et al., 2014), which improved the method and added support for incorporating labeled data in a form of semi-supervised learning.

Supervised learning methods tackle morpheme segmentation by treating it as a sequence tagging task, where each character of a word is assigned a class indicating its position within a morpheme – beginning, middle, and end of a morpheme. The most common supervised models use recurrent and convolutional neural networks. A popular approach is to use Conditional Random Fields (CRF) model trained on character-level bi-directional LSTM networks (BiLSTM-CRF). An early work of this type leveraged CRF model and extended the supervised approach (Ruokolainen et al., 2013) to a semi-supervised approach (Ruokolainen et al., 2014). Instead of CRF model trained on the set of engineered features incorporating linguistic knowledge, Wang et al. (2016) used LSTM network with a sliding window that automatically learns the structure of input sequences and predicts morphological boundaries. Variants of this system are still present in recent works (Erjavec et al., 2023; Moeng et al., 2022). Even with a moderate number of labeled examples, BiLSTM-CRF approaches can be improved upon with a much simpler CRF model trained on manually selected features, as in Moeng et al. (2022).

An extension of the CRF approach is the use of a semi-Markov CRF (see Chipmunk by Cotterell et al., 2015), which was trained jointly on morphological segmentation, stemming and morphological tag classification. This approach learns a supervised model over labeled data and additionally, leverages features constructed from spellchecker results and an explicit list of affixes for the target language.

An interesting work has been started with the use of adaptor grammars (AG) in morphological segmentation in Sirts and Goldwater (2013) and extended with the Eskander et al. (2021). This approach allows the inclusion of language-specific priors in the form of grammar rules. The sequence-to-sequence approach by Kann et al. (2018) showed promising results by leveraging cross-lingual transfer and learning a single model for four related languages.

Several other systems, such as Cotterell et al. (2016), Kann et al. (2016) and Mager et al. (2020), have been developed for the task of morphological segmentation, but their focus has been *canonical segmentation*, a task closely related, but distinct to, *surface-level segmentation* that is the focus of this work.

3. Datasets

We use data from multiple languages. As our approach is supervised, we focus on datasets with labeled instances. Data from high resource languages (English, Finnish, Turkish) are publicly available from the 2010 MorphoChallenge competition (Kurimo et al., 2010a)[1]. These datasets consist predominantly of unlabeled data and only a small fraction of labeled data. In our experiments, we use only the labeled data and discard the rest. Although the MorphoChallenge competition also contains data for German, this data is unlabeled, which is not suitable for our approach. The datasets available from the competition contain training and validation data but the test set is not publicly available. For this reason, we repurposed some of the data as a test set, as follows. When analyzing the MorphoChallenge validation dataset, we found that many examples are closely related, e.g., singular and plural of the same word, and using part of this data for validation and part for testing purposes could bias the results. For this reason, we take 100 examples from the training dataset and use them as validation data, while the remaining 900 examples are used as training data. The entire validation dataset available from the MorphoChallenge is used as the test data.

In addition to the MorphoChallenge data, we use South African language (Swati, Zulu, Xhosa, and Ndebele) data introduced in Eiselen and Puttkammer (2014) and used in Moeng et al. (2022). The preprocessing, filtering and surface-level segmentation steps for these datasets are detailed in Moeng et al. (2022), and we use these prepared datasets for our experiments with the same training, validation and test splits to allow comparison with their approach. The sizes of the actually used dataset are presented in Table 1. For words that have multiple valid segmentations in the training data, we use only the first segementation to train our model. Multiple valid segementations are, on the other hand, preserved in validation and test datasets and taken into account during evaluation.

Language	Train	Validation	Test
English	900	100	686
Finnish	900	100	835
Turkish	900	100	760
Swati	9630	1070	5610
Zulu	15683	1743	3208
Xhosa	14778	1643	2861
Ndebele	11448	1273	2479

Table 1: Sizes of the train, validation and test split for each of the languages used to evaluate the morphological segmentation.

4. Methodology

In Section 4.1, we introduce the novel method LLMSegm, a surface-level segmentation method leveraging large language models. We present several baseline models in Section 4.2, while the evaluation measures are outlined in Section 4.3. The source code of the proposed LLMSegm system is publicly available[2].

4.1. LLMSegm approach

We approach the surface-level morphological segmentation as a binary classification task, where we determine whether a position within the word matches a morpheme boundary or not. To train a segmenter, we have to prepare a dataset. For each word, we generate training examples so that each example contains a single split somewhere within a word. Given a word length of L characters, we generate $L-1$ examples to cover all positions where morphological boundaries may occur. The labels assigned to these examples reflect the correct morphological boundaries. The examples of training instances for the word *unbounded* are contained in Table 2. During inference, we predict for each word the presence of a morphological boundary in each possible position and collect the predictions to form the output segmentation.

Target word	Ground Truth	Train set	Label
unbounded	un-bound-ed	u•nbounded	✗
		un•bounded	✓
		unb•ounded	✗
		unbo•unded	✗
		unbou•nded	✗
		unboun•ded	✗
		unbound•ed	✓
		unbounde•d	✗

Table 2: The training examples of morpheme boundaries for a target word *unbounded*. Every position in the word is labeled with a corresponding label.

Our novelty lies with the application of powerful LLMs to the problem of morphological word segmentation. Note that almost all pretrained LLMs use a subword tokenization algorithm such as SentencePiece, that forms a vocabulary of the model and the corresponding tokenizer. Applying the tokenizer on the input words produces, for each word,

[1] Available from: http://morpho.aalto.fi/events/morphochallenge2010/datasets.shtml#download

[2] https://github.com/sharpsy/llm-morphological-segmenter

one or more tokens based on this vocabulary. In our work, each input example presented to the LLM model contains two occurrences of the target word separated with a "‡" (word boundary) token. First occurrence of the word is an unmodified target word that preserves all tokens used for this word. The second occurrence of the target word contains a "•" (morpheme boundary) token inserted somewhere in the word, as shown in the example below. The position in the word where the "•" token is inserted is labeled by the model as either a true boundary between two morphemes or a position within a single morpheme, as shown in Table 2. The insertion of the morpheme boundary token at the specified position of the target word forces the tokenizer to split any token spanning across this position into two tokens. However, such tokenization risks the loss of information from the original tokens of the target word. Thus, to mitigate this issue, the default tokenization of the target word was retained and included together with the word boundary token, ensuring the preservation of valuable linguistic information. The final input to the LLM tokenizer on the example of segmentation of the word "*unbounded*" with the morpheme boundary positioned in the middle (and corresponding to one of the negative training examples from Table 2), together with a resulting list of tokens, would look like:

Tokenizer input: "unbounded‡unbo•unded"

Tokens: un, bound, ed, ‡, un, bo, •, und, ed

A tokenizer input from above is a single example containing two occurrences of the target word, an unmodified (*unbounded*) and augmented with a morpheme boundary (*unbo•unded*), together with a list of tokens created by the tokenizer applied to this input. The embeddings of the two tokens added to the vocabulary ("•" and "‡")[3]. are initialized to zero and the model learns their representation during fine-tuning.

For all experiments, we use the Glot500-m large language model (LLM) pretrained on a corpus of 511 languages (ImaniGooghari et al., 2023). We fine-tune the model using our binary classification task with the AdamW optimizer (Loshchilov and Hutter, 2017) using batches of 256 examples. After each epoch, we compute the F_1-score on the validation set; the model with the best score is used after the training is completed. We use the linear learning rate schedule with 20 warm-up iterations to a learning rate of 2×10^{-5} and set the dropout to 0.01 during training for a total of 30 epochs[4]. Due to class imbalance, we used different weights for each class. The weights for each of the C classes were calculated using the following heuristic[5]:

$$w_i = \frac{N}{|C| \cdot |C_i|}$$

In the above equation, w_i denotes the weight of the examples of class i, N denotes the total number of training examples, $|C|$ denotes the number of classes (2 for binary classification), and $|C_i|$ denotes the number of examples within the class i.

Our method is novel compared to related work (see Section 2). Approaches that use neural networks are widely used for the task of surface-level morphological segmentation, but none uses pretrained LLMs, such as BERT (Devlin et al., 2019) (or Glot500 in our case). Instead, character-level models in the form of bidirectional LSTM networks are used, often coupled with Conditional Random Fields (CRFs), as in Erjavec et al. (2023); Moeng et al. (2022). The main reason for the lack of LLM-based approaches to morphological word segmentation is their own subword tokenization. A subword token, used in LLM such as BERT, often spans multiple characters or even the whole word. In many cases, token boundaries are not aligned with morpheme boundaries and tagging approaches such as (Ruokolainen et al., 2013) are not applicable. We overcame these issues by proposing a complementary scheme that utilizes the information present in the existing statistical subword tokenization and additional fine-tuning on (at least a few) examples of correct surface-level morphological segmentation. The advantage of the proposed approach compared to a wide range of segmentation models presented in Section 4.2 is demonstrated in Section 5.

4.2. Baseline models

In this section, we present a range of morphological segmenters used as baselines in our empirical evaluation.

4.2.1. Glot500 tokenizer

Our approach relies on fine-tuning the Glot500 model introduced in ImaniGooghari et al. (2023). In order to understand the benefits of fine-tuning the model, and disentangle the performance of the model from the inherent segmentation introduced

[3]The implementation uses custom tokens not present in the vocabulary, "•" and "‡" are used here for the presentation purposes.

[4]The used optimizer, learning rate, and dropout are default values of the fine-tuning procedure implemented in the HuggingFace Transformers library available at https://github.com/huggingface/transformers

[5]The heuristic corresponds to the 'balanced' strategy of the class weight calculation implemented at: https://scikit-learn.org/stable/modules/generated/sklearn.utils.class_weight.compute_class_weight.html

by the underlying tokenizer, we evaluate the tokenizer on its own. The Glot500 tokenizer is derived from the vocabulary of the XLM-R model (Conneau et al., 2020) and extended with additional 151 thousand new tokens, added to the existing 250 thousand tokens present in the XLM-R. New tokens are selected by training a new tokenizer using SentencePiece (Kudo and Richardson, 2018) algorithm with unigram language model (Kudo, 2018) on training data collected in ImaniGooghari et al. (2023).

4.2.2. Chipmunk

ChipMunk is a supervised segmentation, morphological analysis, and stemming tool presented in Cotterell et al. (2015). It is based on the Semi-Markov Conditional Random Fields (Semi-CRF) model (Sarawagi and Cohen, 2004), which has been shown to work well for morphological analysis. It uses custom features like a list of affixes for the target language and spellchecker results together with *n*-gram context features from Ruokolainen et al. (2013). In our benchmarks, we do not train a Chipmunk model but use publicly available pretrained models[6]. Where multiple models are available for the target language, we select the newest model. In contrast to our model that was trained using a single segmentation, Chipmunk can make use of training data containing multiple valid segmentations.

4.2.3. Morfessor

Morfessor is a class of morphological segmentation methods based on a generative probabilistic model. Morfessor searches for the smallest morph lexicon that strikes a balance between the accurate encoding of the training corpus and the size of that lexicon. We use publicly available implementation[7] described in Virpioja et al. (2013). The model is trained in semi-supervised mode, i.e. labeled training data is provided along with the unlabeled validation data.

4.2.4. Feature-based conditional random fields

Use of conditional random fields for the task of morphological segmentation was first presented in Ruokolainen et al. (2013) and successfully applied in a setting with a limited number of labeled examples. The problem is formed as a sequence labeling task where each character of a word is assigned a class corresponding to its position in a morpheme – the beginning of a morpheme, the middle, or the end of a morpheme. As a special case, a class for the single character morpheme is sometimes used. Moeng et al. (2022) have shown that a CRF model using manually created features can outperform a neural network in a limited-data setting on a morphological segmentation task. They used both binary features on characters (is the character a vowel or a consonant, is it uppercase or lowercase letter), as well as character n-grams. An additional advantage of using such features is that user-defined linguistic priors can be easily incorporated into the model. In our evaluation, we use publicly available models from Moeng et al. (2022) and compare them with our approach.

4.2.5. BiLSTM-CRF

A character-based neural network coupled with conditional random fields (CRF) is a popular choice for morphological segmentation when at least some labeled data is available. The model is trained on the sequence labeling task as described in the previous section (Section 4.2.4). One reason for using the neural network is that CRF model requires a set of features to be trained. Neural network can take over the task of feature extraction and learn those features from the data. This approach was used in Moeng et al. (2022) and Erjavec et al. (2023). We use the publicly available models presented in Moeng et al. (2022) using the same training, validation, and testing datasets for Zulu, Swati, Xhosa, and Ndebele languages. In addition, we train models from Erjavec et al. (2023)[8] on each language described in Section 3.

4.3. Evaluation measures

To compare different approaches, we use the boundary precision recall (BPR) metric. It is a widely used (Ruokolainen et al., 2016) and intuitive metric for evaluating the correctness of the morphological segmentation models by comparing the positions of splits in the ground truth and predicted segmentations. Precision (P), recall (R), and BPR F_1 score can be defined by counting the morpheme boundaries that match the ground truth (TP), the boundaries missed by the prediction (FN), and the boundaries predicted by the model but are not present in the ground truth (FP):

$$P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN} \quad F_1 = 2 \cdot \frac{P \cdot R}{P + R}$$

As evident from the definitions, the BPR metric is closely related to the standard metrics in information retrieval (IR). Morphological segmentation, interpreted as an IR task, can be seen as a retrieval of morpheme boundary positions within words. Due to class imbalance, we use macro-averaged metrics implemented in the `morphoeval` Python pack-

[6] https://cistern.cis.lmu.de/chipmunk/
[7] https://morfessor.readthedocs.io/
[8] Provided by the authors of the paper

age[9]. In case of multiple valid segmentations, BPR takes into account the ground truth segmentation that provides the best match with the predicted segmentation. In addition to measuring the BPR F_1-score, which evaluates morpheme boundaries, we also evaluate the word segmentation accuracy, defined as the proportion of words that were entirely correctly segmented. This definition of accuracy is used in (Kann et al., 2018) and Erjavec et al. (2023). We extend it here to the case of multiple valid segmentations by counting a correct prediction if any of the ground truth segmentations matches the prediction, analogous to the BPR metric.

5. Results

In this section, we compare the performance of the LLMSegm approach with the baseline models. The results (F_1-scores and accuracy), measured on the test sets are presented in Table 3. The largest difference in interpretation of BPR F_1-score and accuracy is in their handling of partial matches. The F_1-score assigns partial credit for a partial match, while accuracy counts only completely correct segmentations.

The simplest baseline, tokeniser of the Glot500 model, achieves high accuracy on Swati language as well as notable F_1-score on English. While this model is superior to some specialized morphological segmentation models on English, the best models in our evaluation, LLMSegm, still significantly outperforms it. The accuracy in the segmentation of Swati can be attributed to the prevalence of short words, ranging from two to four characters and composed of a single morpheme in the test corpus. These short words are effectively tokenized as individual tokens, which likely contributes to the accuracy scores.

Publicly available Chipmunk models are available only for English, Finnish, Turkish and Zulu, and we evaluate them on our test data. Likewise, BiLSTM-CRF and Feature-CRF models from Moeng et al. (2022) are available only for Swati, Xhosa, Zulu and Ndebele. We train the rest of the models on our training data, with a minor difference of Morfessor that additionally trains on unlabeled validation data, as it is a semi-supervised model. Chipmunk model performs the best on the Finnish dataset, followed by our LLMSegm method. On all other languages, the LLMSegm method is better than other approaches in both minimally-supervised comparison using only 900 labeled examples and in comparisons using low-resourced language datasets.

The Morfessor algorithm trained in semi-supervised mod was one of the fastest models to train. With a modest dataset comprising 900 labeled examples and 100 unlabeled instances, the algorithm yielded commendable BPR F_1-scores. Nonetheless, the low accuracy metrics suggest that the model tends to generate only partially correct results.

Although in their paper, Moeng et al. (2022) report much higher performance for morphological segmentation on the South African languages than our results, we were not able to reproduce those results using publicly available models and test sets. The authors use custom evaluation code and different F_1 evaluation metrics (we use the F_1-scores calculated by the BPR metric), while accuracy results are not provided. Thus, we reproduce both our metrics using their models.

Slightly different implementation of the BiLSTM-CRF models from Erjavec et al. (2023) performs better than the models from Moeng et al. (2022) and matches the level of performance of feature based CRF from Moeng et al. (2022) on this dataset. The performance of the model from the Erjavec et al. (2023) on English, Finnish and Turkish is significantly lower than on South African languages, but this is to be expected given the supervised nature of the model and amount of training examples present in each language.

Our approach achieves high performance on all datasets used in our experiments. In related work, one can find reported results on different data splits of the MorphoChallenge data, but due to lack of publicly available test data for English, Finnish and Turkish, we cannot compare our results directly – but results presented here look favorably for our method. For example, test set BPR F_1-scores reported in Eskander et al. (2021) using Adaptor Grammars are measured on the MorphoChallenge data, but test dataset was constructed in a different way. Compared to those results, ours are 5 to 12 percentage points higher than the results reported for English, Finnish and Turkish. Moreover, their best models often require explicit linguistic information.

In minimally-supervised setting, LLMSegm improved on the Chipmunk results using only 900 labeled examples on English and Turkish dataset. In Finnish, the result is slightly lower than the Chipmunk's. In addition to using annotated data of comparable size, Chipmunk model is trained with the information on several objectives and contains explicit information about possible affixes for a target language as well as spell-checker results. Tests done on low-resource languages show improved performance over all baselines. Results evaluated using accuracy and BPR F_1-score follow the same trend, except the accuracy shows lower absolute values. This is to be expected as any mistake in the segmentation on the level of a word is reflected

[9] https://github.com/svirpioj/morphoeval

Segmenter / Language	English	Finnish	Turkish	Swati	Zulu	Xhosa	Ndebele
	BPR F_1-score						
Glot500 tokenizer (ImaniGooghari et al., 2023)	60.18	45.32	50.39	47.57	42.55	42.58	40.00
Morfessor (Virpioja et al., 2013)	47.27	55.42	73.78	65.82	70.20	73.25	65.91
Chipmunk (Cotterell et al., 2015)	87.19	**88.46**	82.13	–	75.02	–	–
Feature-CRF (Moeng et al., 2022)	–	–	–	85.63	81.50	81.87	77.01
BiLSTM-CRF (Moeng et al., 2022)	–	–	–	59.01	82.01	76.15	75.65
BiLSTM-CRF (Erjavec et al., 2023)	52.45	21.13	49.95	85.20	80.42	81.91	78.64
LLMSegm (ours)	**89.37**	84.44	**87.69**	**90.68**	**86.28**	**85.14**	**83.44**
	Accuracy						
Glot500 tokenizer (ImaniGooghari et al., 2023)	27.41	5.39	8.42	55.81	12.31	13.39	8.83
Morfessor (Virpioja et al., 2013)	6.71	8.26	20.53	42.03	34.85	38.48	27.39
Chipmunk (Cotterell et al., 2015)	59.77	**65.63**	52.03	–	34.69	–	–
Feature-CRF (Moeng et al., 2022)	–	–	–	63.89	49.78	52.36	44.66
BiLSTM-CRF (Moeng et al., 2022)	–	–	–	58.06	53.71	36.21	37.35
BiLSTM-CRF (Erjavec et al., 2023)	26.53	4.43	9.47	68.24	48.04	48.20	44.25
LLMSegm (ours)	**68.80**	45.39	**52.11**	**73.85**	**62.47**	**59.70**	**55.43**

Table 3: Results of different morphological segmentation methods on seven languages evaluated using the BPR F_1-score and accuracy. The English, Finnish and Turkish datasets are from 2010 MorphoChallenge (Mikko Kurimo and Turunen, 2010) and represent well resourced languages with a small sample of annotated data. The Swati, Zulu, Xosa, and Ndebele datasets from Moeng et al. (2022) represent low-resourced languages with more annotated data.

in the accuracy, while F_1-score better reflects an average performance of the model to detect a split within a word.

6. Conclusion and future work

We introduce a novel approach to surface-level morphological segmentation leveraging the large language models applicable in low-data setting as well as in low-resourced languages. We treat the problem as a binary classification problem and train a large language model to solve it. The novelty of the proposed approach is in encoding the input to utilize the information from existing subword tokenization and language knowledge in BERT-like models to fine-tune the LLMs for the morphological segmentation task. We use Glot500 model from ImaniGooghari et al. (2023) and fine-tune it on the labeled dataset. We test the proposed approach on 7 languages with diverse morphological complexity and improve on existing methods in most of the experiments (6 out of 7 languages), both in terms of F_1-score and accuracy.

While comparable morphological segmentation systems often use language-specific features, like handcrafted rules (Moeng et al., 2022), lists of common affixes and a spellchecker (Cotterell et al., 2015), or require a large amount of data (Grönroos et al., 2014), our system can leverage small amount of annotated data to adapt to a large number of languages supported by massively multilingual BERT-like models.

During training, our model assumes a single valid segmentation for each word and leverages independence of a morpheme boundary on positions of other boundaries. Those assumptions are not realistic as some words can have multiple valid segmentations – a property that is challenging for modeling. Approaches based on neural networks and CRFs (including Chipmunk) are limited to a single prediction and approaches based on the maximum likelihood optimization (like Morfessor) can produce multiple segmentations only if the desired number of segmentations is known in advance (N-best strategy). In contrast to those approaches, BERT-like models (like ours) are inherently strong in using contextual information; therefore, we would like to extend our approach and leverage contextual information to provide context-aware segmentation in those cases.

Our approach can be potentially improved by extending it with the morphological tag classifier that predicts the most likely inflectional features of the target word together with its segmentation. Additionally, cross-lingual performance as evaluated in Kann et al. (2018) could improve the model performance when morphological segmentation of words in related languages is jointly trained. We will explore these directions in future work.

In addition, it makes sense to use the obtained morphological segments in downstream tasks such as POS tagging for morphologically-rich languages. Finally, it would be interesting to test the performance of LLMs such as ChatGPT or LLaMa. While these models shall perform well using in-context learning for well-resourced languages, their abilities on low-resourced languages, as the ones in the African corpora, is questionable.

7. Limitations

Morphological segmentation approach described in this paper implies the use of pretrained large language model even in low-resource scenario. A limiting factor is that low-resourced languages are less likely to be supported by large language models due to not having sufficient training data to support pretraining of those models. For languages totally left-out of the multilingual LLM training, the method is thus not directly applicable.

Input to the LLM is constructed from two occurrences of the target word in order to retain the linguistic information encoded in the tokens learned during pretraining, but different variants of the input are not explored. It's unclear if the proposed model input construction significantly affects the model's performance or if the model merely learns the most typical tokenization of the affix in the target word. Clarifying this would require an ablation study, which would assess how the model performs when the unmodified target word is not provided.

While our method is well-suited for languages with linear word formation, where affixes are appended to the stem, it is less straightforward for handling non-linear morphological processes. Semitic languages, which feature complex verb constructions by interleaving the root with a predefined patterns, pose a challenge to our approach, as they lack clear morpheme boundaries and require additional consideration.

Acknowledgements

The work was supported by the Slovenian Research and Innovation Agency core research programmes Knowledge Technologies (P2-0103) and Language Resources and Technologies for Slovene (P6-0411), as well as the projects Formant combinatorics in Slovenian (J6-3131) and Embeddings-based techniques for Media Monitoring Applications (L2-50070). The work was also supported via the bilateral PROTEUS project Cross-lingual and Cross-domain methods for Terminology Extraction and Alignment (BI-FR/23-24-PROTEUS006).

8. Bibliographical References

Khuyagbaatar Batsuren, Gábor Bella, Aryaman Arora, Viktor Martinovic, Kyle Gorman, Zdeněk Žabokrtský, Amarsanaa Ganbold, Šárka Dohnalová, Magda Ševčíková, Kateřina Pelegrinová, Fausto Giunchiglia, Ryan Cotterell, and Ekaterina Vylomova. 2022. The SIGMORPHON 2022 shared task on morpheme segmentation. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 103–116.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451.

Ryan Cotterell, Thomas Müller, Alexander Fraser, and Hinrich Schütze. 2015. Labeled morphological segmentation with semi-Markov models. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pages 164–174.

Ryan Cotterell, Tim Vieira, and Hinrich Schütze. 2016. A joint model of orthography and morphological segmentation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 664–669.

Mathias Creutz, Teemu Hirsimäki, Mikko Kurimo, Antti Puurula, Janne Pylkkönen, Vesa Siivola, Matti Varjokallio, Ebru Arisoy, Murat Saraçlar, and Andreas Stolcke. 2007a. Morph-based speech recognition and modeling of out-of-vocabulary words across languages. *ACM Trans. Speech Lang. Process.*, 5(1).

Mathias Creutz, Teemu Hirsimäki, Mikko Kurimo, Antti Puurula, Janne Pylkkönen, Vesa Siivola, Matti Varjokallio, Ebru Arisoy, Murat Saraclar, and Andreas Stolcke. 2007b. Morph-based speech recognition and modeling of out-of-vocabulary words across languages. *ACM Transactions on Speech and Language Processing (TSLP)*, 5:3.

Mathias Creutz and Krista Lagus. 2002. Unsupervised discovery of morphemes. In *Proceedings of the ACL-02 Workshop on Morphological and Phonological Learning*, pages 21–30.

Mathias Johan Philip Creutz and Krista Hannele Lagus. 2005. Inducing the morphological lexicon of a natural language from unannotated text. In *Proc. International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning (AKRR'05)*, pages 106–113.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the*

Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Volume 1 (Long and Short Papers), pages 4171–4186.

Christopher Dyer, Smaranda Muresan, and Philip Resnik. 2008. Generalizing word lattice translation. In *Proceedings of ACL-08: HLT*, pages 1012–1020.

Roald Eiselen and Martin Puttkammer. 2014. Developing text resources for ten South African languages. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3698–3703.

Ahmed El-Kishky, Frank F. Xu, Aston Zhang, and Jiawei Han. 2019. Parsimonious morpheme segmentation with an application to enriching word embeddings. *2019 IEEE International Conference on Big Data (Big Data)*, pages 64–73.

Tomaž Erjavec, Marko Pranjić, Andraž Pelicon, Boris Kern, Irena Stramljič Breznik, and Senja Pollak. 2023. Automating derivational morphology for slovenian. page 449–465. Lexical Computing CZ. Nasl. z nasl. zaslona.

Ramy Eskander, Francesca Callejas, Elizabeth Nichols, Judith L. Klavans, and Smaranda Muresan. 2020. Morphagram, evaluation and framework for unsupervised morphological segmentation. In *International Conference on Language Resources and Evaluation*.

Ramy Eskander, Cass Lowry, Sujay Khandagale, Francesca Callejas, Judith Klavans, Maria Polinsky, and Smaranda Muresan. 2021. Minimally-supervised morphological segmentation using Adaptor Grammars with linguistic priors. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3969–3974.

Stig-Arne Grönroos, Sami Virpioja, Peter Smit, and Mikko Kurimo. 2014. Morfessor FlatCat: An HMM-based method for unsupervised and semi-supervised learning of morphology. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1177–1185.

Valentin Hofmann, Janet Pierrehumbert, and Hinrich Schütze. 2021. Superbizarre is not superb: Derivational morphology improves BERT's interpretation of complex words. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3594–3608.

Matthias Huck, Aleš Tamchyna, Ondřej Bojar, and Alexander Fraser. 2017. Producing unseen morphological variants in statistical machine translation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 369–375.

Ayyoob ImaniGooghari, Peiqin Lin, Amir Hossein Kargaran, Silvia Severini, Masoud Jalili Sabet, Nora Kassner, Chunlan Ma, Helmut Schmid, André Martins, François Yvon, and Hinrich Schütze. 2023. Glot500: Scaling multilingual corpora and language models to 500 languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1082–1117.

Katharina Kann, Ryan Cotterell, and Hinrich Schütze. 2016. Neural morphological analysis: Encoding-decoding canonical segments. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 961–967, Austin, Texas. Association for Computational Linguistics.

Katharina Kann, Jesus Manuel Mager Hois, Ivan Vladimir Meza-Ruiz, and Hinrich Schütze. 2018. Fortification of neural morphological segmentation models for polysynthetic minimal-resource languages. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 47–57.

Łukasz Knigawka. 2022. Constructing a derivational morphology resource with transformer morpheme segmentation. In *Proceedings of the 18th Conference on Natural Language Processing (KONVENS 2022)*, pages 104–109.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71.

Mikko Kurimo, Sami Virpioja, and T. Ville Turunen. 2010a. Overview and results of Morpho Challenge 2010. In *Proceedings of the Morpho Challenge 2010 Workshop*.

Mikko Kurimo, Sami Virpioja, Ville Turunen, and Krista Lagus. 2010b. Morpho challenge 2005-2010: Evaluations and results. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, pages 87–95.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Manuel Mager, Özlem Çetinoğlu, and Katharina Kann. 2020. Tackling the low-resource challenge for canonical segmentation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5237–5250.

Sami Virpioja Mikko Kurimo and Ville T. Turunen. 2010. Overview and results of morpho challenge 2010. In *Proceedings of the MORPHO challenge 2010 workshop*.

Tumi Moeng, Sheldon Reay, Aaron Daniels, and Jan Buys. 2022. Canonical and surface morphological segmentation for nguni languages. In *Artificial Intelligence Research*, pages 125–139, Cham. Springer International Publishing.

Christian Monson, Jaime Carbonell, Alon Lavie, and Lori Levin. 2008. Paramor: Finding paradigms across morphology. In *Advances in Multilingual and Multimodal Information Retrieval*, pages 900–907, Berlin, Heidelberg. Springer Berlin Heidelberg.

Teemu Ruokolainen, Oskar Kohonen, Kairit Sirts, Stig-Arne Grönroos, Mikko Kurimo, and Sami Virpioja. 2016. A comparative study of minimally supervised morphological segmentation. *Computational Linguistics*, 42(1):91–120.

Teemu Ruokolainen, Oskar Kohonen, Sami Virpioja, and Mikko Kurimo. 2013. Supervised morphological segmentation in a low-resource learning setting using conditional random fields. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 29–37, Sofia, Bulgaria. Association for Computational Linguistics.

Teemu Ruokolainen, Oskar Kohonen, Sami Virpioja, and Mikko Kurimo. 2014. Painless semi-supervised morphological segmentation using conditional random fields. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2: Short Papers*, pages 84–89.

Sunita Sarawagi and William W Cohen. 2004. Semi-markov conditional random fields for information extraction. In *Advances in Neural Information Processing Systems*, volume 17.

Wolfgang Seeker and Özlem Çetinoğlu. 2015. A Graph-based Lattice Dependency Parser for Joint Morphological Segmentation and Syntactic Analysis. *Transactions of the Association for Computational Linguistics*, 3:359–373.

Kairit Sirts and Sharon Goldwater. 2013. Minimally-supervised morphological segmentation using Adaptor Grammars. *Transactions of the Association for Computational Linguistics*, 1:255–266.

Sami Virpioja, Peter Smit, Stig-Arne Grönroos, and Mikko Kurimo. 2013. Morfessor 2.0: Python Implementation and Extensions for Morfessor Baseline. Technical report.

Linlin Wang, Zhu Cao, Yu Xia, and Gerard de Melo. 2016. Morphological segmentation with window lstm neural networks. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, page 2842–2848.

Adam Wiemerslage, Miikka Silfverberg, Changbing Yang, Arya McCarthy, Garrett Nicolai, Eliana Colunga, and Katharina Kann. 2022. Morphological processing of low-resource languages: Where we are and what's next. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 988–1007.