Neural Morphological Tagging for Nguni Languages

Cael Marquard* **Simbarashe Mawere*** **Francois Meyer**
Department of Computer Science
University of Cape Town
{mrqcae001,mwrsim003}@myuct.ac.za, francois.meyer@uct.ac.za

Abstract

Morphological parsing is the task of decomposing words into morphemes, the smallest units of meaning in a language, and labelling their grammatical roles. It is a particularly challenging task for agglutinative languages, such as the Nguni languages of South Africa, which construct words by concatenating multiple morphemes. A morphological parsing system can be framed as a pipeline with two separate components, a segmenter followed by a tagger. This paper investigates the use of neural methods to build morphological taggers for the four Nguni languages. We compare two classes of approaches: training neural sequence labellers (LSTMs and neural CRFs) from scratch and finetuning pretrained language models. We compare performance across these two categories, as well as to a traditional rule-based morphological parser. Neural taggers comfortably outperform the rule-based baseline and models trained from scratch tend to outperform pretrained models. We also compare parsing results across different upstream segmenters and with varying linguistic input features. Our findings confirm the viability of employing neural taggers based on pre-existing morphological segmenters for the Nguni languages.

1 Introduction

The smallest unit of linguistic meaning that a word can be split into is known as a *morpheme* (Matthews, 1991). Morphological parsing is the task of identifying the grammatical role of each morpheme within a word (Puttkammer and Du Toit, 2021). For example, “izinhlobo” (meaning “types” in isiZulu) is split into the morphemes “i-zin-hlobo”, which is parsed as “i[NPrePre10] - zin[BPre10] - hlobo[NStem]” (Gaustad and Puttkammer, 2022) (see Figure 1). Each bracketed tag labels the preceding morpheme with its grammatical function and noun class (if applicable).

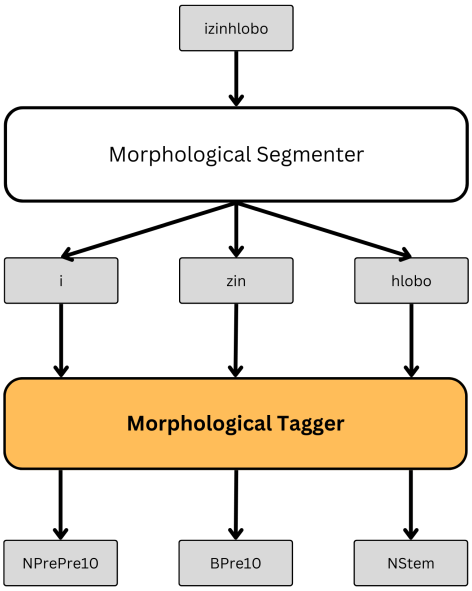

Figure 1: Morphological parsing as a two-step pipeline. We focus on tagging, training our taggers on the outputs of pre-existing morphological segmenters.

Morphological information is especially important for the Nguni languages, a group of related languages (isiNdebele, isiXhosa, isiZulu, and Siswati) spoken across South Africa by more than 23m home language speakers (Eberhard et al., 2019). The Nguni languages are agglutinative, meaning that many words are created by aggregating multiple morphemes (Taljard and Bosch, 2006). They are also written conjunctively—morphemes are concatenated into a single orthographic (space-delimited) word (Taljard and Bosch, 2006). This can produce long, complex word forms consisting of several morphemes, such as the isiXhosa word “andikambuzi”, which means “I have not yet asked them”, composed of the morphemes “a”, “ndi”, “ka”, “m”, “buza”, and “i”.

As a result of this morphological complexity, morphological parsing is a challenging but important task for the Nguni languages. Despite this, few morphological parsers exist for the Nguni lan-

[*]Equal contribution.

Proceedings of the Sixth Workshop on African Natural Language Processing (AfricaNLP 2025), pages 210–220
July 31, 2025 ©2025 Association for Computational Linguistics

guages. Moreover, no existing parsers use neural methods, despite their established performance gains for linguistic annotation tasks (Min et al., 2023). In this paper we explore the viability of neural morphological parsers for the Nguni languages.

Morphological parsing can be framed as a two-step pipeline (Tsarfaty et al., 2013; Puttkammer and Du Toit, 2021), in which raw text is first segmented into morphemes, which are subsequently tagged with morphological labels. The first part of this pipeline is known as *morphological segmentation*, while the second part is known as *morphological tagging*. We visualise this pipeline for the isiZulu word "izinhlobo" in Figure 1. In this work we focus on the second subtask, morphological tagging. Instead of training models for the entire task, we make use of pre-existing morphological segmenters for the Nguni languages (Moeng et al., 2021) and train neural taggers on top of their output.

We train two classes of neural taggers – neural sequence labellers trained from scratch and finetuned pretrained language models (PLMs). Our models trained from scratch are bi-LSTMs (Hochreiter and Schmidhuber, 1997) and conditional random fields (CRFs) (Lafferty et al., 2001) with bi-LSTM features, using either morpheme or character-level input features. For PLMs, we finetune XLM-R-large (Conneau et al., 2020), Afro-XLMR-large (Alabi et al., 2022), and Nguni-XLMR-large (Meyer et al., 2024), which respectively represent different levels of Nguni-language coverage.

We develop neural taggers based on two types of morphological segmentations: canonical and surface segmentations (Cotterell et al., 2016). Canonical segmentation decomposes a word into its constituent morphemes, in their standardised (precomposed) form. For example, the isiXhosa word "zobomi" is canonically segmented into "za-u-(bu)-bomi", where some of the morphemes undergo spelling changes in word composition (Gaustad and Puttkammer, 2022). Surface segmentation decomposes a word into its constituent *morphs*, which are the surface forms of morphemes as they appear in the composed word. For example, "zobomi" is surface-segmented into "zo-bomi". As demonstrated by this example, the canonical and surface-level segmentation of a word can differ.

We evaluate all our models in two settings. In the first, we test our taggers on the morphological segmentations available in our task dataset (Gaustad and Puttkammer, 2022). This provides an idealised setting in which we evaluate our models on gold-annotated segmentations, which we know to be correct, isolating tagging performance from segmentation mistakes. In the second setting, we test our taggers on the segmentations produced by the neural segmenters of Moeng et al. (2021). These are model-predicted segmentations, so some segmentations will not align with morphological boundaries. This can lead to error propagation, in which segmentation errors degrade tagging performance. However, it also provides us with an estimate of how our taggers fare in a real-world setting in which the entire morphological parsing pipeline is predicted by neural models.

Overall, we evaluate four variants of each model configuration – trained on canonical/surface segmentations, and respectively tested on gold-annotated/model-predicted segmentations. Our study is an extensive investigation into the potential of neural parsers for all four Nguni languages. Our main findings can be summarised as follows:

- Neural parsing comfortably outperforms our rule-based baseline, confirming the benefit of data-driven segmentation and tagging.
- Neural sequence labellers trained from scratch outperform finetuned PLMs on the morphological tagging subtask.
- With no access to gold-annotated morphological segmentations, canonical segmentations consistently leads to better parsing performance than surface segmentations.

We are the first to use neural models to train morphological taggers for the Nguni languages. To the best of our knowledge, our morphological parsing results represent state-of-the-art performance. Our models can be used to incorporate morphological information into downstream NLP models, which holds the potential to improve performance for the morphologically complex Nguni languages.

2 Related Work

Morphological parsing has been extensively studied in NLP (Tsarfaty et al., 2013; Klemen et al., 2023). Traditionally, it is performed by incorporating grammatical and morphological rules from the language into a finite-state transducer. This is a time-consuming process in which linguists construct hand-crafted rules (Chapin and Norton, 1968). As in other tasks of linguistic annotation

(Min et al., 2023), neural models provide an effective, data-driven solution approach to morphological parsing.

Several works have trained a single model for morphological parsing, jointly modelling morphological segmentation and tagging (Seker and Tsarfaty, 2020a; Aleçakır, 2020; Abudouwaili et al., 2023; Yshaayahu Levi and Tsarfaty, 2024). Alternatively, Tsarfaty et al. (2013) propose a two-step architecture for parsing morphologically rich languages by first segmenting them into their morphemes and then tagging the morphemes with labels. Because morphological segmenters for Nguni languages already exist (Moeng et al., 2021), we choose to adopt this two-step pipeline approach, visualised in Figure 1. Despite the drawbacks of error propagation, training neural taggers alone is simpler than training joint segmentation-tagging models. The approach is also more modular, allowing for better segmenters to be substituted in as and when they are developed.

A number of works have developed morphological segmenters, taggers, and parsers for the Nguni languages. ZulMorph (Bosch et al., 2008) is a rule-based canonical segmenter and tagger for isiZulu based on finite-state transducers. Puttkammer and Du Toit (2021) develop data-driven (non-neural) canonical segmenters and taggers for all four Nguni languages. They apply TiMBL (Daelemans et al.), a memory-based learning package, to the segmentation step, and MarMoT (Björkelund et al., 2013; Mueller et al., 2013), a trainable CRF pipeline, to the tagging step. Moeng et al. (2021) were the first to apply neural methods to segmentation, using CRFs (Lafferty et al., 2001), LSTMs (Hochreiter and Schmidhuber, 1997), and Transformers (Vaswani et al., 2023) to train canonical and surface-level segmenters for all four Nguni languages. They found that non-neural CRFs were best for surface segmentation, while Transformers outperformed the other methods in canonical segmentation. Despite recent developments in neural models, such as sequence-to-sequence (Akyürek et al., 2019) and sequence labeling models (Ma and Hovy, 2016), no neural morphological taggers currently exist for the Nguni languages.

3 Tagging Models

We now introduce our neural morphological taggers. Our models are trained on sequences of pre-segmented morphemes as input, and are tasked with assigning a morphological label to each morpheme. By focusing on the morphological tagging component of the morphological parsing pipeline (Figure 1), we can use established approaches to neural sequence tagging.

3.1 Neural sequence labellers

We train two types of neural models from scratch: bidirectional long short-term memory (bi-LSTM) networks (Hochreiter and Schmidhuber, 1997) and conditional random fields (CRFs) (Lafferty et al., 2001) with bi-LSTM features. Bi-LSTMs have previously been successfully applied to POS tagging (Pannach et al., 2022) and morphological segmentation (Moeng et al., 2021) for the Nguni languages.

CRFs are probabilistic models for sequence labelling. A CRF estimates the probability of a given output (label) sequence by modelling the interdependence of labels with each other, as well as their dependence on the input sequence. We use linear-chain CRFs because of their lower computational complexity (compared to higher-order CRFs). Traditionally, CRFs use a set of hand-crafted features to assign probabilities (Moeng et al., 2021). However, instead of designing these features by hand, a neural network can be used to automatically learn the features from the data (Moeng et al., 2021; Lample et al., 2016; Ma and Hovy, 2016). We choose a bi-LSTM to generate these features, as this has previously proved successful in POS tagging (Pannach et al., 2022) and morphological segmentation (Moeng et al., 2021) for the Nguni languages.

We experimented with several design choices for our neural models trained from scratch, varying the following factors:

- **Feature level.** Models were trained on either morpheme-level or character-level input features, represented by learned embeddings in both cases. For morpheme-level features, we replaced rare morphemes (<2 examples in the training data) with a special unknown token to help the model generalise to unseen data. For character-level features, we summed character embeddings to produce morpheme-level input embeddings. Surface models also have lowercase variants of these features.

- **Context level.** Models were trained on single words in isolation, or on entire sentences. Our goal was to investigate whether the additional context available to sentence-level sequence models would improve performance.

Word	Morphological analysis
aliqela	a[RelConc6]-li[BPre5]-qela[NStem]
kwibhunga	ku[LocPre]-i[NPrePre5]-(li)[BPre5]-bhunga[NStem]
izincomo	i[NPrePre10]-zin[BPre10]-como[NStem]

Table 1: Three examples from the isiXhosa part of the dataset used in our experiments (Gaustad and Puttkammer, 2022). Only the relevant aspects are included.

3.2 Pretrained language models

We finetune the following three PLMs on our task:

1. XLM-R-large (Conneau et al., 2020): a massively multilingual PLM trained on more than 100 languages, including isiXhosa.
2. Afro-XLMR-large (Alabi et al., 2022): XLM-R further pretrained on 20 African languages, including isiXhosa and isiZulu.
3. Nguni-XLMR-large (Meyer et al., 2024): XLM-R adapted for the four Nguni languages.

The models were selected to represent increasing levels of Nguni language pretraining coverage: XLM-R includes minimal Nguni data (only isiXhosa), Afro-XLMR adds isiZulu, while Nguni-XLMR specifically targets all four Nguni languages. We examine the degree to which these different levels of Nguni language inclusion influence downstream performance.

4 Experimental Setup

4.1 Dataset

We use the morphologically annotated dataset developed by Gaustad and Puttkammer (2022). It contains sentences from South African government publications, wherein each word is annotated with its morphological parse (segmentation and tags, as shown in Table 1), lemma, and part-of-speech. It contains 1,431 parallel paragraphs with roughly 50k words per language. The data is pre-split 90%/10% into train/test sets. The dataset contains only gold-standard canonical segmentations, so gold-standard surface segmentations were obtained through a script provided by Moeng et al. (2021). Predicted segmentations for both canonical and surface forms were created by applying Moeng et al.'s (Moeng et al., 2021) to the raw text column of the dataset.

4.2 Model Configurations

All our models are monolingually trained and evaluated on isiNdebele, isiXhosa, isiZulu, or Siswati. We evaluate four versions of each neural model, varying morphological input in the following ways.

Segmentation types We train models for both types of morphological segmentation, allowing us to evaluate their respective difficulty.

- Canonical segmentation: decompose words into standardised morphemes (e.g., "zobomi" → "za-u-(bu)-bomi").
- Surface segmentation: decompose words into morphs as they appear in composed forms (e.g., "zobomi" → "zo-bomi").

Upstream segmentation During testing, we assess performance across both idealised and practical scenarios.

- Gold-annotated segmentations: apply taggers directly to the linguistically annotated, gold-standard morphological segmentations from the task dataset (Gaustad and Puttkammer, 2022). This provides an idealised setting in which morphological segmentations are known to be correct, isolating tagging performance from segmentation errors.
- Model-predicted segmentations: apply taggers to segmentations generated by neural segmenters (Moeng et al., 2021). We retrain their feature-based CRFs and Transformers on our training set to match our data setup. This simulates a real-world pipeline where segmentation is predicted, allowing for error propagation from segmentation to tagging.

4.3 Evaluation

We use F_1 score to evaluate our models. We only evaluate morphological tagging performance, as opposed to full morphological parsing (segmentation + tagging). However, tagging inherently depends on segmentation in our setup, since models are trained on the pre-segmented morpheme sequences.

In our model-predicted segmentation setting, errors in predicted morphological segmentations can result in fewer or more predicted morphemes than morphological tags. As a result, in some instances we have to compute an F_1 score for predicted and target tag sequences of different lengths. We make use of the aligned multiset F_1 score proposed by Seker and Tsarfaty (2020b). This is an adaptation of the aligned segment F_1 score used in CoNLL18

Hyperparameter	Search space
Neural sequence labellers	
Learning rate	$[10^{-6}, 10^{-1}]$
Weight decay	$\{0\} \cup [10^{-10}, 10^{-3}]$
Hidden state size	$\{2^x : 6 \leq x \leq 11\}$
Dropout	$\{0, 0.1, 0.2, 0.3\}$
Gradient clip	$\{0.5, 1, 2, 4, \infty\}$
Finetuned PLMs	
Learning rate	$\{10^{-5}, 3 \times 10^{-5}, 5 \times 10^{-5}\}$
Epochs	$\{5, 10, 15\}$
Batch size	$\{8, 16, 32\}$

Table 2: The hyperparameter ranges of our grid search.

(Zeman and Hajič, 2018). The key difference is that the aligned multiset F_1 score bases token counts on the multiset intersection between the target and predicted sequences, so that target-prediction length mismatches are ameliorated.

We report both macro F_1 and micro F_1 in our results. Micro F_1 is a calculated by counting the number of true positives/negatives and false positives/negatives for all classes. More common tags therefore have a greater effect on the Micro F_1 score. With one tag per item, it is equivalent to accuracy. Macro F_1 calculates the per-class F_1 score and averages them, weighting all tags equally irrespective of frequency. A high macro F_1 score indicates good performance across all tags, including rare tag types. We focused on macro F_1 during hyperparameter tuning and in discussing our results, as we consider it important for our models to perform well on rare tags. Our evaluation dataset (Gaustad and Puttkammer, 2022) is imbalanced from a tag perspective, so macro F_1 is the more challenging metric to optimise than micro F_1.

4.4 Hyperparameters

The morphologically annotated dateset (Gaustad and Puttkammer, 2022) is split into train and test sets, but does not include a validation set. To prevent over-fitting hyperparameters to the test set, we created our own held-out validation set from 10% of the training set. Hyperparameter settings were tuned to maximise macro F_1 scores on the validation dataset.

For our models trained from scratch, we performed a grid search over the hyperparameter ranges shown in Table 2. We tuned our hyperparameter settings on isiZulu only, because including other languages would lead to a computationally infeasible grid search. Once the best parameters for isiZulu were found, these configurations were applied to the other languages. For our PLMs, we also performed a grid search over finetuning hyperparameters over the grid shown in Table 2.

After we settled on our final hyperparameter settings based on validation set performance, we retrained models on the full, original training set (including our newly created validation set) and evaluated them on the test set. For each model configuration, we train/finetune five models with different random seeds and report the average evaluation metrics.

4.5 Baselines

We compare our neural methods to ZulMorph (Bosch et al., 2008), a traditional, rule-based parser for IsiZulu. ZulMorph is based on finite-state transducers with manually incorporated grammatical rules, stems, and affixes for isiZulu. We use the ZulMorph demo (Pretorius and Bosch, 2018) to evaluate its performance on the test set. Since ZulMorph both segments and tags the input data, we compare it to our taggers trained on model-predicted segmentations.

5 Results

The results based on gold-annotated segmentations are shown in Table 3, while those based on model-predicted segmentations are shown in Table 4.

Overall, our results demonstrate the effectiveness of neural models on the challenging task of morphological tagging for Nguni languages. Our best-performing models based on gold-annotated canonical segmentations consistently achieve micro F_1 scores above 90% and macro F_1 scores above 60%. Even without access to the gold morphological annotations, with models tested on the predicted canonical segmentations of Moeng et al. (2021), our best models consistently achieve micro F_1 scores above 80% and macro F_1 scores above 55%. This confirms the feasibility of basing the full morphological parsing pipeline on neural models.

Comparison to rule-based parsing The neural models comfortably outperform our rule-based baseline, ZulMorph, on isiZulu morphological tagging. ZulMorph (Pretorius and Bosch, 2018) achieves a macro F_1 of 34% and micro F_1 of 71.8% on the test-set. All our isiZulu models surpass this performance, ranging from macro F_1s of 43.1% to 60% and micro F_1s of 72.7% to 85.8%.

Model	**IsiZulu**		**IsiNdebele**		**IsiXhosa**		**Siswati**	
	Mac F_1	Mic F_1	Mac F_1	Mic F_1	Mac F_1	Mic F_1	Mac F_1	Mic F_1
Canonical segmentations as annotated in Gaustad and Puttkammer (2022)								
Trained from scratch								
Word-level								
bi-LSTM, character-sum	**66.9**	**92.4**	67.2	**91.9**	72.3	95.2	66.5	91.2
bi-LSTM, morpheme	66.6	92.1	67.7	91.8	71.5	94.9	65.5	91.0
Sentence-level								
bi-LSTM, character-sum	64.6	91.6	66.6	91.0	72.1	95.5	64.7	90.8
bi-LSTM, morpheme	66.0	92.1	67.9	91.6	74.7	95.7	**67.2**	91.3
CRF, character-sum	65.7	92.1	67.3	91.4	74.7	**95.9**	66.0	91.4
CRF, morpheme	66.1	92.3	**68.1**	91.6	**75.3**	95.8	67.2	**91.4**
Pretrained language models								
Word-level								
Afro-XLMR	**62.5**	**92.0**	62.3	91.4	67.9	95.1	**63.3**	**91.3**
Nguni-XLMR	61.9	92.0	62.8	91.5	**68.1**	**95.1**	61.8	90.7
XLM-R-large	61.8	91.8	**63.6**	**91.6**	67.4	95.0	62.9	91.2
Surface segmentations extrapolated from Gaustad and Puttkammer (2022) by script from Moeng et al. (2021)								
Trained from scratch								
Sentence-level								
bi-LSTM, character-sum	63.3	90.7	65.2	90.4	73.6	94.7	61.3	89.6
bi-LSTM, character-sum-lower	63.2	90.8	65.4	90.4	73.7	94.7	60.8	89.7
bi-LSTM, morpheme	65.6	91.3	68.4	91.1	**76.1**	95.1	**65.9**	90.6
bi-LSTM, morpheme-lower	**66.0**	**91.3**	**68.7**	**91.2**	76.0	**95.3**	65.8	**90.7**
Pretrained language models								
Word-level								
Afro-XLMR	43.8	72.8	47.7	77.4	52.3	78.5	23.4	55.6
Nguni-XLMR	**44.1**	**73.1**	**48.1**	**77.5**	**52.4**	**79.0**	**23.9**	**56.6**
XLM-R-large	43.1	72.6	48.0	77.5	51.7	78.1	22.7	55.4

Table 3: Results for models evaluated on gold-annotated segmentations, given as percentages. This provides an idealised training setting in which all morphological segmentations are correct, allowing us to isolate the performance of morphological tagging. The best models for each approach (pretrained or from scratch) is **bolded**, while the best for each segmentation type (surface or canonical) is underlined.

Since ZulMorph is rule-based and contains manually-incorporated stems and affixes, it likely struggles to generalise to unseen data. For instance, ZulMorph failed to segment and parse "wezentuthuko", and instead produced "wezentuthuko +?". Conversely, the neural models do not explicitly incorporate any information. The models are able to classify text even when there are unknown morphemes present in the text, based on the surrounding context of known morphemes.

Macro vs Micro F_1 Macro F_1 is consistently lower than corresponding micro F_1 scores. This highlights one of the difficulties of morphological tagging for the Nguni languages. The tag set is large and unevenly distributed in the dataset, which make it challenging to accurately model rare tags. This imbalance would explain the mismatch between macro and micro F_1 for neural models, since they are not adequately exposed to rare tags during training. However, the mismatch persists for ZulMorph (Bosch et al., 2008) (see Table 4), which is based on gramatically informed rules, as opposed to being data-driven. This could indicate that some tags are inherently harder to disambiguate.

5.1 Training neural taggers from scratch

As shown in Tables 3 and 4, sentence-level models trained from scratch tended to outperform their word-level counterparts. Sentence-level models are trained on the entire sentence as context, which may allow them to use grammatical dependencies to improve tagging. For example, in the isiXhosa sentence "ipolisa liyahamba", the word "ipolisa" is in noun class 5. The shorted prefix "i" ("**i**polisa") is ambiguous and also appears in class 9 nouns, such as "iteksi". However, combining it with the subject concord for class 5 "li" ("**li**yahamba") provides the information required to correctly disambiguate and

Model	IsiZulu		IsiNdebele		IsiXhosa		Siswati	
	Mac F_1	Mic F_1	Mac F_1	Mic F_1	Mac F_1	Mic F_1	Mac F_1	Mic F_1
ZulMorph online demo (Pretorius and Bosch, 2018)								
ZulMorph	34.0	71.8						
Canonical segmentations as predicted by Moeng et al. (2021)								
Trained from scratch								
Word-level								
bi-LSTM, character-sum	**60.0**	**85.8**	57.8	**84.1**	67.9	92.3	57.0	85.0
bi-LSTM, morpheme	58.3	85.5	58.3	84.1	67.0	92.2	55.7	84.7
Sentence-level								
bi-LSTM, character-sum	57.5	85.1	57.3	83.4	68.1	92.7	55.5	84.8
bi-LSTM, morpheme	58.4	85.7	58.3	83.8	70.7	93.0	57.3	85.2
CRF, character-sum	58.1	85.5	58.4	83.8	69.8	**93.1**	57.2	**85.4**
CRF, morpheme	58.7	85.7	**58.5**	83.7	**71.1**	93.1	**57.8**	85.3
Pretrained language models								
Word-level								
Afro-XLMR	**55.3**	85.5	54.6	84.0	63.4	92.4	**53.4**	**85.1**
Nguni-XLMR	54.8	**85.5**	54.5	83.9	**64.4**	**92.6**	52.5	84.6
XLM-R-large	54.4	85.4	**55.4**	**84.1**	63.5	92.5	52.9	85.0
Surface segmentations as predicted by Moeng et al. (2021)								
Trained from scratch								
Sentence-level								
bi-LSTM, character-sum	53.6	79.6	52.8	78.3	65.6	**87.7**	51.8	80.4
bi-LSTM, character-sum-lower	53.3	79.6	52.9	78.2	65.2	87.5	51.6	80.4
bi-LSTM, morpheme	55.0	79.7	**54.7**	78.4	68.0	87.4	55.2	81.0
bi-LSTM, morpheme-lower	**55.3**	**79.7**	54.6	**78.5**	**68.2**	87.6	**55.8**	**81.0**
Pretrained language models								
Word-level								
Afro-XLMR	43.6	72.8	46.9	77.4	**51.9**	78.5	23.0	55.7
Nguni-XLMR	**43.9**	**73.0**	46.9	77.4	51.7	**78.8**	**23.7**	**56.3**
XLM-R-large	43.1	72.7	**47.7**	**77.5**	51.4	78.0	22.1	55.4

Table 4: Results for models evaluated on model-predicted segmentations, given as percentages. This evaluates the combined use of neural methods for segmentation and tagging, without access to morphological annotations. The best models for each approach (pretrained or from scratch) is **bolded**, while the best for each segmentation type (surface or canonical) is underlined.

tag "ipolisa" as class 5.

Morpheme-level embeddings outperformed character-summing embeddings. While one might expect character-level modelling to improve generalisation across morphemes, this is not necessarily the case. Morphemes representations have previously been shown to be highly effective for syntactic tasks (Üstün et al., 2018). For our task, morpheme-level embeddings allow the model to be more sensitive to small changes in morphemes. For example, the morphemes "ng" and "nga" differ by a single character, but can have totally different meanings ("ng" can be a copulative prefix and "nga" can be an adverb prefix). With character-summed representations, the two morphemes will have highly similar embeddings. With morpheme-level embeddings, each morpheme embedding is learned separately. For rare or previously unseen morphemes, the morpheme-level model is forced to rely on contextual grammatical information (within the word or surrounding sentence), which provides a more reliable grammatical signal than the number of overlapping characters between morphemes.

We do not find substantial performance differences between bi-LSTMs and bi-LSTM CRFs. This indicates that explicitly modeling grammar through tag dependence presents limited advantage. Bi-LSTMs are able to encode such grammatical dependencies, based on morpheme co-occurrence patterns, in their hidden representations.

5.2 Pretrained language models

As shown in Tables 3 and 4, training models from scratch outperformed finetuning PLMs. This contrasts with previous work on linguistic annotation tasks, in which pretrained solutions have outperformed models trained from scratch (Min et al., 2023; Alabi et al., 2022). However, it does align with related work for the Nguni languages, which have achieved high performance levels with neural models trained from scratch (Moeng et al., 2021; Pannach et al., 2022).

Due to computational constraints, we did not finetune PLMs on sentence-level input. The pretrained contextual representations of PLMs are well suited to take advantage of sentence-level context, so it is possible that finetuning sentence-level versions of our PLMs could improve their performance. We leave the exploration of sentence-level PLMs for Nguni-language morphological tagging to future work.

Another factor which could contribute to PLM performance degradation is subword tokenisation. While our models trained from scratch use character or morpheme-level representations, our PLMs are constrained to finetune representations for the subword tokens produced by their pretrained tokenisers. In pretraining, the tokeniser segments raw words. In finetuning, the tokeniser segments pre-segmented morphemes. This misalignment could impede the model's ability to leverage pretrained knowledge during finetuning, since the subword tokens learned in pretraining do not match those of finetuning. This also leads to irregular, morphologically unsound subword tokens. For example, the XLM-R SentencePiece tokeniser (Conneau et al., 2020; Kudo and Richardson, 2018) segments, which is the tokeniser for all our PLMs, segments the isiXhosa morpheme "-bandela" into "-ba", "#ndel", "#a", which is morphologically meaningless. In our pipeline setup for morphological parsing, it is not obvious how to bridge the mismatch between pretraining and finetuning subword tokenisation. It should be viewed as a limitation of PLMs. With neural models trained from scratch, we have the freedom to design our own morphological input features.

5.3 Models based on surface segmentations

In both Tables 3 and 4, the top half of each table reports results for models trained on canonical segmentations (morphemes), while the bottom half reports results for surface-level segmentations (morphs). In general, canonically-based tagging scores are higher than surface-level tagging. The performance gap is particularly notable and consistent for models trained on model-predicted segmentations. While canonical and surface-level tagging scores cannot be directly compared (for some words, the tag sequence will not be the same), our results clearly show that training taggers on top of canonical segmenters is more effective than doing so with surface-level segmenters. We attribute this to two factors.

Firstly, the surface segmentation of a word provides less grammatical information to models than the canonical segmentation. For instance, the word "kwicandelo" is canonically segmented as "ku-i-(li)-candelo" and surface segmented as "kw-i-candelo" (Gaustad and Puttkammer, 2022). Critically, the "(li)" morpheme is lost, which is part of the noun prefix for class 5. The only morpheme left for the noun prefix is thus "i". However, this on its own is ambiguous, and could be the noun prefix for class 5 or class 9. In this case, the canonical tagger would have more information relevant to the tagging decision than the surface tagger.

Secondly, there is often a length mismatch between the surface and canonical morphemes in a word. For example, "kubomi" is canonically segmented into "ku-u-(bu)-bomi", but surface-segmented into "ku-bomi". We evaluate our model on gold-annotated data, which include morphological tags for each word. In a case like "kubomi", this would limit performance to 50% accuracy in the best case scenario. In general, this length mismatch limits the performance of models based surface-level segmentations.

6 Conclusion

In this paper, we explored the feasibility of neural morphological taggers for the Nguni languages. We divide morphological parsing into two subtasks, segmentation and tagging, focussing on the latter. We investigate bi-LSTMs and CRFs trained from scratch, as well as finetuned PLMs. Our neural models comfortably outperform a rule-based baseline, while our models trained from scratch outperform PLMs. Models based on canonical segmentations outperform their surface-level counterparts.

We identify several promising directions for future research to build on our findings. Firstly, our PLM taggers could potentially be improved, either by finetuning on sentence-level input or by exploring ways to align the mismatch between subword tokenisation in pretraining and finetuning. Furthermore, our parsers can be used to incorporate morphological information into downstream task models (Klemen et al., 2023). This has been shown to improve performance in tasks such as language modelling (Nzeyimana and Niyongabo Rubungo, 2022) and machine translation (Nzeyimana, 2024), but has not been explored for the Nguni languages.

Limitations

Our study is limited to the Nguni languages, so our findings may not generalise to other language families or typologies like the Sotho-Tswana languages whose morphology is disjunctive. Further experimentation is needed to validate whether training taggers on model-predicted morphological segmentations is viable for languages with different morphological structures. That being said, the promising performance of our models on the Nguni languages suggests that similar neural approaches could be beneficial for other low-resource, morphologically complex languages.

Additionally, while our models trained from scratch consistently outperformed finetuned PLMs, we do not definitively conclude that PLMs are inferior for this task. As discussed in subsection 5.2, because of computational constraints we did not test sentence-level PLMs. Incoporating sentence-level context could improve PLM performance to be competitive with models trained from scratch. We would need to run further experiments with sentence-level finetuning to evaluate the full potential of PLMs for this task.

References

Gulinigeer Abudouwaili, Kahaerjiang Abiderexiti, Nian Yi, and Aishan Wumaier. 2023. Joint learning model for low-resource agglutinative language morphological tagging. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 27–37, Toronto, Canada. Association for Computational Linguistics.

Ekin Akyürek, Erenay Dayanık, and Deniz Yuret. 2019. Morphological analysis using a sequence decoder. *Transactions of the Association for Computational Linguistics*, 7:567–579.

Jesujoba O. Alabi, David Ifeoluwa Adelani, Marius Mosbach, and Dietrich Klakow. 2022. Adapting pre-trained language models to African languages via multilingual adaptive fine-tuning. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4336–4349, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Hüseyin. Aleçakır. 2020. Joint learning of morphological segmentation, morpheme tagging, part-of-speech tagging, and dependency parsing. Master's thesis, Middle East Technical University.

Anders Björkelund, Özlem Çetinoğlu, Richárd Farkas, Thomas Mueller, and Wolfgang Seeker. 2013. (re) ranking meets morphosyntax: State-of-the-art results from the spmrl 2013 shared task. In *Proceedings of the Fourth Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 135–145.

Sonja Bosch, Laurette Pretorius, Kholisa Podile, and Axel Fleisch. 2008. Experimental fast-tracking of morphological analysers for nguni languages. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Paul G Chapin and Lewis M Norton. 1968. A procedure for morphological analysis.

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, Online. Association for Computational Linguistics.

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, San Diego, California. Association for Computational Linguistics.

W. Daelemans, J. Zavrel, K. Sloot, and A. Bosch. Timbl: Tilburg memory-based learner, version 6.4: reference guide.

David M. Eberhard, Gary F. Simons, , and Charles D. Fenning. 2019. *Ethnologue: Languages of the World*, 22 edition. SIL International.

Tanja Gaustad and Martin J. Puttkammer. 2022. Linguistically annotated dataset for four official south african languages with a conjunctive orthography: Isindebele, isixhosa, isizulu, and siswati. *Data in Brief*, 41:107994.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Matej Klemen, Luka Krsnik, and Marko Robnik-Šikonja. 2023. Enhancing deep neural networks with morphological information. *Natural Language Engineering*, 29(2):360–385.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *CoRR*, abs/1808.06226.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, page 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California. Association for Computational Linguistics.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074, Berlin, Germany. Association for Computational Linguistics.

Peter H. Matthews. 1991. *Morphology*, 2 edition. Cambridge Textbooks in Linguistics. Cambridge University Press, Cambridge, UK.

Francois Meyer, Haiyue Song, Abhisek Chakrabarty, Jan Buys, Raj Dabre, and Hideki Tanaka. 2024. NGLUEni: Benchmarking and adapting pretrained language models for nguni languages. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 12247–12258, Torino, Italia. ELRA and ICCL.

Bonan Min, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heintz, and Dan Roth. 2023. Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Comput. Surv.*, 56(2).

Tumi Moeng, Sheldon Reay, Aaron Daniels, and Jan Buys. 2021. Canonical and surface morphological segmentation for nguni languages. *Preprint*, arXiv:2104.00767.

Thomas Mueller, Helmut Schmid, and Hinrich Schütze. 2013. Efficient higher-order CRFs for morphological tagging. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 322–332, Seattle, Washington, USA. Association for Computational Linguistics.

Antoine Nzeyimana. 2024. Low-resource neural machine translation with morphological modeling. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 182–195, Mexico City, Mexico. Association for Computational Linguistics.

Antoine Nzeyimana and Andre Niyongabo Rubungo. 2022. KinyaBERT: a morphology-aware Kinyarwanda language model. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5347–5363, Dublin, Ireland. Association for Computational Linguistics.

Franziska Pannach, Francois Meyer, Edgar Jembere, and Sibonelo Zamokuhle Dlamini. 2022. Nlapost2021 1st shared task on part-of-speech tagging for nguni languages. *Journal of the Digital Humanities Association of Southern Africa*, 3(01).

L. Pretorius and S. Bosch. 2018. Zulmorph: Finite state morphological analyser for zulu (version 20190103. [Software]. Web demo at.

Martin Puttkammer and Jakobus Du Toit. 2021. Canonical segmentation and syntactic morpheme tagging of four resource- scarce nguni languages. *Journal of the Digital Humanities Association of Southern Africa (DHASA)*, 3.

Amit Seker and Reut Tsarfaty. 2020a. A pointer network architecture for joint morphological segmentation and tagging. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4368–4378, Online. Association for Computational Linguistics.

Amit Seker and Reut Tsarfaty. 2020b. A pointer network architecture for joint morphological segmentation and tagging. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4368–4378, Online. Association for Computational Linguistics.

Elsabe Taljard and Sonja Bosch. 2006. A comparison of approaches to word class tagging: Disjunctively vs. conjunctively written bantu languages. *Nordic Journal of African Studies*, 15.

Reut Tsarfaty, Djamé Seddah, Sandra Kübler, and Joakim Nivre. 2013. Parsing Morphologically Rich Languages: Introduction to the Special Issue. *Computational Linguistics*, 39(1):15–22.

Ahmet Üstün, Murathan Kurfalı, and Burcu Can. 2018. Characters or morphemes: How to represent words? In *Proceedings of the Third Workshop on Representation Learning for NLP*, pages 144–153, Melbourne, Australia. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need. *Preprint*, arXiv:1706.03762.

Danit Yshaayahu Levi and Reut Tsarfaty. 2024. A truly joint neural architecture for segmentation and parsing. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1408–1420, St. Julian's, Malta. Association for Computational Linguistics.

Daniel Zeman and Jan Hajič, editors. 2018. *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. Association for Computational Linguistics, Brussels, Belgium.