KinyaBERT: a Morphology-aware Kinyarwanda Language Model

Anonymous ACL submission

Abstract

Pre-trained language models such as BERT have been successful at tackling many natural language processing tasks. However, the unsupervised sub-word tokenization methods commonly used in these models (e.g., byte-pair encoding – BPE) are sub-optimal at handling morphologically rich languages. Even given a morphological analyzer, naive sequencing of morphemes into a standard BERT architecture is inefficient at capturing morphological compositionality and expressing word-relative syntactic regularities. We address these challenges by proposing a simple two-tier BERT architecture that leverages a morphological analyzer and explicitly represents morphological compositionality. Despite the success of BERT, most of its evaluations have been conducted on high-resource languages, obscuring its applicability on low-resource languages. We evaluate our proposed method on the low-resource morphologically rich Kinyarwanda language, naming the proposed model architecture *KinyaBERT*. A robust set of experimental results reveal that *KinyaBERT* outperforms solid baselines by 2% F1 score on a named entity recognition task and by 4.3% average score of a machine-translated GLUE benchmark. *KinyaBERT* fine-tuning has better convergence and achieves more robust results on multiple tasks even in the presence of translation noise. Code and datasets are released at `https://anonymous.4open.science/r/kinyabert-acl`

1 Introduction

Recent advances in natural language processing (NLP) through deep learning have been largely enabled by vector representations (or embeddings) learned through language model pre-training (Bengio et al., 2003; Mikolov et al., 2013; Pennington et al., 2014; Bojanowski et al., 2017; Peters et al., 2018; Devlin et al., 2019). Language models such as BERT are pre-trained on large text corpora and then fine-tuned on downstream tasks, resulting in better performance on many NLP tasks. Despite attempts to make multilingual BERT models (Conneau et al., 2020), research has shown that models pre-trained on high quality monolingual corpora outperform multilingual models pre-trained on large Internet data (Scheible et al., 2020; Virtanen et al., 2019). This has motivated many researchers to pre-train BERT models on individual languages rather than adopting the "language-agnostic" multilingual models. This work is partly motivated by the same findings, but also proposes an adaptation of the BERT architecture to address other challenges that are specific to low resource morphologically-rich languages such as Kinyarwanda.

In order to handle rare words and reduce the vocabulary size, BERT-like models use statistical sub-word tokenization algorithms such as byte pair encoding (BPE) (Sennrich et al., 2015). While these techniques have been widely used in language modeling and machine translation, they are not optimal for morphologically rich languages. In fact, BPE cannot efficiently handle non-concatenative morphology because it is solely based on the surface forms of words. For example, as shown in Table 1, a BPE model trained on 390 million tokens of Kinyarwanda text cannot extract the true sub-word lexical units (i.e. morphemes) for the given words. This work addresses the above problem by proposing a language model architecture that explicitly represents most of the input words with morphological parses produced by a morphological analyzer. In this architecture BPE is only used to handle words which cannot be decomposed by the morphological analyzer such as misspellings and foreign language words.

Given the output of a morphological analyzer, a second challenge is in how to incorporate the produced morphemes into the model. One naive approach is to feed the produced morphemes to a

Word	Morphemes	Monolingual BPE	Multilingual BPE
twagezeyo '*we arrived there*'	**tu . a . ger . ye . yo**	twag . ezeyo	_twa . ge . ze . yo
ndabyizeye '*I hope so*'	**n . ra . bi . izer . ye**	ndaby . izeye	_ ndab . yiz . eye
umwarimu '*teacher*'	**u . mu . arimu**	umwarimu	_um . wari . mu

Table 1: Comparison between morphemes and BPE-produced sub-word tokens. Stems are underlined.

standard transformer encoder as a single monolithic sequence. This approach is used in (Mohseni and Tebbifakhr, 2019). One problem with this method is that mixing sub-word information and sentence-level tokens in a single sequence does not encourage the model to learn the actual morphological compositionality. Another problem is that position encoding mechanisms used in BERT might become less effective due to the large number of morphemes appearing everywhere in the sequence. We hypothesize that this mixing might make it difficult to learn sentence-level syntactic regularities that would otherwise benefit from relative position information between different parts of speech (POS). We address these issues by proposing a simple yet effective two-tier transformer encoder architecture for expressing morphological compositionality. The first tier encodes morphological information, which is then transferred to the second tier to encode sentence level information. We call this new model architecture KinyaBERT because it uses BERT's masked language model objective for pre-training and is evaluated on the morphologically rich Kinyarwanda language.

This work also represents progress in low resource NLP. Advances in human language technology are most often evaluated on the main languages spoken by major economic powers such as English, Chinese and European languages. This has exacerbated the language technology divide between the highly resourced languages and the underrepresented languages. It also hinders progress in NLP research because new techniques are mostly evaluated on the mainstream languages and some NLP advances become less informed of the diversity of the linguistic phenomena (Bender, 2019). Specifically, this work provides the following research contributions:

- A simple yet effective two-tier BERT architecture for representing morphologically-rich languages.
- New evaluation datasets for Kinyarwanda language including a machine-translated subset of the GLUE benchmark (Wang et al., 2018) and a news categorization dataset.
- Experimental results which set a benchmark for future studies on Kinyarwanda language understanding, and on using machine-translated versions of the GLUE benchmark.
- Code and datasets that are made publicly available for reproducibility[1].

2 Morphology-aware Language Model

Our modeling objective is to be able to express morphological compositionality in a Transformer-based (Vaswani et al., 2017) language model. For morphologically rich languages such as Kinyarwanda, a set of morphemes (typically a stem and a set of functional affixes) combine to produce a word with a given surface form. This requires an alternative to the ubiquitous BPE tokenization, through which exact sub-word lexical units (i.e. morphemes) are used. For this purpose, we use a morphological analyzer which takes a sentence as input and, for every word/token, produces a stem, zero or more affixes and assigns a POS tag to each word/token. This section describes how this morphological information is obtained and then integrated in a two-tier transformer architecture (Figure 1) to learn morphology-aware input representations.

2.1 Morphological Analysis and Part-of-Speech Tagging

Our morphological analyzer for Kinyarwanda was built following finite-state two-level morphology principles (Koskenniemi, 1983; Beesley and Karttunen, 2000, 2003). For every inflectable word type, we maintain a morphotactics model using a directed acyclic graph (DAG) that represents the regular sequencing of morphemes. We effectively model all inflectable word types in Kinyarwanda which include verbals, nouns, qualitative adjectives,

[1] https://anonymous.4open.science/r/kinyabert-acl

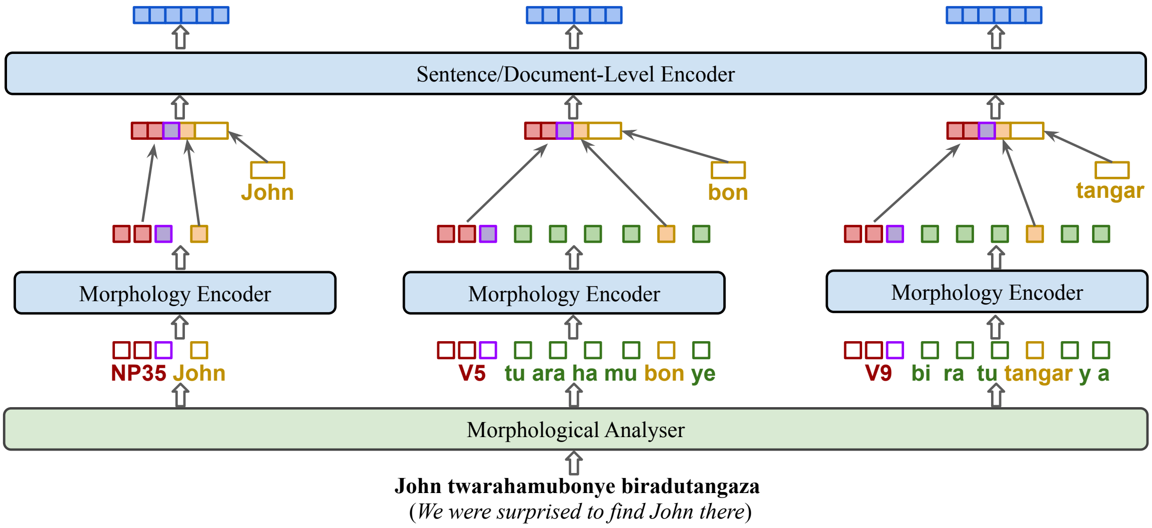


Figure 1: KinyaBERT model architecture: Encoding of the sentence 'John twarahamusanze biradutangaza' (*We were surprised to find John there*). The morphological analyzer produces morphemes for each word and assigns a POS tag to it. The two-tier transformer model then generates contextualized embeddings (**blue** vectors at the top). The **red** colored embeddings correspond to the POS tags, **yellow** is for the stem embeddings, **green** is for the variable length affixes while the **purple** embeddings correspond to the affix set.

possessive and demonstrative pronouns, numerals and quantifiers. The morphological analyzer also includes many hand-crafted rules for handling morphographemics and other linguistic regularities of the Kinyarwanda language. Similar to (Nzeyimana, 2020), we use a classifier trained on a stemming dataset to disambiguate between competing outputs of the morphological analyzer. Furthermore, we improve the disambiguation quality by leveraging a part-of-speech (POS) tagger at the phrase level so that the syntactic context can be taken into consideration.

We devise an unsupervised part-of-speech tagging algorithm which we explain next. Let $x = (x_1, x_2, x_3, ...x_n)$ be a sequence of tokens (e.g. words) to be tagged with a corresponding sequence of tags $y = (y_1, y_2, y_3, ...y_n)$. A sample of actual POS tags used for Kinyarwanda is given in the Appendix. Using Bayes' rule, the optimal tag sequence y^* is given by the following equation:

$$\begin{aligned} y^* &= \arg\max_y P(y|x) \\ &= \arg\max_y \frac{P(x|y)P(y)}{P(x)} \\ &= \arg\max_y P(x|y)P(y) \end{aligned} \tag{1}$$

A standard hidden Markov model (HMM) can decompose the result of Equation 1 using first order Markov assumption and independence assumptions into $P(x|y) = \prod_{t=1}^{n} P(x_t|y_t)$ and $P(y) = \prod_{t=1}^{n} P(y_t|y_{t-1})$. The tag sequence y^* can then be efficiently decoded using the Viterbi algorithm (Forney, 1973). A better decoding strategy is presented below.

Inspired by (Tsuruoka and Tsujii, 2005), we devise a greedy heuristic for decoding y^* using the same first order Markov assumptions but with bidirectional decoding.

First, we estimate the local emission probabilities $P(x_t|y_t)$ using a factored model given in the following equation:

$$\begin{aligned} &P(x_t|y_t) \propto \tilde{P}(x_t|y_t) \\ &\tilde{P}(x_t|y_t) = \tilde{P}_m(x_t|y_t)\tilde{P}_p(x_t|y_t)\tilde{P}_a(x_t|y_t) \end{aligned} \tag{2}$$

In Equation 2, $\tilde{P}_m(x_t|y_t)$ corresponds to the probability/score returned by a morphological disambiguation classifier, representing the uncertainty of the morphology of x_t. $\tilde{P}_p(x_t|y_t)$ corresponds to a local precedence score between competing POS tags. These precedence weights are manually crafted through qualitative evaluation. $\tilde{P}_a(x_t|y_t)$ quantifies the local neighborhood syntactic agreement between Bantu class markers. Like most Bantu languages, Kinyarwanda has 16 class markers (KIMENYI, 1978) that are included in in nouns, verbs, adjectives and pronouns. We leverage this agreement information to improve disambiguation. When there are two or more agreeing class markers in neighboring words, the tagger should be more confident of the agreeing parts of speech. Each of the above unnormalized measures $\tilde{P}$ is mapped to the $[0, 1]$ range using a sigmoid function

$\sigma(z|z_A, z_B)$ given in Equation 3, where z is the score of the measure and $[z_A, z_B]$ is its estimated active range.

$$\sigma(z|z_A, z_B) = [1 + exp(-8\frac{z - z_A}{z_B - z_A})]^{-8} \quad (3)$$

After estimating the local emission model, we greedily decode $y_t^* = \arg\max y_t \tilde{P}(y_t|x)$ in decreasing order of $\tilde{P}(x_t|y_t)$ using a first order bidirectional inference of $\tilde{P}(y_t|x)$ as given in the following equation:

$$\tilde{P}(y_t|x) = \begin{cases} \tilde{P}(x_t|y_t)\tilde{P}(y_t|y_{t-1}^*, y_{t+1}^*)\tilde{P}(y_{t-1}^*|x)\tilde{P}(y_{t+1}^*|x) \\ \qquad \text{if both } y_{t-1}^* \text{ and } y_{t+1}^* \text{ have been decoded;} \\ \tilde{P}(x_t|y_t)\tilde{P}(y_t|y_{t-1}^*)\tilde{P}(y_{t-1}^*|x) \\ \qquad \text{if only } y_{t-1}^* \text{ has been decoded;} \\ \tilde{P}(x_t|y_t)\tilde{P}(y_t|y_{t+1}^*)\tilde{P}(y_{t+1}^*|x) \\ \qquad \text{if only } y_{t+1}^* \text{ has been decoded;} \\ \tilde{P}(x_t|y_t) \text{ otherwise} \end{cases} \quad (4)$$

The first order transition measures $\tilde{P}(y_t|y_{t-1})$, $\tilde{P}(y_t|y_{t+1})$ and $\tilde{P}(y_t|y_{t-1}, y_{t+1})$ are estimated using count tables computed over the entire corpus by aggregating local emission marginals $\tilde{P}(y_t) = \sum_{x_t} \tilde{P}(x_t, y_t)$ obtained by morphological analysis and disambiguation.

2.2 Morphology Encoding

The overall architecture of our model is depicted in Figure 1. This is a two-tier transformer encoder architecture made of a token-level morphology encoder that feeds into a sentence/document-level encoder. The morphology encoder is made of a small transformer encoder that is applied to each analyzed token separately in order to extract its morphological features. The extracted morphological features are then concatenated with the token's stem embedding to form the input vector fed to the sentence/document encoder. The sentence/document encoder is made of a standard transformer encoder as used in other BERT models. The sentence/document encoder uses untied position encoding with relative bias as proposed in (Ke et al., 2020).

The input to the morphology encoder is a set of embedding vectors, 3 vectors relating to the part-of-speech, 1 vector for the stem and 1 vector for each affix when available. The transformer encoder operation is applied to these embedding vectors without any positional information, in a "bag-of-tokens" fashion. This is due to the fact that positional information at the morphology level is inherent because no morpheme repeats and each morpheme always occupies a known(i.e. fixed) morpheme slot in the morphotactics model. The extracted morphological features are the 4 encoder output vectors corresponding to the 3 POS embeddings and 1 stem embedding. Vectors corresponding to the affixes are left out since they are of variable length and the affixes role is to be attended to by the stem and the part-of-speech so that morphological information can be captured. The 4 morphological output feature vectors are further concatenated with another stem embedding at the sentence level to form the input vector for the main sentence/document encoder.

The choice of this transformer-based architecture for morphology encoding is motivated by two factors. First, (Zaheer et al., 2020) has demonstrated the importance of having "global tokens" such as `[CLS]` token in BERT models. These are tokens that attend to all other tokens in the modeled sequence. These "global tokens" effectively encapsulate some "meaning" of the encoded sequence. Second, the POS tag and stem represent the high level information content of a word. Therefore, having the POS tag and stem embeddings be transformed into morphological features is a viable option. The POS tag and stem embeddings thus serve as the "global tokens" at the morphology encoder level since they attend to all other morphemes that can be associated with them.

In order to capture subtle morphological information, we make one of the 3 POS embeddings span an affix set that is a subset of all affixes power set. We form an affix set vocabulary $\mathcal{V}_a$ that is made of N most frequent affix combinations in the corpus. In fact, the morphological model of the language enforces constraints on which affixes can go together for any given part-of-speech, resulting in an affix set vocabulary that is much smaller than the power set of all affixes. Even with limiting the affix set vocabulary $\mathcal{V}_a$ to a fixed size, we can still map any affix combination to $\mathcal{V}_a$ by dropping zero or very few affixes from the combination. Note that the affix set embedding still has to attend to all morphemes at the morphology encoder level, making it adapt to the whole morphological context. The affix set embedding is depicted by the **purple** units in Figure 1.

2.3 Pre-training Objective

Similar to other other BERT models, we use a masked language model objective. Specifically, 15% of all tokens in the training set are considered for prediction, of which 80% are replaced with [MASK] tokens, 10% are replaced with random tokens and 10% are left unchanged. When prediction tokens are replaced with [MASK] or random tokens, the corresponding affixes are randomly omitted 70% of the time or left in place for the other 30% of the time while the units corresponding to POS tags and affix sets are also masked. The pre-training objective is then to predict stems and the associated affixes for all tokens considered for prediction using a two-layer feed-forward module on top of the encoder output.

For the affix prediction task, we face a multi-label classification problem where for each prediction token, we predict a variable number of affixes. In our experiments, we tried two methods. For one, we use the Kullback–Leibler (KL) divergence loss function to solve regression task of the N-length continuous affix distribution vector. For this case, we use a target affix probability vector $\boldsymbol{a_t} \in \mathbb{R}^N$ in which each target affix index is assigned $\frac{1}{m}$ probability and 0 probability for non-target affix indices, where m is the total number of target affixes and N is the total number of all affixes in the language. We call this method "Affix Distribution Regression" (ADR) and model variant KinyaBERT$_{ADR}$. Alternatively, we use cross entropy loss and just predict the affix set associated with each word; we call this method "Affix Set Classification" (ASC) and the model variant KinyaBERT$_{ASC}$.

3 Experiments

In order to evaluate the proposed architecture, we pre-train KinyaBERT (101M parameters for KinyaBERT$_{ADR}$ and 129M for KinyaBERT$_{ASC}$) on a 2.4 GB of Kinyarwanda text along with 3 baseline BERT models. The first baseline is a BERT model pre-trained on the same Kinyarwanda corpus and with the same position encoding (Ke et al., 2020), same batch size and pre-training steps, but using the standard BPE tokenization. We call this first baseline model BERT$_{BPE}$ (120M parameters). The second baseline is a similar BERT model pre-trained on the same Kinyarwanda corpus but tokenized by a morphological analyzer. For this model, the input is just a sequence of morphemes, in a similar fashion to (Mohseni and Tebbifakhr, 2019). We call this second baseline model BERT$_{MORPHO}$ (127M parameters). For BERT$_{MORPHO}$, we found that predicting 30% if the tokens achieves better results than using 15% because of the many affixes generated. The third baseline is XLM-R (Conneau et al., 2020) (270M parameters) which is pre-trained on 2.5 TB of multilingual text. We evaluate the above models by comparing their performance on downstream NLP tasks.

Language	Kinyarwanda
Publication Period	2011 - 2021
Websites/Sources	370
Documents/Articles	840K
Sentences	16M
Tokens/Words	390M
Text size	2.4 GB

Table 2: Summary of the pre-training corpus.

3.1 Pre-training details

KinyaBERT model was implemented using Pytorch version 1.9. The morphological analyzer and part-of-speech tagger were implemented in a shared library using POSIX C. Morphological parsing of the corpus was performed as a pre-processing step, taking 20 hours to segment the 390M-token corpus on an 12-core desktop machine. Pre-training was performed using RTX 3090 and RTX 2080Ti desktop GPUs. Each KinyaBERT model takes on average 22 hours to train for 1000 steps on one RTX 3090 GPU or 29 hours on one RTX 2080Ti GPU. Baseline models (BERT$_{BPE}$ and BERT$_{MORPHO}$) were pre-trained on cloud tensor processing units (TPU v3-8 devices each with 128 GB memory) using an PyTorch/XLA package and a TPU-optimized fairseq toolkit (Ott et al., 2019). Pre-training on TPU took 2.3 hours per 1000 steps. The baselines were trained on TPU because there were no major changes needed to the existing Roberta(base) architecture implemented in fairseq and the TPU resources were available and efficient. In all cases, pre-training batch size was set to 2560 sequences, with maximum 512 tokens in each sequence. The maximum learning rates was set to 4×10^{-4} which is achieved after 2000 steps and then linearly decays to 0 at targeted 200K steps. Our main results and ablation results were obtained from models pre-trained for 32K steps in all cases. Other pre-training details, model architectural dimensions and other hyper-parameters are given in the Appendix.

Task:	MRPC	QNLI	RTE	SST-2	STS-B	WNLI
#Train examples:	**3.4K**	**104.7K**	**2.5K**	**67.4K**	**5.8K**	**0.6K**
Translation score:	**2.7/4.0**	**2.9/4.0**	**3.0/4.0**	**2.7/4.0**	**3.1/4.0**	**2.9/4.0**
Model	**Validation Set**					
XLM-R	84.2/78.3±0.8/1.0	79.0±0.3	58.4±3.2	78.7±0.6	77.7/77.8±0.7/0.6	55.4±2.0
BERT$_{BPE}$	83.3/76.6±0.8/1.4	81.9±0.2	59.2±1.5	80.1±0.4	75.6/75.7±7.8/7.3	55.4±1.9
BERT$_{MORPHO}$	84.3/77.4±0.6/1.1	81.6±0.2	59.2±1.5	81.6±0.5	76.8/77.0±0.8/0.7	54.2±2.5
KinyaBERT$_{ADR}$	**87.1/82.1**±0.5/0.7	81.6±0.1	61.8±1.4	81.8±0.6	79.6/79.5±0.4/0.3	54.5±2.2
KinyaBERT$_{ASC}$	86.6/81.3±0.5/0.7	**82.3**±0.3	**64.3**±1.4	**82.4**±0.5	**80.0/79.9**±0.5/0.5	**56.2**±0.8
Model	**Test Set**					
XLM-R	82.6/76.0±0.6/0.6	78.1±0.3	56.4±3.2	76.3±0.4	69.5/68.9±1.0/1.1	63.7±3.9
BERT$_{BPE}$	82.8/76.2±0.6/0.8	81.1±0.3	55.6±2.8	79.1±0.4	68.9/67.8±1.8/1.7	63.4±4.1
BERT$_{MORPHO}$	82.7/75.4±0.8/1.3	80.8±0.4	56.7±1.0	80.7±0.5	68.9/67.8±1.5/1.3	65.0±0.3
KinyaBERT$_{ADR}$	84.4/**78.7**±0.5/0.6	81.2±0.3	58.1±1.1	80.9±0.5	73.2/72.0±0.4/0.3	65.1±0.0
KinyaBERT$_{ASC}$	**84.6**/78.4±0.2/0.3	**82.2**±0.6	**58.8**±0.7	**81.4**±0.6	**74.5/73.5**±0.2/0.2	65.0±0.2

Table 3: Performance results on the machine translated GLUE benchmark (Wang et al., 2018). The translation score is the sample average translation quality score assigned by volunteers. For MRPC, we report accuracy and F1. For STS-B, we report Pearson and Spearman correlation. For all others, we report accuracy. The best results are shown in **bold** while equal top results are underlined.

Task:	NER	
#Train examples:	**2.1K**	
Model	**Validation Set**	**Test Set**
XLM-R	80.3±1.0	71.8±1.5
BERT$_{BPE}$	83.4±0.9	74.8±0.8
BERT$_{MORPHO}$	83.2±0.9	72.8±0.9
KinyaBERT$_{ADR}$	**87.1**±0.8	**77.2**±1.0
KinyaBERT$_{ASC}$	86.2±0.4	76.3±0.5

Table 4: Micro average F1 scores on Kinyarwanda NER task (Adelani et al., 2021).

Task:	NEWS	
#Train examples:	**18.0K**	
Model	**Validation Set**	**Test Set**
XLM-R	83.8±0.3	84.0±0.2
BERT$_{BPE}$	87.6±0.4	**88.3**±0.3
BERT$_{MORPHO}$	86.9±0.4	86.9±0.3
KinyaBERT$_{ADR}$	**88.8**±0.3	88.0±0.3
KinyaBERT$_{ASC}$	88.4±0.3	88.0±0.2

Table 5: Accuracy results on Kinyarwanda NEWS categorization task.

3.2 Evaluation tasks

Machine translated GLUE benchmark – The General Language Understanding Evaluation (GLUE) benchmark (Wang et al., 2018) has been widely used to evaluate pre-trained language models. In order to assess KinyaBERT performance on such high level language tasks, we used Google Translate API to translate a subset of the GLUE benchmark (MRPC, QNLI, RTE, SST-2, STS-B and WNLI tasks) into Kinyarwanda. CoLA task was left because it is English-specific. MNLI and QQP tasks were also not translated because they were too expensive to translate with Google's commercial API. While machine translation adds more noise to the data, evaluating on this dataset is still relevant because all models compared have to cope with the same noise. To understand this translation noise, we also run user evaluation experiments, whereby 4 volunteers proficient in both English and Kinyarwanda evaluated a random sample of 6000 translated GLUE examples, and assigned a score to each example on a scale from 1 to 4 (See Table 11 in Appendix). These scores help us characterize the noise in the data and contextualize our results with regards to other GLUE evaluations. Results on these GLUE tasks are shown in Table 3.

Named entity recognition (NER) – We use the Kinyarwanda subset of the MasakhaNER dataset (Adelani et al., 2021) for NER task. This is a high quality NER dataset annotated by native speakers for major African languages including Kinyarwanda. The task requires predicting four entity types: Persons (PER), Locations (LOC), Organizations (ORG), and date & time (DATE). Results on this NER task are presented in Table 4.

News Categorization Task (NEWS) – For a document classification experiment, we collected a set of categorized news articles from seven major news websites that regularly publish in Kinyarwanda. The authors had already categorized the articles, therefore no more manual labeling was needed. This dataset is similar to (Niyongabo et al., 2020), but in our case, we limited the number collected articles per category to 3000 in order to have a more balanced label distribution (See Table 10 in the Appendix). The final dataset contains a total of 25.7K articles spanning 12 categories and has been split into training, validation and test sets in the ratios of 70%, 5% and 25% respectively. Results on this NEWS task are presented in Table 5.

3.3 Main results

The main results are presented in Table 3, Table 4, and Table 5. Each result is the average of 10 independent fine-tuning runs. Each average result is shown with the standard deviation of the 10 runs. Except for XLM-R, all other models are pre-trained on the same corpus (See Table 2) for 32K steps using the same hyper-parameters.

On the GLUE task, KinyaBERT$_{ASC}$ achieves 4.3% better average score than the strongest baseline. KinyaBERT$_{ASC}$ also leads to more robust results on multiple tasks. It is also shown that having just a morphological analyzer is not enough: BERT$_{MORPHO}$ still under-performs even though it uses morphological tokenization. Multilingual XLM-R achieves least performance in most cases, possibly because it was not pre-trained on Kinyarwanda text and uses inadequate tokenization.

On the NER task, KinyaBERT$_{ADR}$ achieves best performance, about 3.2% better average F1 score than the strongest baseline. One of the architectural differences between KinyaBERT$_{ADR}$ and KinyaBERT$_{ASC}$ is that KinyaBERT$_{ADR}$ uses 3 POS tag embeddings while KinyaBERT$_{ASC}$ uses 2. Assuming that POS tagging facilitates named entity recognition, this empirical result suggests that increasing the amount of POS tag information in the model, possibly through diversification (i.e. multiple POS tag embedding vectors per word), can lead to better NER performance.

The NEWS categorization task resulted in differing performances between validation and test sets. This may be a result that solving such task does not require high level language modeling but rather depends on spotting few keywords. Previous research on a similar task (Niyongabo et al., 2020) has shown that simple classifiers based on TF-IDF features suffice to achieve best performance.

The morphological analyzer and part of speech tagger used, inherently have some level of noise because they do not always perform with perfect accuracy. While we did not have a simple way of assessing the impact of POS tagger noise in this work, we can logically expect that the lower the noise the better the results could be. Improving the POS tagger and quantitatively evaluating its accuracy is part of future work. Even though our POS tagger uses some heuristic methods and was evaluated mainly through qualitative exploration, we can still see its positive impact on the pre-trained language model.

Additional results, which are added to the appendix, indicate that KinyaBERT fine-tuning has better convergence (See Figure 2 in Appendix for the loss curves). It is also shown that positional attention (Ke et al., 2020) learned by KinyaBERT has more uniform and smoother relative bias while BERT$_{BPE}$ and BERT$_{MORPHO}$ have more noisy relative positional bias (See Figure 3 in Appendix). This is possibly an indication that KinyaBERT allows learning better part-of-speech -relative syntactic regularities.

3.4 Ablation study

We conducted an ablation study to clarify some of the design choices made for KinyaBERT architecture. We make variations along two axes: (i) morphology input and (ii) pre-training task which gave us four variants that we pre-trained for 32K steps and evaluated on the same 8 downstream tasks.

- **AFS→STEM+ASC**: Morphological features are captured by 2 POS tag and 1 affix set vectors. We predict both the stem and affix set. This corresponds to KinyaBERT$_{ASC}$ presented in the main results.

- **POS→STEM+ADR**: Morphological features are carried by 3 POS tag vectors and we predict the stem and affix probability vector. This corresponds to KinyaBERT$_{ADR}$.

- **AVG→STEM+ADR**: Morphological features are captured by 2 POS tag vectors and the average of affix hidden vectors from the morphology encoder. We predict the stem and affix probability vector.

Task:	MRPC	QNLI	RTE	SST-2	STS-B	WNLI	NER	NEWS
Morphology→Prediction			**Validation Set**					
AFS→STEM+ASC	86.6/81.3	**82.3**	**64.3**	**82.4**	**80.0/79.9**	56.2	86.2	88.4
POS→STEM+ADR	**87.1/82.1**	81.6	61.8	81.8	79.6/79.5	54.5	**87.1**	**88.8**
AVG→STEM+ADR	85.5/80.3	81.4	63.0	82.1	79.6/79.5	55.8	86.6	88.3
STEM→STEM	86.4/81.5	80.4	63.4	77.5	79.7/79.5	50.4	86.6	88.0
Morphology→Prediction			**Test Set**					
AFS→STEM+ASC	**84.6**/78.4	**82.2**	58.8	**81.4**	**74.5/73.5**	65.0	76.3	88.0
POS→STEM+ADR	84.4/**78.7**	81.2	58.1	80.9	73.2/72.0	65.1	**77.2**	88.0
AVG→STEM+ADR	84.0/78.2	81.7	59.4	80.7	73.6/72.6	65.0	76.9	88.2
STEM→STEM	84.2/78.6	80.3	59.8	77.5	73.3/72.0	59.6	76.4	**88.4**

Table 6: Ablation results: each result is an average of 10 independent fine-tuning runs. Metrics, dataset sizes and noise statistics are the same as for the main results in Table 3, Table 4 and Table 5.

- **STEM→STEM**: We omit the morphology encoder and train a model with only the stem parts without affixes and only predict the stem.

Ablation results presented in Table 6 indicate that using affix sets for both morphology encoding and prediction gives better results for many GLUE tasks. The under-performance of "STEM→STEM" on high resource tasks (QNLI and SST-2) is an indication that morphological information from affixes is important. However, the utility of this information depends on the task as we see mixed results on other tasks.

4 Related Work

BERT-variant pre-trained language models (PLMs) were initially pre-trained on monolingual high-resource languages. Multilingual PLMs that include both high-resource and low-resource languages have also been introduced Devlin et al. (2019); Conneau et al. (2020); Xue et al. (2020). However, it has been found that these multilingual models are biased towards high-resource languages and use fewer low quality and uncleaned low-resource data (Caswell et al., 2021). The included low-resource languages are also very limited because they are mainly sourced from Wikipedia articles, where languages with few articles like Kinyarwanda are often left behind (Joshi et al., 2020; ∀ et al., 2020).

Joshi et al. (2020) classify the state of NLP for Kinyarwanda as "Scraping-By", meaning it has been mostly excluded from previous NLP research, and require the creation of dedicated resources and models. Kinyarwanda has been studied mostly in descriptive linguistics (Kimenyi, 1976, 1978; KIMENYI, 1978; Kimenyi, 1988; Jerro, 2016). Few recent NLP works on Kinyarwanda include Morphological Analysis (Muhirwe, 2009; Nzeyimana, 2020), Text Classification (Niyongabo et al., 2020), Named Entity Recognition (Rijhwani et al., 2020; Adelani et al., 2021; Sälevä and Lignos, 2021), POS tagging (Garrette and Baldridge, 2013; Garrette et al., 2013; Duong et al., 2014; Fang and Cohn, 2016; Cardenas et al., 2019), and Parsing (Sun et al., 2014; Mielens et al., 2015). There is no prior study on pre-trained language modeling for Kinyarwanda.

There are very few works on PLMs for African languages. To the best of our knowledge there is currently only AfriBERT (Ralethe, 2020) that has been pre-trained on Afrikaans, a language spoken in South Africa. In this paper, we aim to increase the inclusion of African languages in NLP community by introducing a PLM for Kinyarwanda. Differently to the previous works which solely pre-trained unmodified BERT models, we propose an improved BERT architecture for morphologically rich languages.

5 Conclusion

This work demonstrates the effectiveness of explicitly incorporating morphological information in language model pre-training. The proposed two-tier Transformer architecture allows the model to represent morphological compositionality. Experiments conducted on Kinyarwanda, a low resource morphologically rich language, reveal significant performance improvement on several downstream NLP tasks when using the proposed architecture. These findings should motivate more research into morphology-aware language models.

References

David Ifeoluwa Adelani, Jade Abbott, Graham Neubig, Daniel D'souza, Julia Kreutzer, Constantine Lignos, Chester Palen-Michel, Happy Buzaaba, Shruti Rijhwani, Sebastian Ruder, et al. 2021. Masakhaner: Named entity recognition for african languages. *arXiv preprint arXiv:2103.11811*.

Fady Baly, Hazem Hajj, et al. 2020. Arabert: Transformer-based model for arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15.

Kenneth R Beesley and Lauri Karttunen. 2000. Finite-state non-concatenative morphotactics. *arXiv preprint cs/0006044*.

Kenneth R Beesley and Lauri Karttunen. 2003. Finite-state morphology: Xerox tools and techniques. *CSLI, Stanford*.

Emily M Bender. 2019. The# benderrule: On naming the languages we study and why it matters. *The Gradient*, 14.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *The journal of machine learning research*, 3:1137–1155.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

José Canete, Gabriel Chaperon, Rodrigo Fuentes, and Jorge Pérez. 2020. Spanish pre-trained bert model and evaluation data. *PML4DC at ICLR*, 2020.

Ronald Cardenas, Ying Lin, Heng Ji, and Jonathan May. 2019. A grounded unsupervised universal part-of-speech tagger for low-resource languages. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2428–2439.

Isaac Caswell, Julia Kreutzer, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, et al. 2021. Quality at a glance: An audit of web-crawled multilingual datasets. *arXiv preprint arXiv:2103.12028*.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.

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.

Pieter Delobelle, Thomas Winters, and Bettina Berendt. 2020. RobBERT: a Dutch RoBERTa-based Language Model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3255–3265, Online. Association for Computational Linguistics.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Long Duong, Trevor Cohn, Karin Verspoor, Steven Bird, and Paul Cook. 2014. What can we get from 1000 tokens? a case study of multilingual pos tagging for resource-poor languages. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 886–897.

Meng Fang and Trevor Cohn. 2016. Learning when to trust distant supervision: An application to low-resource pos tagging using cross-lingual projection. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 178–186.

∀, Wilhelmina Nekoto, Vukosi Marivate, Tshinondiwa Matsila, Timi Fasubaa, Taiwo Fagbohungbe, Solomon Oluwole Akinola, Shamsuddeen Muhammad, Salomon Kabongo Kabenamualu, Salomey Osei, Freshia Sackey, et al. 2020. Participatory research for low-resourced machine translation: A case study in african languages. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 2144–2160.

G David Forney. 1973. The viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278.

Dan Garrette and Jason Baldridge. 2013. Learning a part-of-speech tagger from two hours of annotation. In *Proceedings of the 2013 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 138–147.

Dan Garrette, Jason Mielens, and Jason Baldridge. 2013. Real-world semi-supervised learning of pos-taggers for low-resource languages. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 583–592.

Kyle Jerro. 2016. The locative applicative and the semantics of verb class in kinyarwanda. *Diversity in African languages*, page 289.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the nlp world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293.

Guolin Ke, Di He, and Tie-Yan Liu. 2020. Rethinking the positional encoding in language pre-training. *arXiv preprint arXiv:2006.15595*.

A KIMENYI. 1978. A relational grammar of kinyarwanda. *University of California, Publications in Linguistics Berkeley, Cal*, 91:1–248.

Alexandre Kimenyi. 1976. Subjectivization rules in kinyarwanda. In *Annual Meeting of the Berkeley Linguistics Society*, volume 2, pages 258–268.

Alexandre Kimenyi. 1978. Aspects of naming in kinyarwanda. *Anthropological linguistics*, 20(6):258–271.

Alexandre Kimenyi. 1988. Passiveness in kinyarwanda. In *Passive and Voice*, page 355. John Benjamins.

Kimmo Koskenniemi. 1983. Two-level model for morphological analysis. In *IJCAI*, volume 83, pages 683–685.

Fajri Koto, Afshin Rahimi, Jey Han Lau, and Timothy Baldwin. 2020. Indolem and indobert: A benchmark dataset and pre-trained language model for indonesian nlp. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 757–770.

John Koutsikakis, Ilias Chalkidis, Prodromos Malakasiotis, and Ion Androutsopoulos. 2020. Greek-bert: The greeks visiting sesame street. In *11th Hellenic Conference on Artificial Intelligence*, pages 110–117.

Y Kuratov and M Arkhipov. 2019. Adaptation of deep bidirectional multilingual transformers for russian language. In *Komp'juternaja Lingvistika i Intellektual'nye Tehnologii*, pages 333–339.

Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoit Crabbe, Laurent Besacier, and Didier Schwab. 2020. Flaubert: Unsupervised language model pre-training for french. In *LREC*.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: a tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, Online. Association for Computational Linguistics.

Mihai Masala, Stefan Ruseti, and Mihai Dascalu. 2020. Robert–a romanian bert model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6626–6637.

Jason Mielens, Liang Sun, and Jason Baldridge. 2015. Parse imputation for dependency annotations. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1385–1394.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. *arXiv preprint arXiv:1310.4546*.

Mahdi Mohseni and Amirhossein Tebbifakhr. 2019. MorphoBERT: a Persian NER system with BERT and morphological analysis. In *Proceedings of The First International Workshop on NLP Solutions for Under Resourced Languages (NSURL 2019) co-located with ICNLSP 2019 - Short Papers*, pages 23–30, Trento, Italy. Association for Computational Linguistics.

Jackson Muhirwe. 2009. Morphological analysis of tone marked kinyarwanda text. In *International Workshop on Finite-State Methods and Natural Language Processing*, pages 48–55. Springer.

Dat Quoc Nguyen and Anh Tuan Nguyen. 2020. PhoBERT: Pre-trained language models for Vietnamese. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1037–1042, Online. Association for Computational Linguistics.

Rubungo Andre Niyongabo, Qu Hong, Julia Kreutzer, and Li Huang. 2020. Kinnews and kirnews: Benchmarking cross-lingual text classification for kinyarwanda and kirundi. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5507–5521.

Antoine Nzeyimana. 2020. Morphological disambiguation from stemming data. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4649–4660, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Sello Ralethe. 2020. Adaptation of deep bidirectional transformers for afrikaans language. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 2475–2478.

Shruti Rijhwani, Shuyan Zhou, Graham Neubig, and Jaime G Carbonell. 2020. Soft gazetteers for low-resource named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8118–8123.

Piotr Rybak, Robert Mroczkowski, Janusz Tracz, and Ireneusz Gawlik. 2020. Klej: Comprehensive benchmark for polish language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1191–1201.

Jonne Sälevä and Constantine Lignos. 2021. Mining wikidata for name resources for african languages. *arXiv preprint arXiv:2104.00558*.

Raphael Scheible, Fabian Thomczyk, Patric Tippmann, Victor Jaravine, and Martin Boeker. 2020. Gottbert: a pure german language model. *arXiv preprint arXiv:2012.02110*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Fábio Souza, Rodrigo Nogueira, and Roberto Lotufo. 2020. Bertimbau: Pretrained bert models for brazilian portuguese. In *Brazilian Conference on Intelligent Systems*, pages 403–417. Springer.

Liang Sun, Jason Mielens, and Jason Baldridge. 2014. Parsing low-resource languages using gibbs sampling for pcfgs with latent annotations. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 290–300.

Yoshimasa Tsuruoka and Jun'ichi Tsujii. 2005. Bidirectional inference with the easiest-first strategy for tagging sequence data. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 467–474.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. 2019. Multilingual is not enough: Bert for finnish. *arXiv preprint arXiv:1912.07076*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. 2020. Big bird: Transformers for longer sequences. *arXiv preprint arXiv:2007.14062*.

Appendix A Data Tables, Hyper-parameters & Additional results

Module	Values
Morphology Encoder:	
Number of Layers	4
Attention heads	4
Hidden Size	128
Attention head size	32
FFN inner hidden size	512
Morphological embedding size	128
Sentence/Document Encoder:	
Number of Layers	12
Attention heads	12
Hidden Size	768
Attention head size	64
FFN inner hidden size	3072
Stem embedding size	256

Table 7: KinyaBERT Architectural dimensions.

Model	Size
XLM-R:	
Sentence-Piece tokens	250K
BERT$_{BPE}$**:**	
BPE Tokens	43K
BERT$_{MORPHO}$**:**	
Morphemes & BPE Tokens	51K
KinyaBERT$_{ADR}$**:**	
Stems & BPE Tokens	34K
Affixes	0.3K
POS Tags	0.2K
KinyaBERT$_{ASC}$**:**	
Stems & BPE Tokens	34K
Affix sets	34K
Affixes	0.3K
POS Tags	0.2K

Table 8: Vocabulary sizes for embedding layers.

Hyper-parameter	Values
Dropout	0.1
Attention Dropout	0.1
Warmup Steps	2K
Max Steps	200K
Weight Decay	0.01
Learning Rate Decay	Linear
Peak Learning Rate	4e-4
Batch Size	2560
Optimizer	LAMB
Adam ϵ	1e-6
Adam β_1	0.90
Adam β_2	0.98
Gradient Clipping	0

Table 9: Pre-training hyper-parameters

Category	#Articles
entertainment	3000
sports	3000
security	3000
economy	3000
health	3000
politics	3000
religion	2020
development	1813
technology	1105
culture	994
relationships	940
people	852
Total	25724

Table 10: NEWS categorization dataset label distribution.

Score	Translation quality
1	Invalid or meaningless translation
2	Invalid but not totally wrong
3	Almost valid, but not totally correct
4	Valid and correct translation

Table 11: Machine-translated GLUE benchmark scoring prompt levels.

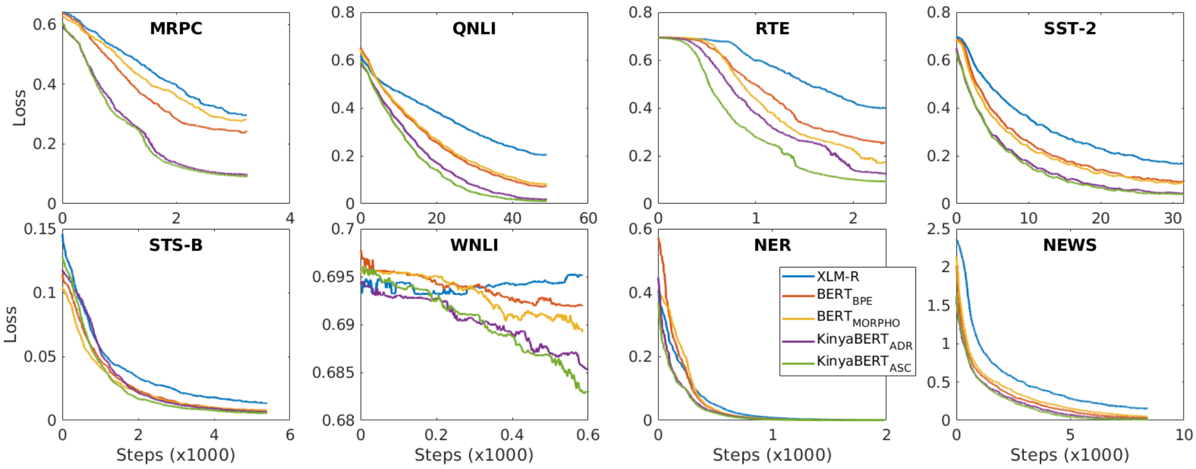


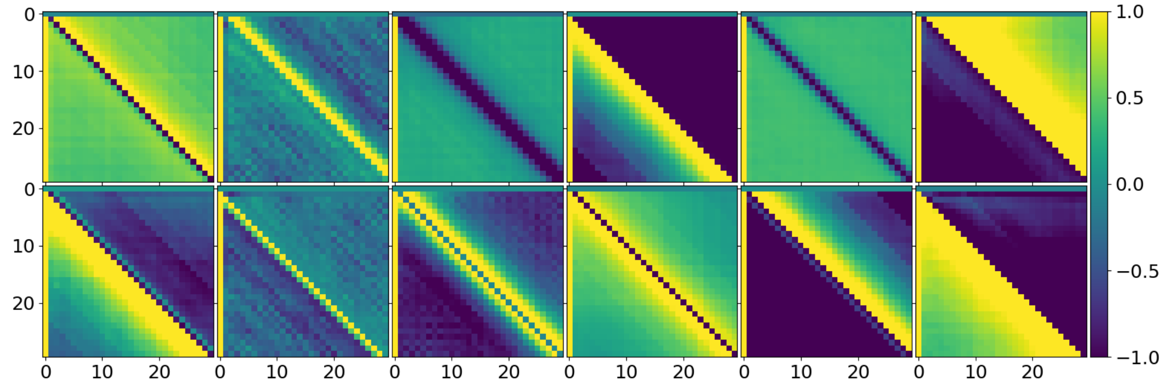
Figure 2: Comparison of fine-tuning loss curves between KinyaBERT and baselines on the evaluation tasks. KinyaBERT$_{ASC}$ achieves the best convergence in most cases, indicating better effectiveness of its model architecture and pre-training objective.

Hyperparameter	**MRPC**	**QNLI**	**RTE**	**SST-2**	**STS-B**	**WNLI**	**NER**	**NEWS**
Peak Learning Rate	1e-5	1e-5	2e-5	1e-5	2e-5	1e-5	5e-5	1e-5
Batch Size	16	32	16	32	16	16	32	32
Learning Rate Decay	Linear	Linear	Linear	Linear	Linear	Linear	Linear	Linear
Weight Decay	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
Max Epochs	15	15	15	15	15	15	30	15
Warmup Steps proportion	6%	6%	6%	6%	6%	6%	6%	6%
Optimizer	AdamW	AdamW	AdamW	AdamW	AdamW	AdamW	AdamW	AdamW

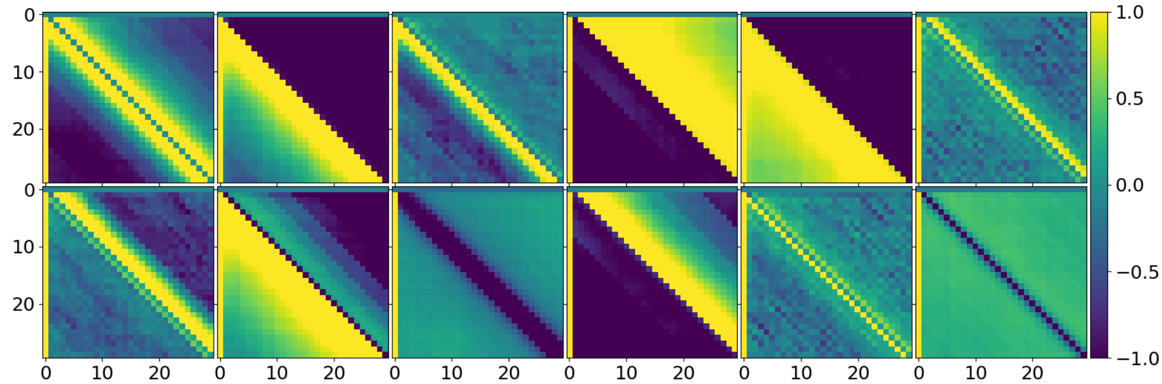
Table 12: Downstream task fine-tuning hyper-parameters.

Paper	**Language**	**Loss Function**	**Positional Embedding**	**Input Representation**
Mohseni and Tebbifakhr (2019)	Persian	MLM+NSP	Absolute	Morphemes
Kuratov and Arkhipov (2019)	Russian	MLM+NSP	Absolute	BPE
Masala et al. (2020)	Romanian	MLM+NSP	Absolute	BPE
Baly et al. (2020)	Arabic	WWM+NSP	Absolute	BPE
Koto et al. (2020)	Indonesian	MLM+NSP	Absolute	BPE
Chan et al. (2020)	German	WWM	Absolute	BPE
Delobelle et al. (2020)	Dutch	MLM	Absolute	BPE
Nguyen and Tuan Nguyen (2020)	Vietnamese	MLM	Absolute	BPE
Canete et al. (2020)	Spanish	WWM	Absolute	BPE
Rybak et al. (2020)	Polish	MLM	Absolute	BPE
Martin et al. (2020)	French	MLM	Absolute	BPE
Le et al. (2020)	French	MLM	Absolute	BPE
Koutsikakis et al. (2020)	Greek	MLM+NSP	Absolute	BPE
Souza et al. (2020)	Portuguese	MLM	Absolute	BPE
Ralethe (2020)	Afrikaans	MLM+NSP	Absolute	BPE
This work	Kinyarwanda	MLM: STEM+AFFIXES	TUPE-R	Morphemes+BPE

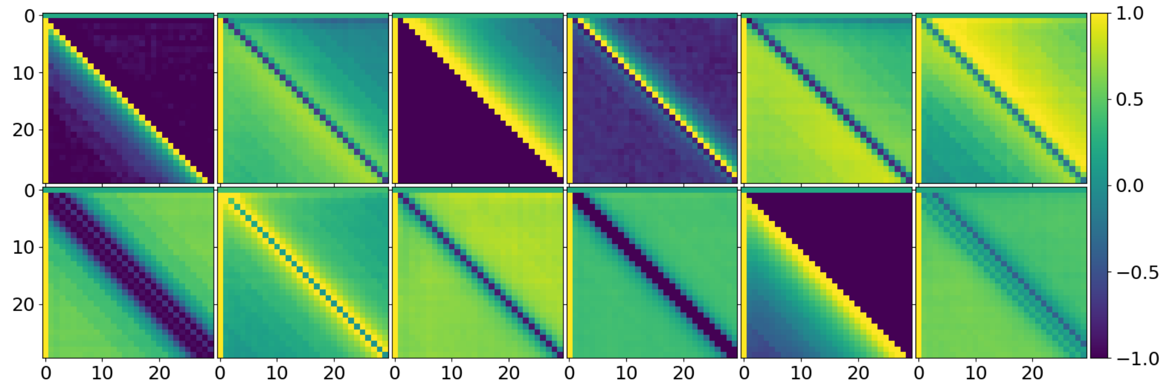
Table 13: The comparison of KinyaBERT with other monolingual BERT-variant PLMs. We only compare with the previous works that have been published in either journals or conferences, since reviewing all works is out of the scope of this paper. NSP: Next Sentence Prediction, WWM: Whole Word Masked.



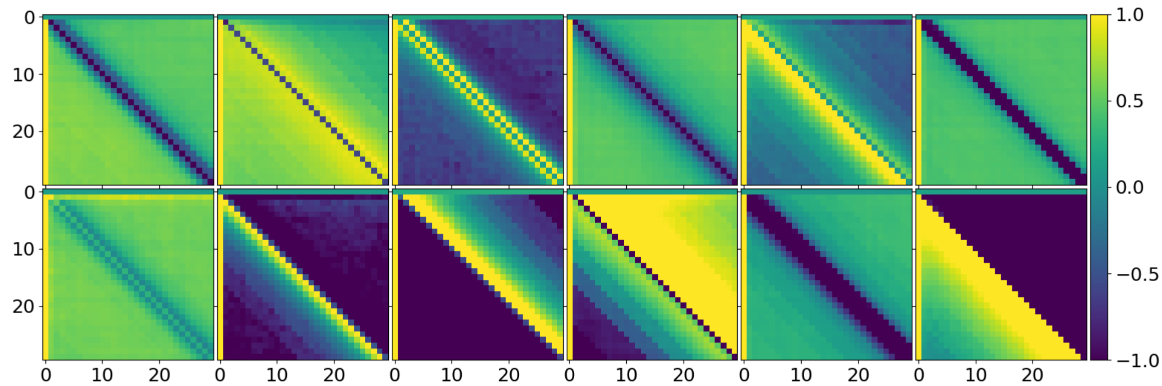
BERT$_{BPE}$



BERT$_{MORPHO}$



KinyaBERT$_{ADR}$



KinyaBERT$_{ASC}$

Figure 3: Visualization of the positional attention bias (normalized) of the 12 attention heads. Each (i, j) attention bias (Ke et al., 2020) indicates the positional correlations between the i^{th} and j^{th} words/tokens in a sentence.

Tag	Description	Example
V#000	Infinitive Verb	kuvuga (ku-vug-a) *'to say'*
V#001	Verbal with Nominal Augment	uwavuze (u-a-vug-ye) *'the one who said'*
N#011	Noun without augment	mwana (mu-ana) *'the child'*
N#012	Noun with augment	umwana (u-mu-ana) *'child'*
DE#017	Demonstrative with 'nka' prefix	nkawe (nka-u-e) *'like you'*
DE#020	Demonstrative with 1st or 2nd person	njyewe (njy-ewe) *'me'*
PO#022	Possesive without augment, with owner marker	wa (u-a) *'of'*
PO#025	Possesive with augment, with owner marker	uwacu (u-a-cu) *'ours'*
QA#026	Qualitative adjective	mwiza (mu-iza) *'good/beautiful'*
NU#030	Numeral	babiri (ba-biri) *'two (persons)'*
OT#031	Quantifier	bose (ba-ose) *'all'*
NP#035	Proper noun	Mugenzi
DI#036	Digits	1000
SP#054	Spatial	haruguru *'up'*
PR#057	Preposition	ku *'on'*
CJ#071	Conjunction	ko *'that...'*
PT#085	Punctuation mark: comma	,

Table 14: Examples of part-of-speech tags used in KinyaBERT

Original (#0)	Weapons of Mass Destruction Found in Iraq.
Translated	*Intwaro yo Kurimbura Misa Yabonetse muri Iraki.*
Translated meaning	Weapon for destroying a mass(prayer) has been found in Iraq.
Original (#299)	Kerry hit Bush hard on his conduct on the war in Iraq.
Translated	*Kerry yakubise Bush ku myitwarire ye ku ntambara yo muri Iraki.*
Translated meaning	Kerry punched Bush about his conduct on the war in Iraq.

Table 15: Examples of noisy translated sentences from the RTE training set