BantuBERTa: Using Language Family Grouping in Multilingual Language Modeling for Bantu Languages

Jesse **Parvess**

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Denkleiers • Leading Minds • Dikgopolo tša Dihlalefi

Faculty of Engineering, Built Environment & IT,
Department of Computer Science, University of Pretoria, Pretoria.

A mini-Dissertation submitted to the Faculty of Science in fulfilment of the requirements for the Master degree in Big Data Science.

Supervised by
1st Supervisor -Dr Vukosi **Marivate**
2nd Supervisor - Dr Verrah **Akinyi**

February 15, 2023

Declaration

I, Jesse Parvess, hereby declare the content of this dissertation to be my own work unless otherwise explicitly referenced. This dissertation is submitted in partial satisfaction of the requirements for Master degree in Big Data Science at the University of Pretoria, Pretoria. This work has not been submitted to any other university, nor for any other degree.

Signed: Jesse Parvess

Date: 2023-02-15

Abstract

It was researched whether a multilingual Bantu pretraining corpus could be created from freely available data. Here, to create the dataset, Bantu text extracted from datasets that are freely available online (mainly from Huggingface) were used. The resulting multilingual language model (BantuBERTa) from this pretraining data proved to be predictive across multiple Bantu languages on a higher-order NLP task (NER) and in a simpler NLP task (classification). This proves that this dataset can be used for Bantu multilingual pretraining and transfer to multiple Bantu languages. Additionally, it was researched whether using this Bantu dataset could benefit transfer learning in downstream NLP tasks. BantuBERTa under-performed with respect to other models (XlM-R, mBERT, and AfriBERTa) bench-marked on MasakhaNER's Bantu language tests (Swahili, Luganda, and Kinyarwanda). Additionally, it produced state of the art results for the Bantu language benchmarks (Zulu, and Lingala) in the African News Topic Classification dataset. It was surmised that the pretraining dataset size (which was 30% smaller than AfriBERTa's) and dataset quality were the main cause for the poor performance in the NER test. We believe this is a case-specific failure due to poor data quality resulting from a pretraining dataset consisting mainly of web-scraped pages. Here, the resulting dataset consisted mainly of MC4 and CC100 Bantu text. However, on lower-order NLP tasks, like classification, pretraining on languages solely within the language family seemed to benefit transfer to other similar languages within the family. This potentially opens a method for effectively including low-resourced languages in low-level NLP tasks.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter the current paradigm of multilingual language modeling in natural language processing (NLP) is introduced along with the main challenges faced by low-resourced languages in this space. Furthermore, the benefit of using related languages in the pretraining corpus to improve cross-lingual transfer between low-resourced languages is introduced.

In section 1.1 the main multilingual language models used in current state of the art NLP applications are introduced. In section 1.2 the main issues faced when using these models in low-resourced language settings are explored. Additionally, in section 1.3 the Bantu language family is qualified as a low-resourced language family that can benefit from using only related languages in the multilingual pretraining corpus for downstream cross-lingual transfer. Lastly, in section 1.4 BantuBERTa is introduced as a potentially optimised approach for multilingual modeling for low-resourced languages.

1.1 Low-resource Languages in the Current Multilingual Language Modeling Paradigm

Current state of the art multilingual language models use transformers [Vaswani et al., 2017] pretrained on many languages to produce multilingual contextual word embeddings. Here, multilingual pretrained transformers like mBERT [Devlin et al., 2018] and XLM-R [Conneau et al., 2019] have produced consistently competitive results when transferred to different languages on downstream NLP applications.

These models were pretrained using a multilingual corpus of over a 100 languages scraped from the web or Wikipedia. The assumption underpinning this approach is that the multilingual language representation learned during pretraining, from many different languages, would result in a universal language embedding [Pires et al., 2019]. This robust universal language representation (embedding) would then be able to be transferred to a downstream NLP application via fine-tuning the model for a specific language and task. In this case the multilingual transformer model would not need to be pretrained on many examples (or at all) from the specific target language in question as the universal language representation would be sufficient K et al. [2019].

This is particularly useful when there is not much data available for language modeling in a particular language, and the benefits of a transformer are required. One can clearly see that this poses great potential for low-resourced languages Muller et al. [2021]. Here, these languages piggy-back off the universal language representation learned from other languages. However, this approach has been shown to be biased towards higher-resourced languages while offering a non-optimal solution for low-resourced languages.

Joshi et al. [2020] classified the languages of the world with respect to the available resources required to create natural language processing applications. It was found that many under-resourced languages are African languages. In fact the only African language included in mBERT and XLM-R was Swahili, and it comprised a very small portion of its pretraining data Ogueji et al. [2021]. Therefore, the currently used multilingual language models available are ill-equiped in handling low-resourced African languages. The specific reasons for this are discussed in the following sections.

1.2 The Main Issues in Multilingual Language Modeling

Most of the languages used in the pretraining data of mBERT and XLM-R are high-resourced [Ogueji et al., 2021]. This would not necessarily present an issue if the entire transformer model's architecture consisted solely of universal language parameters. In this case the composition of the multilingual pretraining corpus would provide little issue as the multilingual language model would be language agnostic and downstream task transfer could occur between any language with out any bias.

However, it has been shown that these multilingual language models have language specific parameters in their architecture [Wang et al., 2020]. Here, to reduce loss during pretraining, the model devotes certain parameters to specific languages. Naturally, if the pretraining corpus consists mainly of higher-resourced languages, these language specific parameters are devoted to these languages. This process is compounded by the "curse of multilinguality" . Here the model's performance degrades as more languages are added to the pretraining corpus at a fixed model size [Conneau et al., 2019].

The addition of more languages in the pretraining corpus causes competition for these language specific parameters. Evidently, topographically similar languages optimally share the same language-specific parameters when the multilingual model size is constrained. Here, high-resourced-etymologically-similar languages (with many examples in the pretraining corpus) bias the multilingual model towards these languages [Conneau et al., 2019]. Most of the model's language specific parameters cater for and are shared by these high-resourced languages. Thus, languages that are underrepresented in the pretraining corpus, and are topographically dissimilar to the high-resourced languages used, are not optimally catered for in language specific parameters.

This would present little issue if high-resourced languages were topographically similar to low-resourced languages. The learned language specific component in pretraining could contribute to some form of transfer in this case. However, this is not the case. Low-resourced languages are often structurally very different [Kambarami et al., 2021]. For instance, the high-resourced Germanic languages used in mBERT and XLM-R do not use agglutination in their morphology like low-resourced Bantu languages [Pretorius et al., 2009]. Therefore, to optimise multilingual models for transfer to low-resource languages, the pretraining corpus should contain languages that are topographically aligned (morphologically similar) to the target language(s) in question.

Furthermore, a low-resourced language family that contains languages that are all topographically similar would benefit from the above approach. Here, a topographically similar low-resourced language family can be used solely in the multilingual pretraining corpus to create a language model. The model will be biased towards low-resourced languages that have the same topography and transferred effectively to other low-resourced languages within the family.

1.3 The Bantu Language Family

A prime candidate language family that can benefit from the above approach is the Bantu language family.

The Bantu languages are considered a universally low-resourced language group [Joshi et al., 2020]. Furthermore, a significant population of people (approximately 500 million people) speak languages classified within the family [Pauw et al., 2012]. This implies that downstream NLP applications, developed from a single effective multilingual Bantu language model, that can transfer to multiple Bantu languages, has a wide reach. Additionally, Bantu languages are all topographically similar [Pretorius et al., 2009] as they use agglutination with morphemes.

Therefore, by combining the available Bantu language text into a single multilingual pretraining corpus, a multilingual Bantu language model can be pretrained. It will potentially have reduced negative interference from language specific parameters as all languages in the pretraining corpus are structurally similar. This will potentially provide a more optimised means for reaping the benefits from transformers for low-resourced languages.

1.4 BantuBERTa

We introduce BantuBERTa: A multilingual language model that uses solely low-resourced topographically similar languages (Bantu language text) in pretraining to optimise transfer to other similar low-resourced languages. A similar implementation was performed by Ogueji et al. [2021]. They presented AfriBERTa, which was pretrained solely on 11 low-resourced African languages. However, unlike our approach, these languages came from different language families and where topographically different. The authors compared AfriBERTa to XLM-R and mBERT on named entity recognition (NER) tasks provided by MasakhaNER [Adelani et al., 2021].

Similarly, BantuBERTa is compared to AfriBERTa, XLM-R and mBERT on MasakhaNER tasks presented for Bantu languages (Swahili, Luganda, and Kinyarwanda). In addition to the NER NLP task, BantuBERTa was fine-tuned on the Bantu languages presented in the African News Topic classification dataset (Zulu and Lingala) created by Alabi et al. [2022]. Here, Alabi et al. [2022] produced the best benchmarks for these languages using Multilingual Adaptive Fine-Tuning (MAFT) of XLM-R-base.

By comparing BantuBERTa to AfriBERTa, XLM-R, and mBERT in the difficult natural language processing (NLP) task of NER and the simple NLP task of classification, across multiple Bantu languages (Swahili, Luganda, Kinyarwanda, Zulu and Lingala), we look to answer the following research questions:

1. Can a comparatively small multilingual Bantu language pretraining corpus be created from the freely available online multilingual datasets?
2. Can pretraining on this multilingual Bantu corpus benefit transfer learning to Bantu languages in downstream NLP tasks?

Chapter 2

Literature Review

In section 2.1 the main pretrained language models used in multilingual modelling are discussed (mBERT and XLM-R) along with their general characteristics. In section 2.2 the theory of transferring a multilingual language model to multiple languages are investigated. Furthermore, in section 2.3 the current issues with transferring these models to low-resource languages are expanded. In section 2.4 Bantu languages are investigated along with their morphology and the considerations required for multilingual language modelling.

2.1 Multilingual Language Models: mBERT and XLM-R

Multilingual Bidirectional Encoder Representations from Transformers (mBERT) was introduced by Devlin et al. [2018]. This language model was pretrained on 104 languages from Wikipedia, using masked language modelling (MLM) and next sentence prediction (NSP) learning objectives.

In masked language modelling, a certain percentage of tokens are masked with [MASK] in the example sentence. In this case mBERT had 15% of all tokens in a sentence masked. The model's objective is to predict the missing word based on the sequences of words to the left and right of the masked entity (the token's context). For next sentence prediction the model is shown two sentences, the task is to predict if the second sentence follows the first, based on the context of both sentences in sequence. However, this has been shown by Zhuang et al. [2021] to negatively impact the the language representation learned. Subsequently many models have have been trained without it.

Language models achieve these objectives via an attention mechanism [Vaswani et al., 2017]. The attention mechanism is widely used as an interpretability mechanism, where the "attention weight has a clear meaning: how much a particular word will be weighted when computing the next representation for the current word" [Rogers et al., 2020].

XLM-R was introduced by Conneau et al. [2019]. This model was trained with the same training objectives as roBERTa [Zhuang et al., 2021] with MLM and no NSP. Furthermore, it was trained on 100 languages from the CommonCrawl corpus. In total XLM-R was trained on 2395 GB of text, which is significantly more than the 100 GB used by mBERT.

2.2 Multilingual Model Transfer in Zero-shot Learning

To illustrate the main paradigms within transferring to other languages using multilingual language models, zero-shot transfer is discussed. Zero-shot transfer occurs when a multilingual language model is shown a language it has never been pretrained on [Lin et al., 2021]. Here, the new language is tokenized using the learned tokenizer from the model's original pretraining corpus. The model is then simply fine-tuned (or transferred) on a downstream NLP task for the unseen language and assessed.

Pires et al. [2019] showed that mBERT is effective at zero-shot transfer to new unseen languages. They found that transfer is possible even with languages of different scripts to the language scripts used in the model's pretraining corpus. Bender et al. [2021] stated that when comparing mBERT to monolingual BERT in zero-shot transfer, mBERT was found to be more effective due to linguistic knowledge acquired from other languages during pretraining. This indicates, that pretraining with multiple languages allows for better zero-shot transfer.

Pires et al. [2019] found that even with zero lexical overlap between the pretrained languages used in mBERT and the target language, zero-shot transfer to unseen languages are possible. Pires et al. [2019] believed this indicated that mBERT has a universal multilingual language representation as there are no reference words that the model could use contextually when tested on the unseen target language. Therefore, the model had to rely solely on its learned generic language representation to transfer learnings to the new language.

Language agnostic representations have even been observed in monolingual BERT models. Here, zero-shot transfer from the monolingual pretrained language to an unseen different target language was observed Gogoulou et al. [2021]. This suggests that transformers can learn universal language agnostic representations, even in monolingual settings, however using multiple languages improves this representation Bender et al. [2021]. Yet, we will show later that indefinitely adding languages to the pretraining corpus subsequently hurts transfer.

It is thought that the masked learning objective (MLM) enables this behaviour, as the model is tasked with using representations of the words surrounding the masked entity to predict its identity. MLM, therefore, enables contextual language representation that is learned from the pretraining corpus [Devlin et al., 2018]. When multiple languages are present in the pretraining corpus, MLM enables a generic language representation that is able to generalise across the multiple languages present. Furthermore, Pires et al. [2019] found that this learned language agnostic parameter representation mainly resides within the model's upper layers.

Pires et al. [2019] also found that performance in zero-shot transfer improves when the unseen target language is topographically similar to some of the languages used in pretraining. The authors concluded that overlap in sub-word vocabulary between pretrained languages plays a pertinent role in acquiring multilinguality. Conneau et al. [2020] supported this finding by stating that source languages for pretraining should be chosen such that they have more word-pieces with the target language used in zero-shot applications.

Lauscher et al. [2020] indicates that zero-shot performance (either from a monolingual or multilingual pretrained base) is mainly dependent on language topological similarity (word ordering, morphology etc). This claim validates Pires et al. [2019] statements, that sub-word overlap mainly enables multilingual representations. This is because topographically similar languages may naturally produce similar word-pieces during tokenization. This is because the language similarity may have be a function of a common language ancestor that relates these languages into a single family. Therefore, the subsequent constituent parts of the two languages are similar.

Lauscher et al. [2020] varied the language similarity and the size of the pretraining corpus in pretraining. These variables were then tested in zero-shot cross lingual transfer. They found that language topographical similarity plays an important role for low-level tasks (like classification) and high-level tasks as well (like natural language inference: NLI) when transferring to unseen languages. For high-level tasks Lauscher et al. [2020] found that the pretraining corpus size was deemed important because learning highly specialised functions (like those used in named entity recognition) in downstream tasks require effective language representations. Effective language representations require many pretraining examples.

However, large pretraining corpus sizes rule out low-resource languages for many high-level tasks. Thus, these low-resourced languages only can use language similarity in the pretraining corpus to access low-level NLP tasks performed by multilingual models. Therefore, to allow low-resource languages access to the transformer language model paradigm, the model must be pretrained on languages that are topographically similar to the low-resourced language being transferred to.

In summary, the studies on zero-shot transfer from a pretrained multilingual language models to an unseen target language, indicate that transfer is enabled by a learned universal language space. However, optimising transfer occurs when the pretrained languages are topographically similar to the target language. Since low-resourced languages (like Bantu languages) do not have access to large corpora, enhancing performance can only occur by aligning the pretraining base of languages to be all topographically similar to the target language. If there are a significant number of low-resource target languages that are all topologically similar, these can all benefit simultaneously from a single model that uses languages in the pretraining corpus that are all topologically aligned.

2.3 Multilingual Modeling Issues for Low-resourced Languages

Ogueji et al. [2021] states that current multilingual language modelling approaches have been shown to work well on many downstream natural language processing (NLP) tasks. However, to do so requires high data volumes to create generic language representations [Conneau et al., 2020]. This has thus ruled out many low resourced languages from being represented within the model's parameters as they constitute a small proportion of the pretraining corpus. Upon reviewing mBERT and XLM-R, Ogueji et al. [2021] found that many of the 100+ languages used in pretraining were mainly highly resourced and topographically similar.

The effects of interference from multiple languages when using structurally different languages in multilingual pretraining was investigated by Wang et al. [2020]. Originally the objective of pretraining on a large set of different languages was for the proposed positive learning transfer to low resource languages through a better generic language representation [Conneau et al., 2019]. Here the benefits conferred from learning universal language structures from multiple different languages was thought to allow for a better generic language representation as more languages are added to the pretraining corpus. This benefit was thought to be acquired from the multilingual model's language agnostic space (as discussed above). Wang et al. [2020] found that model parameters learned in this setting do indicate a language-universal structure. However, it was found that language-specific parameters do exist within the model's architecture. They hypothesize that the language-specific component causes negative interference when transferring to a target language that is vastly different (topographically) to the languages used in pretraining. Here, parameters learned to cater for high-resourced languages negatively impact performance when transferring to low-resourced languages that are structurally different.

Chi et al. [2020] thought it would have been reasonable for the mBERT to learn "a private model for each language" thus avoiding destructive interference. Here, parts of the model's architecture are devoted solely to a particular language. However, due to the model's size constraints, parameter sharing between languages is unavoidable. Furthermore, Conneau et al. [2019] showed that adding more languages in pretraining negatively impacts the model's performance after a certain point. This was dubbed the "curse of multilinguality". Here, at fixed model capacity, a conflict arises between languages for parameters needed to learn language specific entities.Wang et al. [2020] support this hypothesis by showing that model capacity (depth and total number of parameters) is crucial for cross-lingual performance. Here, the curse of multilinguality is mitigated by increasing model capacity [Conneau et al., 2019].

Conneau et al. [2019] investigated the curse of multilinguality in XLM-R. They pretrained multiple models of different sizes on varying numbers of languages. Here, as the number of languages used in pretraining increased, they increased the transformer's width. They found that the added capacity allowed for larger models (trained on more languages) to perform the same as smaller representations that were pretrained on fewer languages (with less negative language interference). Thus, model capacity represents a limiting factor in mitigating negative interference from disparate pretraining languages.

It should be noted that increasing the model capacity to cater for cross-lingual transfer to low resource languages is not a viable solution. Here a larger model may allow for the inclusion of more disparate low-resourced languages in the current multilingual models. However, training larger models require immense amounts of computation resources which have a significant environmental impact [Strubell et al., 2019]. Therefore, this is not an option.

In summary, the above discussion indicates that the current paradigm of pretraining on many disparate languages in a multilingual setting creates model capacitance issues. Here languages compete for parameters within the model. The model becomes biased towards over-represented high-resourced etymologically similar languages within the pretraining corpus. These models cannot be indefinitely increased in size to cater for more languages. This essentially leaves under-represented low-resourced languages, that are topologically dissimilar, to be ineffectively applied to current state-of-the-art NLP technology. Therefore, there is a need to pretrain a multilingual model on languages that are topographically similar to multiple low-resourced languages. This topographical alignment in the pretraining corpus will potentially reduce the size constraints required by the model and allow for more effective transfer to many low-resource languages. A language family suited for this application (by the sheer number of low-resource languages represented) is the Bantu language family.

2.4 Modeling Considerations for Bantu Languages

2.4.1 Bantu Language Grouping

Bantu languages fall part of the NigerCongo grouping of African languages, with the other African language groupings being Nilo-Saharan, Afroasiatic, and Khoisan.

The expansion and evolution of Bantu languages because of migration from central Africa (due to agricultural innovation) has been extensively studied [Whiteley et al., 2019]. It has been postulated that Bantu people migrated from an area around Nigeria-Cameroon. Here ancestral Bantu languages spread into central, west-central, eastern and southern Africa [Whiteley et al., 2019]. During migration, dialects evolved to separate languages in distinct geographic regions. However, there is disagreement on the routes used by these populations and how the various languages arose [Currie et al., 2013].

Currie et al. [2013] studied the phylo-geography of Bantu languages as they arose. Their results suggest that people moved south through the Congo rain forest. Upon emerging, migration occurred in two branches. One branch moved south-west towards Angola and the other moved to towards the Great Lakes to form the Eastern Bantu languages like Swahili. From here migration occurred southward to Southern African to create southern Bantu language groups like Nguni and Sotho. Their two-branch hypothesis roughly matches Malcom Gurthrie's Bantu language geographic groupings from A to S [Guthrie, 2017]. Here, the groups J,M and S roughly separate the Bantu languages into west and east groupings (two branches).

This implies that Gutherie's geographic Bantu language groupings can be used as a proxy for Bantu language similarity. Successively similar languages arose as people migrated, therefore Gutherie groupings which are close together (within each respective branch) should produce more similar languages than groups that are geographically further apart. Therefore, Bantu languages spoken by peoples that are geographically close together, can be grouped together as a proxy for topographical similarity.

Figure 2.1 shows Gutherie's Bantu language family groupings. Languages in groups J, F, M mark the boundary between Western and Eastern Bantu languages. Furthermore, group S contains the Southern Bantu Languages, characterised by Nguni and Sotho Bantu languages.

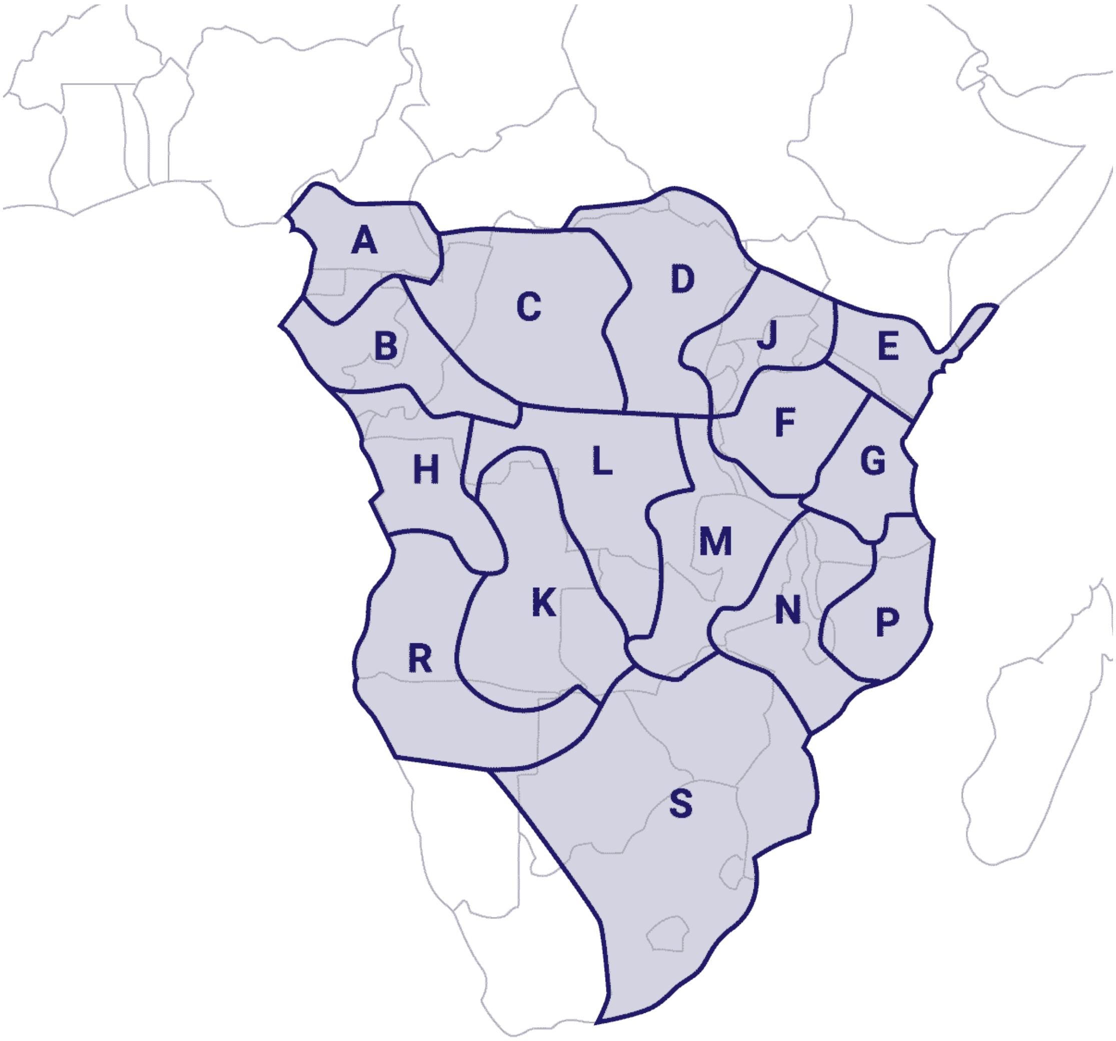

Figure 2.1: Gutherie's Bantu Language Geographical Groupings

2.4.2 Bantu Language Morphology

All Bantu languages exhibit similar structural similarity [Pretorius et al., 2009]. Here, all Bantu languages are agglutinative. Agglutinative languages use two or more morphemes as affixes on a root or radical to form new word expressions in nouns and verbs Kambarami et al. [2021]. In Bantu languages noun prefixal morphemes play an important role in linking nouns to other words in the sentence, to ensure concordial agreement.

However, Bantu languages differ significantly in terms of their orthography. Here, Bantu language orthography falls on a continuum between disjunctive (white space used between morphemes) and conjunctive (no white space between morphemes). No one language is fully conjunctive or disjunctive [Taljard and Bosch, 2006]. To illustrate the difference between conjunctive and disjunctive Bantu languages consider the following two examples from Taljard and Bosch [2006]:

1. Northern Sotho, Disjunctive: Basemane ba bagolo ba ka bala dipuku – The big boys may read the books
2. Zulu, Conjunctive: Abafana abakhulu bangazifunda izincwadi – The big boys may read the books

The first example is of Northern Sotho, a Southern Bantu Language (SBL) that is disjunctive. This is observed in the spaces seen in the morphemes used in the sentence. Here, three orthographic elements (morphemes separated by white space) are used to write a single linguistic word ("ba ka bala" – "they may read").

The second example is of isiZulu, which is a Nguni language (a subgroup of the Southern Bantu Languages) that is conjunctive. Here, it can be observed that there are no spaces between morphemes used. Here one orthographic word corresponds to a single linguistic word ("bangazifunda" – "they may read them"). Note that in both cases, the combination of different morphemes can be used to convey complex semantics in a single orthographic word ("they may read .."). Morphemes provide many combinations to form many subtle semantics. Therefore, Bantu languages have many unique orthographic words. It was shown by Mesham et al. [2021] that combining conjunctive and disjunctive languages in the pretraining corpus does not significantly impact language modeling.

The above discussion implies that Bantu languages are morphologically very different from many of the languages used in multilingual modelling. For instance, they are not isolating languages like English, Mandarin or Vietnamese, which consists of a morpheme to word ratio close to one and is therefore morphologically simplistic compared to Bantu languages [Kambarami et al., 2021]. For Bantu languages, changes in the morphemes used within an orthographic word primarily are responsible for grammatical relations and semantics. This is unlike typical grammatical relations conveyed by the position of a word in a sentence, as used in English (the main language used in mBERT and XLM-R) [Hayward et al., 2020].

Furthermore, Bantu languages use diacritics to denote language specific sounds that are used to indicate tone and disambiguate morphemes [Taljard and Bosch, 2006]. Therefore, the difference in scrips used in Bantu languages and those used in multilingual models impacts the model's ability to contextually represent these languages. Additionally, Ogueji et al. [2021] compared the amount of African languages used in mBERT and XLM-R. The only Bantu language used by these models was Swahili and it consisted of 0.04% and 0.07% of the models pretraining corpus size respectively.

In summary, Bantu languages use agglutination of morphemes to convey meaning. This is a major difference to the language morphology of the high-resourced languages used in mBERT and XLM-R. This implies that Bantu languages are not well catered for in the current multilingual language modelling paradigm.

2.4.3 Bantu Languages as Low Resourced Languages

Joshi et al. [2020] investigated the context of low-resourced languages and their adoption in the natural language processing community. Joshi et al. [2020] call into question the purely “language agnostic status of current models” used for multilingual modelling. They classified the languages of the world into 6 categories. Here, the taxonomy is viewed within the paradigm of dataset quality. Two features are used for this classification, namely the number of unlabelled resources vs. number of labelled resources.

The number of unlabelled textual data sources is important to the natural language community. Large unlabelled text datasets can be used within the masked language modelling pretraining objective used by transformers [Conneau et al., 2019]. Here, no labels are required to predict a word that is inherent to the text being learned by the model - i.e., the word is the label being used in prediction. Therefore, these datasets are indicative of the ability to represent a language within a contextual word embedding generated by a transformer.

Furthermore, supervised textual data that has an annotated label per example, is also foundational to the NLP community. This is because, only through these datasets can specialised linguistic functions be learned, like sentiment detection, natural language inference and named entity recognition. These datasets are indicative of the ability to create specialised linguistic applications for users of a certain language.

Joshi et al. [2020] classified low-resourced languages into 4 categories, which consists of the following:

1. Class 0: Speakers are not digitally active and therefore there are almost no unlabelled textual datasets produced. Labelled textual datasets are non-existent.
2. Class 1: There are some unsupervised textual datasets available due to digital engagement. However, there are almost no labelled textual textual datasets for the language.
3. Class 2: There are small sets of labelled textual data available. Digital resources enable unsupervised textual datasets that are small and cannot be used in the transformer pretraining paradigm.
4. Class 3: Labelled datasets are present but requires more development for more specialised linguistic functions like natural language inference. Unsupervised datasets are large enough to be used within the transformer pretraining paradigm.

Of the languages they investigated by Joshi et al. [2020], isiZulu was classified as class 3. Here, isiZulu is spoken by 12 million speakers and represents the upper limit of the resource hierarchy of Bantu languages Mesham et al. [2021]. Other Bantu languages that are represented as class 3 languages are Swahili (16 million speakers) and isiXhosa (13 million speakers). For instance, Swahili is the official language of Tanzania, Kenya, and Uganda Martin et al. [2021].These languages all have significant speaker bases that are online and generating unsupervised text at a scale that can be used in pretraining transformers. However, the development of labelled data still requires major effort. It must be noted, that of the 300+ Bantu languages, the majority have much smaller speaker bases Pauw et al. [2012] and therefore will fall into lower resourced classes (0, 1, and 2).

The above discussion implies, that Bantu languages grouped into classes 2 and 3, may be used to create a multilingual model that uses a pretraining corpus that is closely topologically aligned to the low-resourced Bantu languages in classes 0 and 1. This will enable a better multilingual language representation to be used for these languages, while simultaneously also up-resourcing class 3 languages.

2.4.4 Bantu Language Tokenization

Bantu language's agglutinative nature influences the tokenisation strategy used in NLP applications. Whole-word tokenisation on white-space for agglutinating languages will create massive vocabulary sizes and data sparsity, making this approach sub-optimal Mesham et al. [2021]. This is because agglutinating languages can create completely new semantically different orthographic words by changing the composition of the morphemes used to create it [Kambarami et al., 2021]. Therefore, the mechanism of word generation in Bantu languages creates high uniqueness. This would result in a high vocabulary size for the language when tokenization on white-space is used. Tokenizing on white-space is sub-optimal for Bantu language modelling as the model will not generalise well on unseen data. Here, the model will encounter new words/tokens in the new Bantu text, that it has not been trained on (out of vocabulary, OOV). Furthermore, Mesham et al. [2021] state that character-based tokenisation requires large model capacities to learn very long sequences. Intuitively, if every sentence is tokenised by character, the length of tokenised sentences may explode, and require large models to handle this effectively.

Because of these issues, Bantu languages often use byte-pair encoding [Mesham et al., 2021] to create word-pieces that are thought to closely approximate the language's morphemes. Here, byte-pair encoding is a compression algorithm that starts with characters and then iteratively finds pairs of tokens the occur most frequently together [Shibata et al., 1999]. This represents a middle ground between word-level tokenisation and character-level tokenisation. Common groups of characters are grouped into tokens, thus reducing token sequence length and vocabulary size [Mesham et al., 2021].

However, byte-pair encoding is not optimal in handling morphologically rich languages like agglutinative Bantu languages. This is because, byte-pair encoding use surface form to tokenise sentences and therefore does not capture all the morphological details [Nzeyimana and Rubungo, 2022].Therefore, the sub-words found by byte-pair encoding approximates the morphemes used in the Bantu language and not the true units themselves [Park et al., 2021]. Unfortunately, including exact morphological tags for effective tokenization per Bantu language poses a difficult task, as this information is missing most of the time (as this requires a significant amount of labelled data, this is only available for class 4,5 and 6 higher resourced languages as discussed above).

For instance, Bosch and Pretorius [2017] use linguistic knowledge and morphological taggers (ZulMorph) to effectively tag morphemes within Zulu text. For many Bantu languages this degree of knowledge and technological development is not present due to many being under resourced. This makes an exact morpheme tokenization approach per language in a multilingual low-resource language modelling setting infeasible. Additionally, this would not be feasible as the tokenization strategy must generalise across multiple Bantu languages, especially when transferring to an unseen Bantu language, in a zero-shot fashion. Therefore, to pretrain on several Bantu languages in a multilingual setting, requires a generalised statistical tokenisation method like byte-pair encoding to be utilised.

Chapter 3

Dataset and Methodology

In section 3.1 Huggingface as a source of freely available Bantu text is discussed. Additionally, the various Bantu text datasets found, and how they were created are discussed. Furthermore, the methods used to clean the collated Bantu text is addressed. In section 3.2 the hyper-parameter search experiments used for multilingual language modeling is discussed. Additionally, the methodology for transferring the resulting language models is addressed. In section 3.3 the final model selection and its bench-marking are discussed.

3.1 Bantu Language Pretraining Data

3.1.1 HuggingFace as an Online Resource

Huggingface [Wolf et al., 2020] is an open-source library that contains a repository of various transformer models and language datasets. Their mission is to democratise natural language processing by making state-of-the-art NLP resources freely available. Therefore, this is an excellent resource to collate available Bantu language text (associated with other multilingual datasets) to create a pretraining corpus for multilingual language model development. Most of the datasets used in this study were uploaded on Huggingface. Huggingface contains a massive repository linguistic datasets used for NLP applications. The macro-statistics of the languages on the site can be useful in putting the subsequent collated Bantu language dataset in context with respect to other available higher-resource language data. Upon reviewing the multilingual datasets that are available and contain Bantu languages the following was found:

In figure 3.1 it was found that of the datasets and the languages they contain at Huggingface, Bantu languages (in orange) are severely under represented in terms of number of datasets containing Bantu languages. English was removed from this analysis as it contains an order of magnitude more datasets than the next language (German). This skews the image's representation. This finding is expected due to the low resource nature of Bantu languages as discussed above, and implies there are very few annotated datasets containing Bantu languages, that are freely available or that exist.

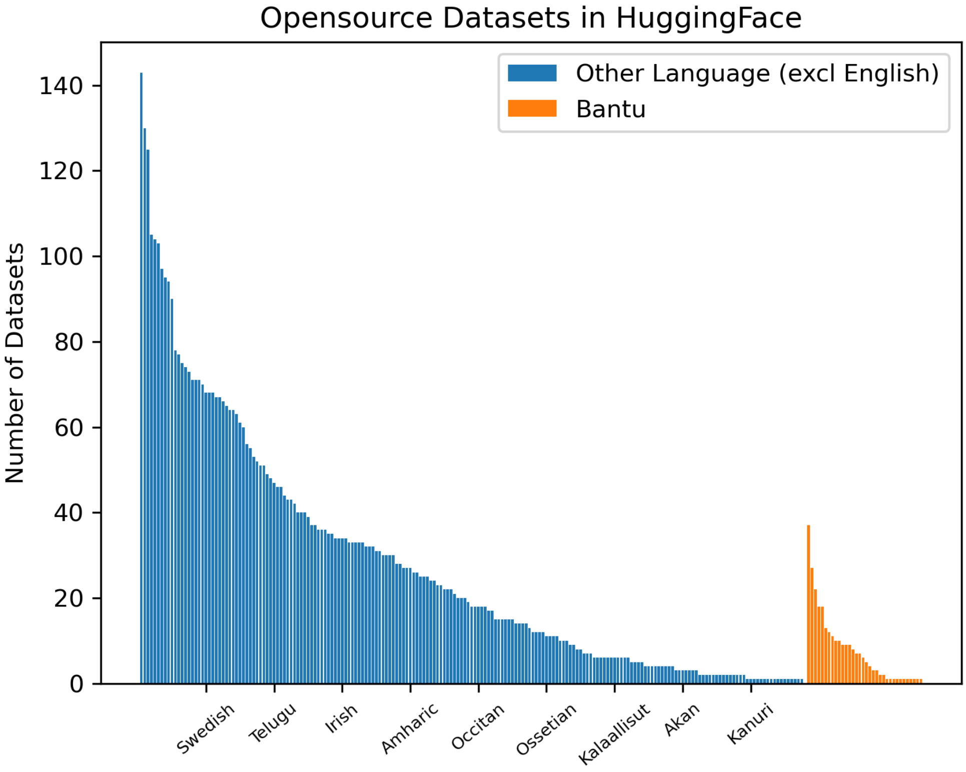

Figure 3.1: The Distribution of Language Data on HuggingFace

In figure 3.2 the distribution of Bantu languages on Huggingface was analysed. It was found that main languages represented, in terms of the number of datasets containing these languages, were Swahili, Kinyarwanda, Xhosa, and Zulu. This is expected as these languages have significant speaker bases and thus have labelled and unlabelled datasets. These languages would be classified as group 3 by Joshi et al. [2020], and thus be on the upper-end of the resource spectrum for low resourced languages. Subsequently, this implies that the dataset composed from Huggingface's available Bantu language text will be biased towards group 3 languages.

In figure 3.3 the available language models and datasets per language at HuggingFace were plotted. It can be seen that the Bantu languages exist in a depressed zone. Here, very few language models have been developed for these languages. This zone has other languages that follow this trend. This regime ends at approximately 100 datasets containing a certain language and approximately 100 language models developed to cater for the language. This zone can be thought of the low resourced language regime at HuggingFace. From here, the point of departure enters a high-resourced language regime where the number of language models available exceeds the number of datasets containing the particular language. This high-resourced language regime implies there are multiple model iterations on datasets containing the high resourced languages.

The above discussion implies, that due to the low amount of available Bantu language data on Huggingface, the subsequent collated Bantu language pretraining corpus will be much smaller than the datasets used by mBERT and XLM-R. Additionally, due to the continuum of resources

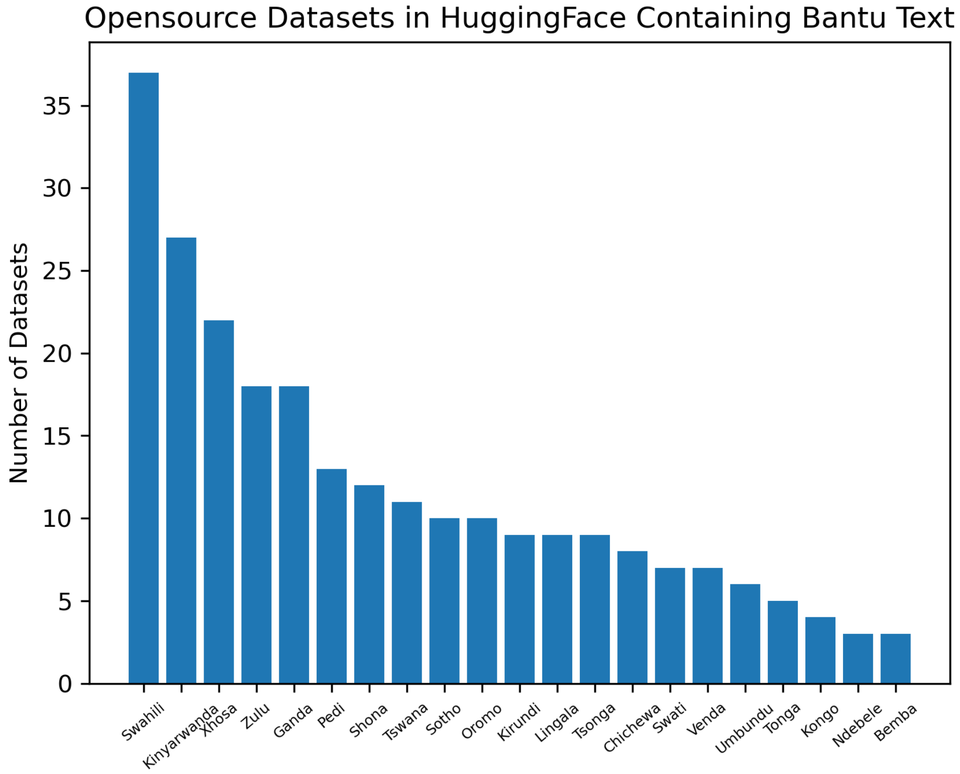

FIGURE 3.2: The Distribution of Bantu Language Data on HuggingFace

available for each Bantu language, some are higher resourced than others, and therefore, the subsequent pretraining corpus will be imbalanced and biased towards group three languages.

3.1.2 Datasets Used to Compose the Pretraining Corpus

In this section, the various datasets used to create a single collated Bantu dataset are discussed. Specific attention is paid to how these datasets were created and the potential impact this has on the downstream multi-lingual language modelling task.

3.1.2.1 Extracted Bantu Text from CC100

Common Crawl 100 (CC100) was created by Wenzek et al. [2020]. Here, the authors used the monthly snapshots on Common Crawl to create monolingual datasets. According, to Wenzek et al. [2020], ”Each snapshot contain between 20 and 30TB of uncompressed plain text, corresponding to approximately 3 billion web pages (for instance the Feb. 2019 snapshot contains 24TB of data).” This implies the dataset is curated from a large mass of web data.

To preprocess the resulting text, the authors remove duplicated text and detect the text’s language using the language detection model, fastText Joulin et al. [2017]. Here, if the model returns a probability less than 0.5, the text is classified as unidentifiable and removed. Lastly, to improve the quality of the text Wenzek et al. [2020] use language model filtering. Here, the

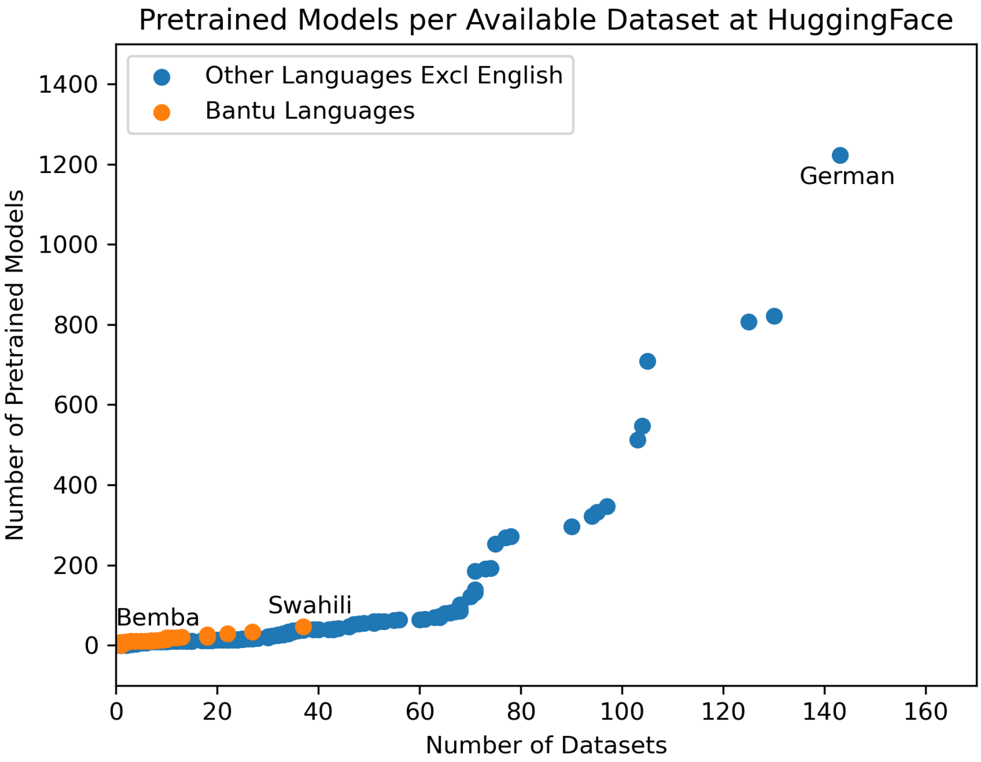

FIGURE 3.3: Language Models vs Datasets per Language at HuggingFace

language model is trained on clean Wikipedia text. This model is then transferred to unseen text of the same language. If the perplexity of the model is low on this text - the text is deemed of low quality as the model returns poor predictive results in masked language modeling. This implies the text is degraded to the point that a language model cannot use the text's context to make accurate predictions. It should be noted however that Joshi et al. [2020] recently showed ”that web-crawled multilingual corpora available for many languages, especially low-resource ones, are usually of very low quality” [Ogueji et al., 2021]. This implies that the Bantu text extracted from this data source potentially lacks the quality required for adequate multilingual modeling.

From the dataset on HuggingFace the following Bantu languages were extracted from CC100: Swahili, Luganda, Lingala, Oromo, Swati, Tswana, Xhosa, and Zulu.

3.1.2.2 Extracted Bantu Text from MC4

MC4 was created by Raffel et al. [2019]. Here, the authors sort to leverage Common Crawl as a source of text for language modeling. Additionally, like Joshi et al. [2020] they state that ”unfortunately, the majority of the resulting text is not natural language. Instead, it largely comprises gibberish or boiler-plate text like menus, error messages, or duplicate text”.

To overcome these data quality issues, Raffel et al. [2019] use the following techniques to improve the text's quality:

1. They only use text that ends with a linguistically recognised terminal punctuation mark.
2. They retain web-pages with at least five sentences.
3. All sentences with less than three words were discarded.
4. They remove all web-pages with any offensive or obscene vocabulary.
5. All web-pages with code warnings are removed.
6. Web-pages with placeholder text (”lorem ipsum”) were removed.
7. All web-pages containing code were removed.

Therefore, MC4 can be thought of as a more curated version of CC100. From the dataset on HuggingFace the following Bantu languages were extracted from MC4: Chichewa, Kinyarwanda, Shona, Southern Sotho, Swahili, Xhosa, and Zulu.

3.1.2.3 Extracted Bantu Text from the English Luganda Parallel Corpus

The English Luganda Parallel Corpus was created by Mukiibi et al. [2021]. Here, researchers from the AI and Data science research lab at at Makerere University partnered with a team of Luganda teachers, students, and freelancers. Here, the dataset was generated by translating English sentences to Luganda in an iterative, crowdsourcing approach. The dataset was intended for a machine translation task from English to Luganda.

For our application, the Luganda text component was extracted. This dataset is considered highly curated and clean.

3.1.2.4 Extracted Bantu Text from the KINNEWS and KIRNEWS Dataset

The KINNEWS and KIRNEWS dataset was created by Niyongabo et al. [2020]. Here, the authors created a classification dataset. Various news titles and article content in Kinyarwanda and Kirundi from websites and news papers were collated. The authors then annotated the dataset into various categories, based on the category in which the text was designated on the website or paper. Furthermore, the text was cleaned by removing special characters, and stopwords.

For our purposes, we extract the uncleaned Kinyarwanda and Kirundi text and remove special characters further on in the processing pipeline. This is because stopwords provide crucial context within the text, and therefore, should not be removed in language modelling. This dataset is considered clean due to the formal nature of the text used in news applications.

3.1.2.5 Extracted Bantu Text from the Openslr Dataset

The Openslr dataset was created by van Niekerk et al. [2017] for text-to-speech applications for four South African languages (three Bantu languages and Afrikaans). To create the text component of the dataset, the authors used Xhosa, Sotho, and Tswana sentences from children's stories that are freely available online. The text was then further reviewed for familiarity to the volunteer speakers, that were subsequently used to generate the audio component of the dataset.

This dataset is considered highly curated and clean as it was reviewed by native speakers. Tswana, Xhosa, and Sotho texts were extracted from the dataset.

3.1.2.6 Extracted Bantu Text from the OPUS 100 Dataset

The OPUS 100 dataset was created by Zhang et al. [2020] for multilingual neural machine translation. The authors created an ”English-centric dataset, meaning that all training pairs include English on either the source or target side”. They created the dataset from sampling from OPUS [Tiedemann, 2012]. Their selection criteria was to provide a minimum of one million training pairs per language combination. OPUS, from which OPUS 100 is curated, consists of many sources such as the Bible and movie subtitles. From this dataset other Bantu texts can be extracted like the OPUS Xhosa-Navy dataset.

The following Bantu language texts were extracted from the OPUS datasets: Xhosa, Kinyarwanda, Zulu, and Swahili.

3.1.2.7 Extracted Bantu Text from the Swahili News Dataset

The Swahili News dataset was created by David [2020]. Here, websites that provide only news in Swahili were scraped. Furthermore, it was categorized into six different topics (local news, international news , finance news, health news, sports news, and entertainment news) based on which tab it occurred under in the website.

It was not indicated how this dataset was cleaned (in terms of ensuring the quality of Swahili text and the fidelity of labelling it as Swahili text). Therefore, its quality is in question.

3.1.2.8 Extracted Bantu Text from the XL-Sum Dataset

The XL-Sum dataset was created by Hasan et al. [2021] for abstractive summarising for 44 different languages. The data was sourced from the BBC. Here, the BBC typically provides a summary of a whole article in the form of a bold paragraph containing one or two sentences at the beginning of each article.” This allowed the authors to create text (the article) and summary pairs for each language.

This dataset was well curated and is considered clean due to the formal nature of the domain (formal news) that generated the text in question. Kirundi, Swahili, and Oromo text was extracted from the dataset. Both the article and its associated summary were used as representative Bantu text.

3.1.2.9 Extracted Bantu Text from the WiLI Benchmark Dataset

The WiLI Benchmark dataset was created by Thoma [2018] as a monolingual language identification dataset. The dataset was curated from Wikipedia pages in different languages. Additionally, the data was cleaned to ensure that the average paragraph length per example was 371 characters. The WiLI Benchmark contains 11 Niger-Congo languages. Of which, Luganda, Lingala, Kinyarwanda, Tswana, Swahili, and Xhosa text were extracted for the pretraining corpus. This dataset is consider clean due to its curated nature and the domain (Wikipedia) in which the text was written.

3.1.2.10 Extracted Bantu Text from the SABC News Headlines Dataset

The SABC News dataset was created by Marivate and Sefara [2020]. Here, text from the ”Motsweding FM (An SABC Setswana radio station) Facebook Page and the Thobela FM (An SABC Sepedi radio station) Facebook Page” were collected.

There is no mention of how the data was cleaned. Sepedi and Tswana text was extracted from the dataset.

3.1.2.11 Extracted Bantu Text from the NCHLT Dataset

The NCHLT dataset was created by Eiselen and Puttkammer [2014] to develop text resources for South African languages. Here, according to Eiselen and Puttkammer [2014]: ”corpus data was sourced from South African government websites and documents, with some smaller sets of news articles, scientific articles, magazine articles and prose”. To clean the data, frequently used tokens were verified with spelling checkers. Additionally, language experts were used to check unrecognised words for the correct spelling.

From this dataset, Ndebele, Sepedi, Sotho, Swati, Tsonga, Tswana, Venda, Xhosa, and Zulu text was extracted. Due to its curated nature this data source is considered clean.

3.1.2.12 Summary

To summarise the above sections the languages collected and the associated number of datasets (used to create the pretraining corpus) per language are presented:

Figure 3.4 indicates that Swahili, Xhosa, Tswana, and Zulu are the most prevalent languages in the datasets used to create the pretraining corpus. As mentioned above, this implies that the

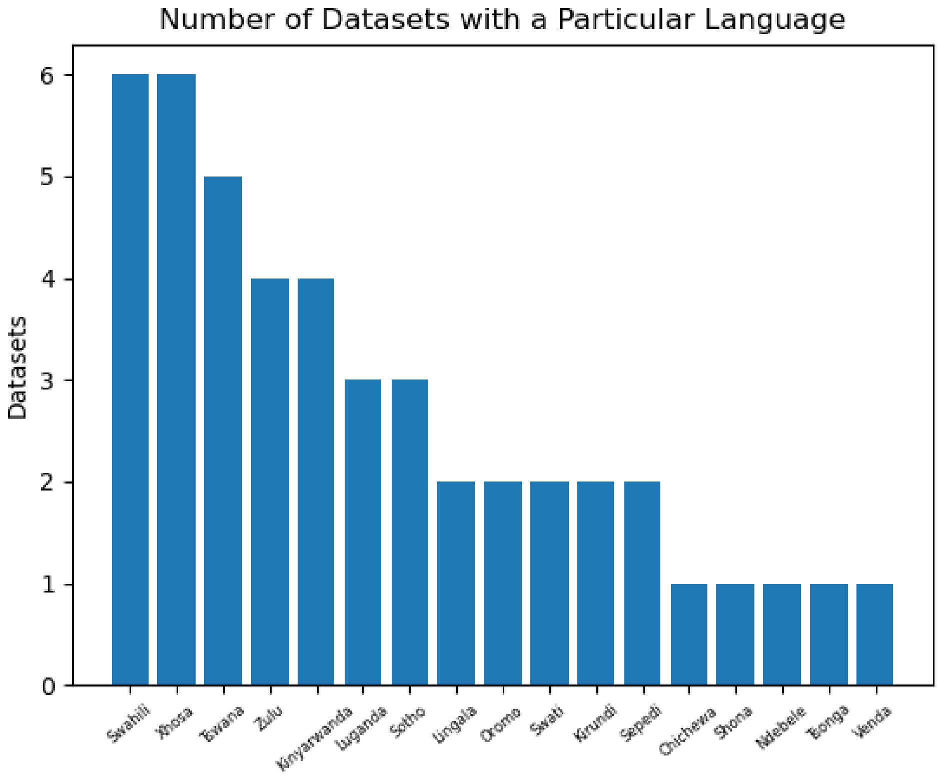

FIGURE 3.4: The Number of Datasets Containing a Particular Bantu Language

resulting corpus will be skewed towards these languages (group 3 languages according to Joshi et al. [2020]).

It Should also be noted that all of the languages presented in collated Bantu text (in figure 3.4) are Eastern Bantu languages. This implies that the resulting multilingual model will be biased towards Eastern Bantu languages. Additionally, the following dataset summary is provided in table 3.1. It should be noted that of the datasets that are considered not clean or partly clean, that these (MC4 and CC100) comprise of the largest source of text available. Therefore, considerations must be taken to clean the final dataset further as this will impact the multilingual language model pretraining. Furthermore, some of the datasets link to a common text base like Common Crawl and Wikipedia. Therefore, when processing the final pretraining corpus, duplicates of Bantu text from the same source must be removed.

3.1.3 Cleaning the Collated Bantu Text

To ensure that the cleanest possible pretraining corpus is used in pretraining, general text cleaning steps are required. Upon concatenating the Bantu text the following text cleaning steps were performed:

1. Null values were dropped as these contain no linguistic information.

Table 3.1: Datasets Used to Create the Multilingual Bantu Pretraining Corpus

Dataset	**Bantu Languages**	**Considered Clean**
NCHLT	9	*yes*
CC100	8	*no*
MC4	7	*partly*
OPUS 100	4	*yes*
Openslr	3	*yes*
XL-Sum	3	*yes*
WiLI	3	*yes*
KINNEWS KIRNEWS	2	*yes*
SABC News	2	*no*
English Luganda Parallel Corpus	1	*yes*
Swahili News	1	*no*

2. Duplicated text was removed due to the duplicity of the source of some datasets (discussed above).

3. Text that consisted solely of blank space was removed as these contain no linguistic information.

4. The text's string length was used to filter out short phrases. This was varied to generated successively cleaner text to assess its impact on modelling performance. Here, longer text sequences are surmised to contain more linguistic information, and is therefore cleaner.

5. The following special characters were removed: —, , ¡, £, ¤, ¥, ¦, §, ©, ª, ≪, ¬, ®, ¯, °, ±, ², ³, ´, µ, ¶, ·, ¸, ¹, º, ≫, ¼, ½, ¾,*, +,-,/,¨,¿. Special attention was paid to not removing any letters with diacritics as Bantu languages employ these characters.

6. @ mentions in social media were standardised to ”mentionhere”. This was done to standardise names and unique entities used in social media.

7. Hashtags in social media were standardised to ”hastaghere”.

8. FastText language detection implemented by Joulin et al. [2017], was used to detect languages in the collated Bantu corpus. The language detected probability was used to filter out detected languages that were not Bantu and had high probabilities. The filtering threshold was varied to generated successively cleaner text to assess its impact on modelling performance. This was done to assess the impact of cleaner vs dirtier Bantu text on multilingual modeling in a low-resource (few examples in pretraining) regime.

3.2 Pretraining Hyper-parameter Search

3.2.1 Hyper-parameter Experiments

To pretrain the multilingual transformer a similar approach was followed to Ogueji et al. [2021]. Here, the authors pretrained a randomly initialised XLM-roBERTa architecture (for these experiments the same is utilised) on a small amount of multilingual text composed of low resourced languages. It was shown by Zhuang et al. [2021] that BERT's training regime is non-optimal due to next sentence prediction. Therefore, roBERTa's [Zhuang et al., 2021] training regime, that uses only masked language modeling (MLM), was used in all experiments.

A hyper-parameter search is performed before settling on high-level parameters for the final multilingual model. This was done to find the optimal model depth, optimal number of attention heads, and optimal vocabulary size for this problem. Here, the searches contained shortened pretraining runs (2 epochs). Shortened runs allow for many experiments to be run in tandem to home in on the optimal parameters. These experiments produced models that were then fine-tuned on downstream NER tasks. Here, the validation sets from MasakhaNER [Adelani et al., 2021] were used to assess the impact of the hyper-parameter changes. Assessing on three different Bantu languages allows for multilingual transferal to be assessed.

Ogueji et al. [2021] showed for small pretraining data in multilingual transformer transfer that:

1. The number of attention heads are only important when the model depth is shallow: ”Shallower models need more attention heads to attain competitive performance. However, when the model is deep enough, it is very competitive with as few as two attention heads”.
2. High vocabulary sizes yields poor results on small datasets.
3. Low vocabulary sizes yields poor results due to competition between languages for tokens in a multilingual setting.
4. Deeper models always outperform shallower models. However, this performance gain diminishes with model depth. Furthermore, that at a fixed small pretraining dataset size, increasing the model depth leads to few gains.

Since deeper models were found to perform better in general, we take Ogueji et al. [2021]'s baseline of 8 hidden layers and investigate an increase to 10 hidden layers. Furthermore, since low vocabulary sizes universally perform poorly in multilingual settings, we only investigate an increase from 70k tokens to 100k tokens. Additionally, since our models are not very shallow, testing many attention heads are not very important to achieve similar performance. Also, testing very few heads makes little sense with deeper models. Therefore we test an increase from a medium number of attention heads of 6 to 8 attention heads.

In summary the following hyper-parameters were varied to assess their impact on downstream NER tasks:

1. The dataset quality (by varying the language detection threshold and the minimum text string length).
2. The vocabulary size used in BPE tokenisation was varied from 70k to 100k tokens.
3. The number of attention heads in the model's architecture was varied from 6 to 8 attention heads.
4. The number of hidden layers in the model's architecture was varied from 8 to 10 hidden layers.

It should also be noted, additionally, that due to computational constraints, the number of attention heads tested were not extensively varied past 8 attention heads.

Furthermore, it should be noted that the number of hidden layers were varied from 8 to 10. This was not extensively varied due to computational constraints and because the dataset used was comparatively small (30% smaller than Ogueji et al. [2021]'s). Kaplan et al. [2020] showed that language model performance improves when model depth and pretraining data size are scaled in tandem. Since, the pretraining dataset size remained small throughout the experiments, there was little justification to test much deeper models past 10 hidden layers.

During the hyper-parameter search, the following parameters were kept constant between runs:

1. The maximum positional embedding was held at 514.
2. Due to resource constraints, each run had a training and validation batch size of 8.
3. The learning rate was kept constant at 1e-4.
4. 40000 linear warmup steps were used.
5. Byte-pair encoding tokenization was used for all experiments.
6. AdamW optimisation [Loshchilov and Hutter, 2017] was used in each experiment.
7. The percentage of a sentence's masked tokens was kept constant at 15%.
8. All hyper-parameter search experiments ran for 2 epochs.

A total of 16 different pretrained models were created. These parameters were chosen from Ogueji et al. [2021]'s experimental runs. The hyper-parameters of each experiment are summarised in table 3.2.

It should be noted that in table 3.2 the column ”Detected Language Probability” refers to the probability of a language detected by fastText. If the language detected by fastText on the text was not a Bantu language and had a probability greater than the value displayed in table 3.2 it was removed from the pretraining corpus. The lower the threshold for this probability the stricter the criterion. Therefore, this detected language probability can be used to clean out potentially non-Bantu text. The reason for this is that fastText had Swahili text as the only Bantu language used in its language detection training. Therefore, if fastText detects a language on text (and the detection is not Swahili) with a high probability, there is a chance that the text is non-Bantu text. Here, the model is predicting with a high confidence on text that should have a low predictability due to its original identification as some other Bantu language that is not Swahili.

Additionally, in table 3.2 ”Minimum String Length” refers to the minimum number of characters (including spaces) a text may have in the pretraining corpus. If this minimum criteria is increased longer sentences will remain. The text will be cleaner as more linguistic information is present for the model to pretrain on. This logic is only approximate as text with many spaces could be present and offer no linguistic information for the model to pretrain on.

TABLE 3.2: Hyper-parameter Search Experiments in Model Pretraining

Detected Language Probability	**Minimum String Length**	**Tokenizer Vocabulary Size**	**Attention Heads**	**Hidden Layers**
0.6	30	70k	6	8
0.6	30	100k	6	8
0.6	30	70k	8	8
0.6	30	100k	8	8
0.6	30	70k	6	10
0.6	30	100k	6	10
0.6	30	70k	8	10
0.6	30	100k	8	10
0.4	90	70k	6	8
0.4	90	100k	6	8
0.4	90	70k	8	8
0.4	90	100k	8	8
0.4	90	70k	6	10
0.4	90	100k	6	10
0.4	90	70k	8	10
0.4	90	100k	8	10

3.2.2 NER Transfer Experiments

Once a hyper-parameter experiment finished running (after 2 epochs) the pretrained multilingual transformer model was transferred to a NER task by fine-tuning on MasakhaNER's Bantu datasets (Swahili, Luganda, and Kinyarwanda).

This task was chosen because NER is considered one of the most impactful NLP tasks Tjong Kim Sang and De Meulder [2003]. According to Adelani et al. [2021]: "NER is an important information extraction task and an essential component of numerous products including spell-checkers, localization of voice and dialogue systems, and conversational agents." Therefore, benchmarking on NER tasks represents a good direction for developing NLP resources in Africa. Additionally, MasakhaNER offers 3 different Bantu languages that one can test this application on. There are not, in general, many resources available for African languages where one can benchmark across multiple languages for the same task.

Here, the following parameters were kept constant when fine-tuning on the NER tasks:

1. The learning rate was kept constant at 6e-5.
2. The training batch size was 32.
3. The validation batch size was 16.
4. The number of training epochs were 50.
5. AdamW optimisation [Loshchilov and Hutter, 2017] was used for each experiment.

The best performing model checkpoint, during each run, was then tested on the associated NER validation dataset.

To quantify the impact of changing each hyper-parameter, regression analysis was performed. Here, regression models were fit using the hyper-parameter experiment values as independent variables (see table 3.2). The average of the F1 scores, across the three languages, was used as the response variable. The effect of the hyper-parameter variables were considered. In each case, the statistical significance of the variable was determined, and the effect on the adjusted R Squared score was noted. This allows for an understanding of the effect of each hyper-parameter on multilingual language modeling for Bantu languages.

Furthermore, differences in the pretraining experiment's validation loss were analysed, along with any major differences between the two datasets used in the hyper-parameter experiments (cleaner vs dirtier). This was done to understand the affects of dataset size and quality on the resulting model.

3.3 Final Model Selection and Bench-marking

The regression results from the hyper-parameter search are used to inform the following:

1. The depth of the final model and whether increasing this metric is important.
2. The number of attention heads required for the final model and whether increasing this metric is important.
3. The vocabulary size of the final model and whether increasing this metric is important.
4. The dataset used and whether using a smaller dataset that is cleaner is more important.

Once the high-level hyper-parameters of the final XLM-roBERTa model were chosen, by considering the above points, the final model is pretrained as follows:

1. The maximum positional embedding was held at 514.
2. Due to resource constraints, the final run had a training and validation batch size of 8.
3. The learning rate was kept constant at 1e-4.
4. 40000 linear warmup steps were used.
5. Byte-pair encoding tokenization was used.
6. AdamW optimisation [Loshchilov and Hutter, 2017] was used in the final run.
7. The percentage of a sentence's masked tokens was kept constant at 15%.
8. The final run occurred over 8 epochs.

From above, the number of training epochs were drastically increased. This was done as it has been shown by Conneau et al. [2020] that pretraining past the point at which validation loss effectively stops decreasing yields better cross-lingual results.

To benchmark the final model, the model is finetuned on MasakhaNER datasets (as discussed in section 3.2.2) and scored on the test datasets for MasakhaNER's Bantu datasets (Swahili, Luganda, and Kinyarwanda). Here, the F1 score's attained from the optimised BantuBERTa are compared to XLM-R and mBERT. This will allow for the comparison of BantuBERTa to much larger models, that were trained on much larger datasets. Additionally, BantuBERTa was compared to AfriBERTa. This allows for comparision with a similar implementation. Here, NER was chosen as a benchmark, to see the effect on high-level NLP task.

Furthermore, the final optimised model was fine-tuned on the African News Topic Classification (ANTC) dataset created by Alabi et al. [2022]. Here the authors created the topic classification dataset for five African languages from the following news sources: VOA, BBC, Global Voices, and isolezwe. BantuBERTa was finetuned on the classification task as follows:

1. The learning rate was kept constant at 1e-5.
2. The number of training epochs were limited to 10 epochs.
3. The batch sizes for training and validation were limited to 16 due to computational constraints.
4. AdamW optimisation [Loshchilov and Hutter, 2017] was used.

BantuBERTa was tested on the ANTC dataset's Bantu language test sets (Zulu and Lingala). It should be noted that Alabi et al. [2022] present results for Kinyarwanda and Swahili News Classification benchmarks. The Kinyarwanda News Classification dataset was created by Niyongabo et al. [2020] and the Swahili News Dataset was created by David [2020]. Both of these dataset's text occur within BantuBERTa's pretraining corpus and therefore testing the model on these datasets would constitute a form of data-leakage and bias and therefore is invalid. BantuBERTa was compared to the following models trained on the ANTC dataset: AfriBERTa and XLM-R. Classification was chosen to see the effect on a low-level NLP task.

It should be noted that for the NER and news classification fine-tuning tasks that the languages were chosen because of what was available. There is no specific reason for these languages expect for their identity as being part of the Bantu-family and being topographically similar to the base set of languages used in pretraining.

Chapter 4

Results and Discussion

In section 4.1 the results from the hyper-parameter experiments are discussed. Specifically, differences in the two datasets used are explored, the pretraining runs are evaluated, and the regression analysis on the experiments is discussed. In section 4.2 the final selected model results are discussed along with reasons for its performance.

4.1 Hyper-parameter Search Results

4.1.1 Differences in Data Used

In the hyper-parameter search experiments, two separate datasets were created. The one was cleaner than the other. This was produced by varying the minimum string length within the dataset along with fastText's cutoff language detection probability. In figure 4.1 below the predicted language probability distribution generated from fastText on the collated Bantu text is shown. The thresholds used to create each dataset are shown (threshold 1 for the dirtier dataset, threshold 2 for the cleaner dataset). Here, all text to the left of each respective threshold is used in the final pretraining corpus. All text to the right of the respective thresholds are discarded. FastText used only Swahili as the single Bantu language in its language detection training. Therefore, if it produces a high probability of detection for languages other than Swahili, this text potentially contains non-Bantu text. All text detected as Swahili was used in each corpus.

When iterating over various thresholds, the percentage of text discarded was determined in figure 4.2. One can see that at threshold 1 of 0.6, approximately 15% of the original dataset was discarded. At threshold 2 of 0.4, approximately 25% of the original text was discarded in the cleaner dataset.

Table 4.1 below indicates the high-level differences between the datasets. Table 4.1 indicates that the dirty dataset is 30% larger than the clean dataset, in terms of the total number of Bantu sentences contained in it (3.79 million sentences vs 2.66 million sentences). Table 4.1 also indicates the distribution of Bantu languages between both datasets. Both indicate that Swahili

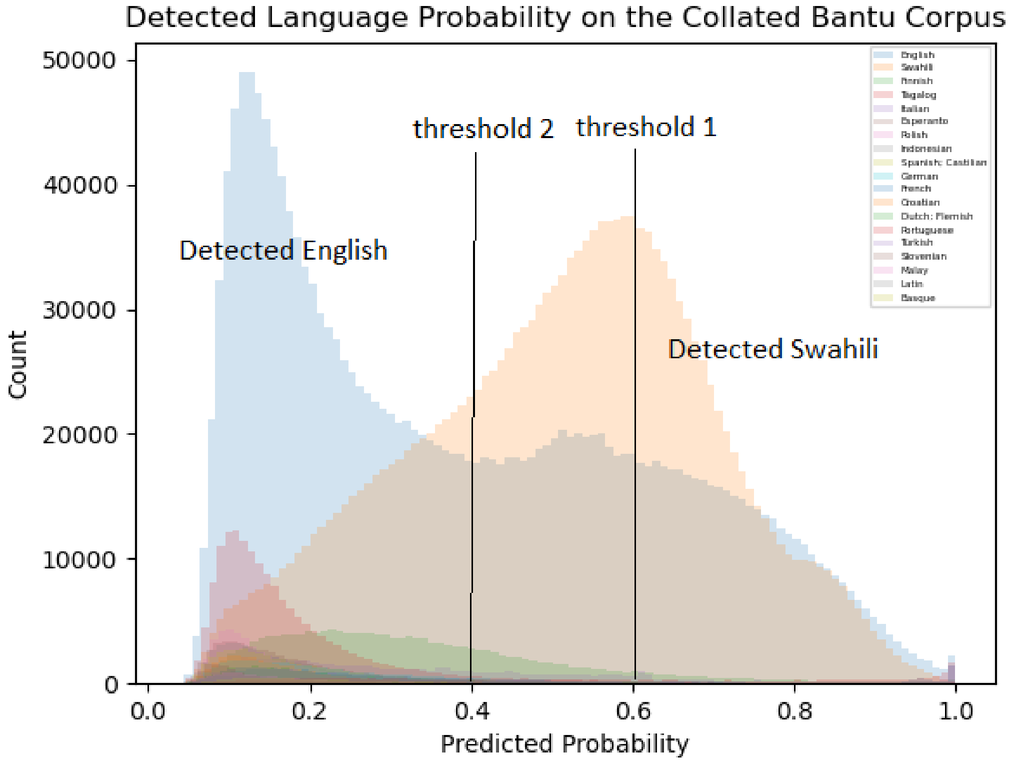

Figure 4.1: Predicted Language Probability Distribution on the Collated Bantu Corpus

is over-represented in both sets, with over 45% of the sentences as Swahili. Furthermore, the top 2-5 represented languages are all of Southern-Bantu origin. Additionally, the set contains solely East-Bantu languages. Therefore, the resulting datasets are biased, and will produce models that are biased towards Swahili, and South African Bantu languages.

It should also be noted that the Bantu languages represented in these sets, that correspond to MasakhaNER's Bantu datasets, are either very strongly represented (Swahili) or very weakly represented (Luganda, and Kinyarwanda). Therefore, assessing the subsequent transfer from models pretrained on these datasets, are appropriate as transfer under both scenarios can be tested.

4.1.2 Hyper-parameter Experiment Pretraining Runs

Figure 4.3 indicates the decrease in validation loss across pretraining epochs. The cleaner data runs are represented by '-x-' and experiments that used a vocabulary of 70k are coloured red, and those that used a vocabulary of 100k are coloured black. Furthermore, the blue dashed line represent approximately the lowest loss for these experiments. Since all hyper-parameter runs used 2 epochs in pretraining, the cleaner dataset (which has fewer examples) would have fewer training steps. Therefore, these curves end earlier on the graph.

It can be seen that the models produced from the cleaner dataset had a slightly higher final validation loss on average (when looking at the experiments using a vocabulary size of 70k). By

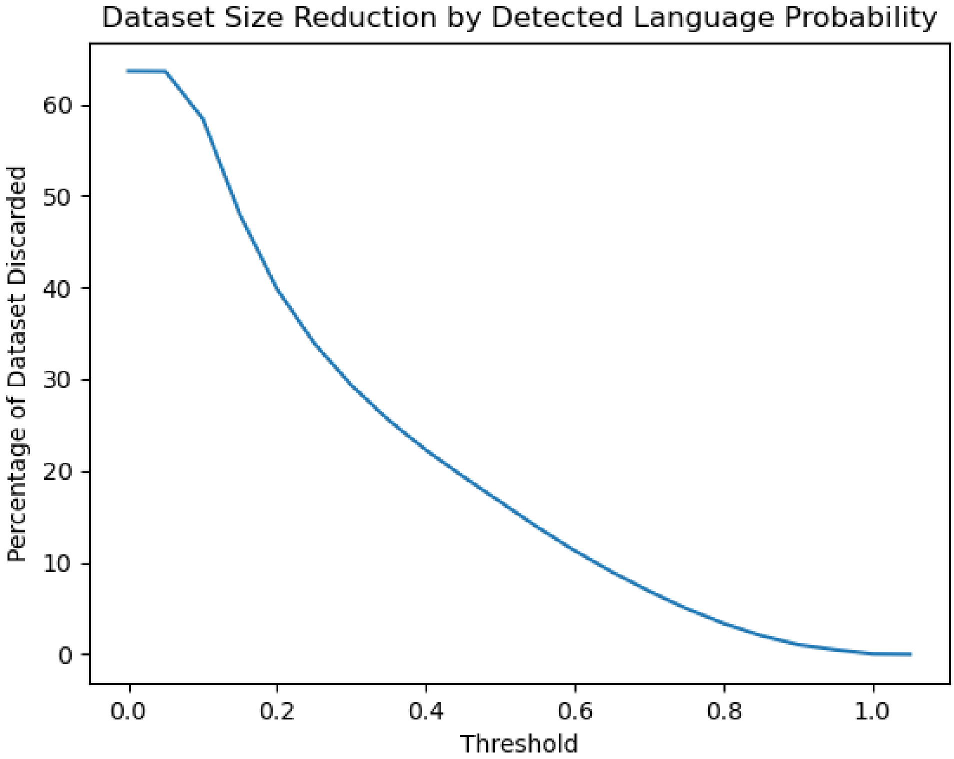

FIGURE 4.2: Percentage of Discarded Text vs Probability of Detected Language

TABLE 4.1: Language Distributions used in the Hyper-parameter Experiments

Language	**# Sent. Dirty**	**Per. Dirty**	**# Sent. Clean**	**Per. Clean**
Swahili	1712710	45.1%	1501101	56.3%
Zulu	462563	12.2%	262398	9.9%
Xhosa	408722	10.8%	247397	9.3%
Tswana	257232	6.8%	78996	3.0%
Shona	224725	5.9%	147477	5.5%
Oromo	199840	5.3%	107005	4.0%
Luganda	179050	4.7%	94792	3.6%
Chichewa	111035	2.9%	68302	2.6%
Sotho	69920	1.8%	58303	2.2%
Lingala	58307	1.5%	30090	1.1%
Kinyarwanda	45781	1.2%	23715	0.9%
Kirundi	25063	0.7%	17439	0.7%
Swati	10971	0.3%	6608	0.3%
Tsonga	7816	0.2%	5176	0.2%
Sepedi	7002	0.2%	5141	0.2%
TOTAL	3794506	100%	2663593	100%

using the same number of epochs (2) across experiments, each model saw each dataset twice in total. Therefore, the additional examples from the dirtier dataset had a positive impact on pretraining as its final validation loss is lower on average, despite being dirtier. This indicates that the number of pretraining sentences is an important factor in pretraining.

Furthermore, when viewing the change in vocabulary size between experiments, it can be seen for both datasets, that a larger vocabulary size negatively impacts pretraining. This potentially validated Ogueji et al. [2021]'s finding that states: "increasing the vocabulary size does not always yield good results on smaller datasets".

The effect of attention heads and model depth could not be clearly visualised. Therefore, these hyper-parameters are evaluated in the following section.

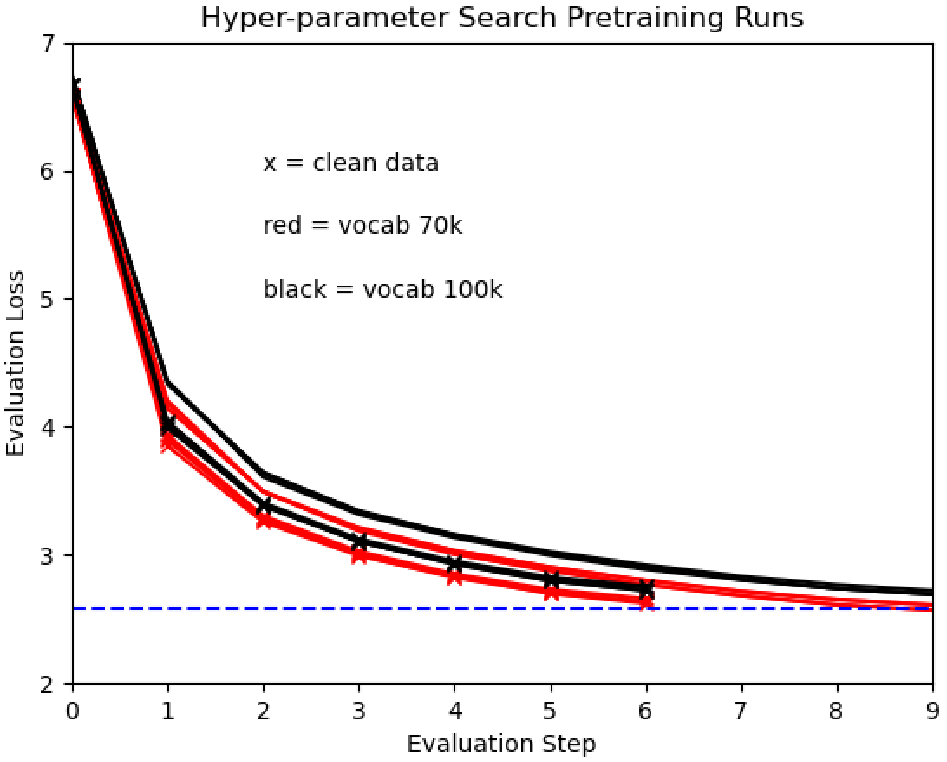

Figure 4.3: Hyper-parameter Search Pretraining Loss with respect to Evaluation Step

4.1.3 Hyper-Parameter F1 Results

Table 4.2 below indicates the best performing F1 score on MasakhaNER's Bantu validation datasets (Swahili, Luganda, and Kinyarwanda). Additionally, the average score of the three languages are shown per pretrained model. The highest performing runs are highlighted. On a high-level it can be seen that the majority of best performing runs occur for the dirtier datasets. Further analysis of the effect from hyper-parameter changes are discussed below.

4.1.3.1 Regression Analysis of All Hyper-parameters

Table 4.3 below indicates the regression analysis using all of the hyper-parameters as independent input variables. The response variable is the average F1 score across the 3 languages. Table 4.3 indicates that of all the features, hyper-parameters that impact the data quality (Language Probability and Minimum String Length) are most important. In fact, "Detected Language

Table 4.2: Hyper-parameter Search F1 Score on the Validation Set

Detected Language Probability	**Minimum String Length**	**Tokenizer Vocabulary Size**	**Attention Heads**	**Hidden Layers**	**Swa**	**Kin**	**Lug**	**Avg**
0.6	30	$70k$	6	8	0.85	0.75	0.65	0.750
0.6	30	$100k$	6	8	**0.87**	0.76	0.63	0.753
0.6	30	$70k$	8	8	0.86	0.77	0.66	0.763
0.6	30	$100k$	8	8	0.86	**0.80**	0.66	**0.773**
0.6	30	$70k$	6	10	**0.87**	0.78	0.66	0.770
0.6	30	$100k$	6	10	0.85	0.79	0.65	0.763
0.6	30	$70k$	8	10	0.84	0.78	**0.67**	0.763
0.6	30	$100k$	8	10	0.86	0.75	**0.67**	0.760
0.4	90	$70k$	6	8	0.86	0.77	0.64	0.757
0.4	90	$100k$	6	8	0.85	0.76	**0.67**	0.760
0.4	90	$70k$	8	8	0.86	0.77	0.60	0.743
0.4	90	$100k$	8	8	0.85	0.77	0.63	0.750
0.4	90	$70k$	6	10	0.84	0.79	**0.67**	0.767
0.4	90	$100k$	6	10	0.85	0.76	0.66	0.757
0.4	90	$70k$	8	10	0.86	0.77	0.63	0.753
0.4	90	$100k$	8	10	0.86	0.77	0.64	0.757

Table 4.3: Regression Analysis of All Hyper-parameters

Variable	**p-Value**	**Coefficient**	**Regression Semantics**
Language Probability	$5.9E-12$	1.07	–
Min. string Length	$1.6E-11$	$3E-3$	–
Vocabulary	0.83	$2.8E-5$	–
No. Heads	0.67	$-8E-4$	–
Depth	0.22	$2.5E-3$	–
Adjusted R Squared	–	–	0.91
Anova Significance F	–	–	$1.2E-20$

Probability" and "Minimum String Length" are the only statistically significant features (p-Value less than 0.05, at a 95% confidence interval) in this analysis. Furthermore, looking at the ANOVA F score significance, it can be seen that this regression is statistically significant and that features explain 91% of the variance (looking at the adjusted R-squared value).

One should note that the variable "Minimum String Length" is less statistically significant than "Detected Language Probability" (by looking at the respective p-Values). "Minimum String Length" has a coefficient that is essentially zero. This indicates that "Detected Language Probability" is the most important hyper-parameter effecting transfer in this NER application. Furthermore, the variable "Detected Language Probability" has a positive coefficient, indicating that an increase in the cut-off detected language probability (and thus the number of examples shown in pretraining) will positively impact downstream transfer. This is supported by the validation loss found in figure 4.3, the dirtier dataset had the lower validation loss at the same number of training epochs.

Table 4.4: Regression Analysis of Model Hyper-parameters on the Dirtier Dataset

Variable	p-Value	Coefficient	Regression Semantics
Vocabulary	0.31	$1.7E-3$	–
No. Heads	0.17	$3.4E-2$	–
Depth	0.07	$4.2E-2$	–
Adjusted R Squared	–	–	0.79
Anova Significance F	–	–	$3.7E-5$

Table 4.5: Regression Analysis of Model Hyper-parameters on the Cleaner Dataset

Variable	p-Value	Coefficient	Regression Semantics
Vocabulary	0.31	$1.8E-3$	–
No. Heads	0.26	$2.8E-2$	–
Depth	0.06	$4.5E-2$	–
Adjusted R Squared	–	–	0.79
Anova Significance F	–	–	$4.6E-5$

4.1.3.2 Regression Analysis of Model Hyper-parameters

In section 4.1.3.1 it was shown that the hyper-parameters effecting dataset quality are significantly more important than hyper-parameters effecting model architecture (vocabulary size, number of attention heads, and hidden layer depth). To understand the effect of the remaining hyper-parameters, the changes in data quality were isolated. Here, two regression models were fit per dataset. Table 4.4 represents the regression analysis performed across model hyper-parameter values only for the dirtier dataset. Table 4.5 represents the regression analysis performed across model hyper-parameter values only for the cleaner dataset. It can be seen that both tables are very similar. The p-values and coefficients for each variable are all similar across datasets. This potentially indicates that the effect of dataset quality has been isolated.

The tables indicate that the only statistically significant feature is model depth (when considering a p-value less than 0.1 at a 90% confidence interval). In both tables, vocabulary size and the number of attention heads are not statistically significant for the range tested. Therefore, considering only the model depth, one can see that the coefficient is positive. This indicates that an increase in model depth causes an increase in model performance.

4.1.4 Final Model Selection

From the above discussion the following hyper-parameters are chosen for the final model:

1. They dirtier dataset was chosen for the final pretraining dataset, as the regression results indicate that more pretraining examples are important.
2. A model depth of 10 hidden layers were chose, as this was the next most significant variable. Deeper models yielded better results.
3. The number of attention heads selected were 6. Choosing between 6 and 8 seemed to make little difference in the hyper-parameter results.
4. The vocabulary size chosen was 70k tokens.

TABLE 4.6: F1 Scores on MasakhaNER Test Data

Model	**No. Parameters**	**Swa**	**Kin**	**Lug**
XLM-R (base)	$270M$	0.874	0.739	0.807
mBERT	$172M$	0.864	0.710	0.806
AfriBERTa (base)	$111M$	0.880	0.732	0.793
BantuBERTa	$125M$	0.850	0.694	0.730

4.2 Final Model Benchmarks

4.2.1 NER Benchmarks

Table 4.6 indicates the F1 scores on the MasakhaNER test data for each model. Here, results for mBERT and XLM-R were attained from Adelani et al. [2021] and results for AfriBERTa were attained from Ogueji et al. [2021]. It can be seen that BantuBERTa under-performs for each test. However, it can be seen that each test is predictive (F1 $>$ 50%) for BantuBERTa. Here, it is clear that the multilingual nature of the pretraining corpus has allowed for transfer between languages. This is because Swahili composed the majority of the pretraining data (45%) and Kinyarwanda and Luganda compose a minority (1.2% and 4.7% respectively). This allows for the first research question to be answered. It is possible to create a multilingual pretraining corpus, solely composed of Bantu text, from freely online data. This pretraining corpus can be used to create a predictive multilingual model. However, this model does not represent sate of the art results for MasakhanER's benchmarks.

Furthermore, table 4.6 allows for the second research question to be answered. Here, pretraining on this multilingual Bantu corpus was done to see whether it benefited transfer learning to Bantu languages in downstream NLP tasks. Subsequently, BantuBERTa under-performs with respect to models that use pretraining data comprising of disparate languages. This suggests that aligning the pretraining corpus to a language family, potentially hinders multilingual language modeling. However, this may be case specific due to the nature of the data collected and is not a general indictment of the theory.

The regression results in section 4.1 suggest that dataset size is a significant factor in pretraining. Therefore, the pretraining datasets used for the models represented in table 4.7 are analysed (since it XLM-R and mBERT used much larger datasets, these are not shown). It can be seen that the collated Bantu pretraining text is 30% smaller than the data used by Ogueji et al. [2021]. Therefore, the smaller dataset size used for BantuBERTa potentially is a source for under-performance here.

Additionally, Joshi et al. [2020] raised the issue of text quality with respect to web-scraped text: ”web-crawled multilingual corpora available for many languages, especially low-resource ones, are usually of very low quality”. Since some of the datasets used in BantuBERTa were web-scraped text, the breakdown of dataset sources for BantuBERTa's pretraining corpus is shown in figure 4.4. It can be seen in figure 4.4 that over 80% of the pretraining corpus used consisted of MC4 and CC100 data. These datasets derive from monthly snapshots of Common Crawl data from web pages. This potentially implies that a significant portion of the data used was of general low quality. Therefore, this could potentially explain BantuBERTa's under-performance.

Table 4.7: Pretraining Dataset Comparison Between Similar Models

Model	**No. Sentences**	**Percentage Bantu**
AfriBERTa (base)	5.45M	19%
BantuBERTa	3.79M	100%

Table 4.8: F1 Scores on the ANTC Test Data

Model	**No. Parameters**	**Lin**	**Zul**
AfriBERTa-base (with MAFT)	111M	0.549	0.764
XLM-R-base (with MAFT)	270M	0.586	0.796
BantuBERTa	125M	0.589	0.797

There could also be other sources of under-performance. Namely, the hyper-parameter space was not extensively varied. There exists the potential for more optimised implementations with different model hyper-parameters (such as deeper models with more attention heads). Additionally, general computation constraints may have been a contributing factor. For instance, the batch-size used during pretraining was very small.

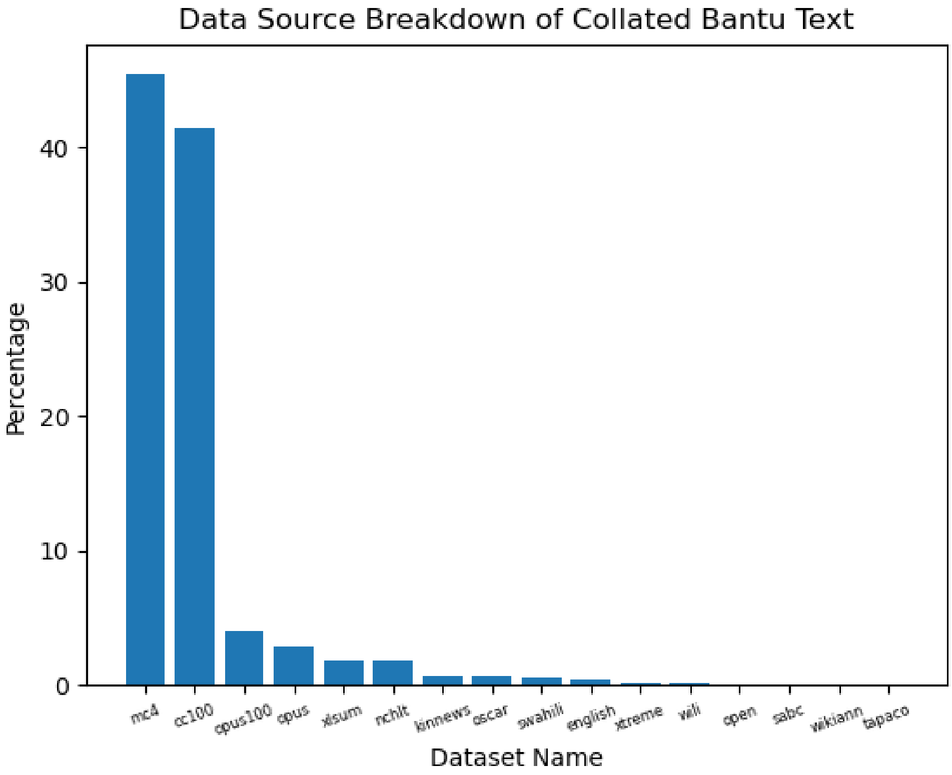

Figure 4.4: Data Source Breakdown of Collated Bantu Text

4.2.2 Topic classification Benchmarks

Table 4.8 indicates the F1 scores attained from the best performing models in Alabi et al. [2022]'s study. Note that these models used multilingual adaptive finetuning (those without performed worse than those presented). It can be seen that BantuBERTa was able to outperform XLM-R and AfriBERTa on the classification task for Lingala and Zulu. BantuBERTa was able to marginally beat XLM-R, while having much fewer parameters. This potentially implies that on low-level NLP tasks (like classification), transfer is enhanced when the pretraining corpus of the multilingual model is similar to the transfer language in question.

The above discussion validates Lauscher et al. [2020]'s finding regarding multilingual model transfer and the difficulty of NLP transfer tasks:

1. Language topography is important for low-level NLP tasks (simpler tasks like topic classification).
2. Language topography and the pretraining corpus size is important for high-level NLP tasks (more difficult tasks like NER).

Here, the pretraining dataset languages used for BantuBERTa were much more similar than those used by XLM-R and AfriBERTa for classifying Zulu and Lingala news. Additionally, for the higher-level NER task, the dataset size was smaller than XLM-R's and AfriBERTa's, and as discussed this may have had a negative impact.

Chapter 5

Next Steps and Conclusion

5.1 Next Steps

The previous section has shown that the pretraining Bantu dataset created is deficient in three areas, namely:

1. The pretraining dataset has very few examples for language modeling.
2. The data sources used to collate the pretraining data mainly come from web-scraped pages and are potentially of very low quality.
3. The dataset is biased towards higher resourced Bantu languages (Swahili, Zulu, and Xhosa). Subsequently, the language composition of the dataset comprises mainly of Eastern and Southern Bantu languages.

This potentially explains why creating a Bantu-centric language model with this data has failed for the MasakhaNER tasks. Therefore, if this study were to be repeated, sources for cleaner datasets are required for as many Bantu languages as possible. Here, these can be collected from certain domains where text is largely considered formal - such as from books published within the African country or from news outlets. These domains often have clean text that have been edited and the general language of the text is known. A caveat of this approach is that this requires much manual work identifying books and extracting text from news outlets. Additionally, due to the low-resource nature of many Bantu languages, this approach may be impossible. Another approach is to extract text from social media. However, the exact language identity and text quality cannot be assured.

Other modeling approaches could also be attempted. For instance, it has been shown that even monolingual language models can contribute to zero-shot transfer to unseen languages. In this case it may be much easier to curate high-quality text for a single higher-resourced Bantu language like Swahili or Zulu and pretrain a monolingual (or bi/tri-lingual) model that has learnt on Bantu language's unique morphology. Here, catering for Bantu-topology with a single language may prove beneficial. This may be a better approach than en masse collation of many different texts with varying quality and proportions of languages. Additionally, one could use language adaptive fine-tuning (LAFT) or multilingual adaptive fine-tuning (MAFT) to adapt XLM-R or AfriBERTa to a specific language or group of languages [Alabi et al., 2022].

Furthermore, different tokenization strategies can be explored. In this study only BPE tokenization was attempted. Different types of tokenization strategies can be tested as a hyper-parameter. For instance, WordPiece tokenisation [Wu et al., 2016], which is similar to BPE, can also be attempted. Only two NLP tasks were tested in this application. To make this study more comprehensive other Bantu NLP tasks should be utilised (if available). Additionally, the hyper-parameter space explored was very limited. Here, model depth, vocabulary size, and the number of attention heads used in the model architecture could have been explored over a wider range. Lastly, areas that were constrained due to computational power (like batch power) could also be improved by pretraining on larger machines.

5.2 Conclusion

In this study two research questions were addressed. It was researched whether a multilingual Bantu pretraining corpus could be created from freely available data. Here, to create the dataset, Bantu text extracted from datasets that are freely available online were used. The resulting multilingual language model from this pretraining data proved to be predictive across multiple Bantu languages on a higher-order NLP task (NER) and a lower-order NLP task (classification). This proves that this dataset can be used for Bantu multilingual pretraining and transfer. Additionally, it was researched whether using this Bantu dataset could benefit transfer learning in downstream NLP tasks. BantuBERTa under-performed with respect to other models benchmarked on MasakhaNER's tests. It was surmised that the pretraining dataset size and dataset quality were the main cause for this poor performance (due to dataset importance discovered in the hyper-parameter experiments). This does not imply that using language similarity in multilingual pretraining is invalid. Here, this approach could be tested on a higher-resourced language family with cleaner more bountiful text to prove the concept. We believe this is a case-specific failure due to poor data quality resulting from a dataset consisting mainly of web-scraped pages. Additionally, BantuBERTa performed well on the African News Topic Classification dataset. It beat XLM-R which has twice as many parameters as BantuBERTa. This potentially implies that on low-level NLP tasks, composing the pretraining corpus solely of similar languages with respect to the transfer language enhances transfer. This opens up a path for low-resourced languages. Grouping by language family, collating the associated text, and pretraining on that text potentially produces models that better serve other low-resourced languages within that family.

Bibliography

[Adelani et al., 2021] David Ifeoluwa Adelani, Jade Abbott, Graham Neubig, Daniel D'souza, Julia Kreutzer, Constantine Lignos, Chester Palen-Michel, Happy Buzaaba, Shruti Rijhwani, Sebastian Ruder, Stephen Mayhew, Israel Abebe Azime, Shamsuddeen H. Muhammad, Chris Chinenye Emezue, Joyce Nakatumba-Nabende, Perez Ogayo, Aremu Anuoluwapo, Catherine Gitau, Derguene Mbaye, Jesujoba Alabi, Seid Muhie Yimam, Tajuddeen Rabiu Gwadabe, Ignatius Ezeani, Rubungo Andre Niyongabo, Jonathan Mukiibi, Verrah Otiende, Iroro Orife, Davis David, Samba Ngom, Tosin Adewumi, Paul Rayson, Mofetoluwa Adeyemi, Gerald Muriuki, Emmanuel Anebi, Chiamaka Chukwuneke, Nkiruka Odu, Eric Peter Wairagala, Samuel Oyerinde, Clemencia Siro, Tobius Saul Bateesa, Temilola Oloyede, Yvonne Wambui, Victor Akinode, Deborah Nabagereka, Maurice Katusiime, Ayodele Awokoya, Mouhamadane MBOUP, Dibora Gebreyohannes, Henok Tilaye, Kelechi Nwaike, Degaga Wolde, Abdoulaye Faye, Blessing Sibanda, Orevaoghene Ahia, Bonaventure F. P. Dossou, Kelechi Ogueji, Thierno Ibrahima DIOP, Abdoulaye Diallo, Adewale Akinfaderin, Tendai Marengereke, and Salomey Osei. MasakhaNER: Named entity recognition for African languages. *Transactions of the Association for Computational Linguistics*, 9:1116–1131, 2021. doi: 10.1162/tacl_a_00416. URL https://aclanthology.org/2021.tacl-1.66.

[Alabi et al., 2022] Jesujoba O. Alabi, David Ifeoluwa Adelani, Marius Mosbach, and Dietrich Klakow. 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, October 2022. International Committee on Computational Linguistics. URL https://aclanthology.org/2022.coling-1.382.

[Bender et al., 2021] Emily Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? pages 610–623, 03 2021. doi: 10.1145/3442188.3445922.

[Bosch and Pretorius, 2017] Sonja E Bosch and Laurette Pretorius. A computational approach to zulu verb morphology within the context of lexical semantics. *Lexikos*, 27:152–182, 2017.

[Chi et al., 2020] Ethan A. Chi, John Hewitt, and Christopher D. Manning. Finding universal grammatical relations in multilingual BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5564–5577, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.493. URL https://aclanthology.org/2020.acl-main.493.

[Conneau et al., 2019] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale, 2019. URL https://arxiv.org/abs/1911.02116.

[Conneau et al., 2020] Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. Emerging cross-lingual structure in pretrained language models. pages 6022–6034, 01 2020. doi: 10.18653/v1/2020.acl-main.536.

[Currie et al., 2013] Thomas Currie, Andrew Meade, Myrtille (Mimi) Guillon, and Ruth Mace. Cultural phylogeography of the bantu languages of sub-saharan africa. *Proceedings. Biological sciences / The Royal Society*, 280:20130695, 05 2013. doi: 10.1098/rspb.2013.0695.

[David, 2020] Davis David. Swahili : News classification dataset, December 2020. URL https://doi.org/10.5281/zenodo.5514203. The news version contains both train and test sets.

[Devlin et al., 2018] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018. URL https://arxiv.org/abs/1810.04805.

[Eiselen and Puttkammer, 2014] Roald Eiselen and Martin J Puttkammer. Developing text resources for ten south african languages. In *LREC*, pages 3698–3703, 2014.

[Gogoulou et al., 2021] Evangelia Gogoulou, Ariel Ekgren, Tim Isbister, and Magnus Sahlgren. Cross-lingual transfer of monolingual models, 2021. URL https://arxiv.org/abs/2109.07348.

[Guthrie, 2017] Malcolm Guthrie. *The Classification of the Bantu Languages Bound with Bantu Word Division*, volume 11. Routledge, 2017.

[Hasan et al., 2021] Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. XL-sum: Large-scale multilingual abstractive summarization for 44 languages. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4693–4703, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.413. URL https://aclanthology.org/2021.findings-acl.413.

[Hayward et al., 2020] Luc Hayward, Stuart Mesham, and Jared Shapiro. Comparison of language modelling techniques for nguni languages. 2020.

[Joshi et al., 2020] Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 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, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.560. URL https://aclanthology.org/2020.acl-main.560.

[Joulin et al., 2017] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain, April 2017. Association for Computational Linguistics. URL https://aclanthology.org/E17-2068.

[K et al., 2019] Karthikeyan K, Zihan Wang, Stephen Mayhew, and Dan Roth. Cross-lingual ability of multilingual bert: An empirical study, 2019. URL https://arxiv.org/abs/1912.07840.

[Kambarami et al., 2021] Farayi Kambarami, Scott McLachlan, Bojan Bozic, Kudakwashe, Dube, and Herbert Chimhundu. Computational modeling of agglutinative languages: The challenge for southern bantu languages. 2021.

[Kaplan et al., 2020] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020. URL https://arxiv.org/abs/2001.08361.

[Lauscher et al., 2020] Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. From zero to hero: On the limitations of zero-shot language transfer with multilingual Transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4483–4499, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.363. URL https://aclanthology.org/2020.emnlp-main.363.

[Lin et al., 2021] Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona T. Diab, Veselin Stoyanov, and Xian Li. Few-shot learning with multilingual language models. *CoRR*, abs/2112.10668, 2021. URL https://arxiv.org/abs/2112.10668.

[Loshchilov and Hutter, 2017] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2017. URL https://arxiv.org/abs/1711.05101.

[Marivate and Sefara, 2020] Vukosi Marivate and Tshephisho Sefara. South african news data, February 2020. URL https://doi.org/10.5281/zenodo.3668495.

[Martin et al., 2021] Gati L. Martin, Medard E. Mswahili, and Young-Seob Jeong. Sentiment classification in swahili language using multilingual bert, 2021. URL https://arxiv.org/abs/2104.09006.

[Mesham et al., 2021] Stuart Mesham, Luc Hayward, Jared Shapiro, and Jan Buys. Low-resource language modelling of south african languages, 2021. URL https://arxiv.org/abs/2104.00772.

[Mukiibi et al., 2021] Jonathan Mukiibi, Babirye Claire, and Nakatumba-Nabende Joyce. An english-luganda parallel corpus, May 2021. URL https://doi.org/10.5281/zenodo.4764039.

[Muller et al., 2021] Benjamin Muller, Antonios Anastasopoulos, Benoît Sagot, and Djamé Seddah. When being unseen from mBERT is just the beginning: Handling new languages with multilingual language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 448–462, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.38. URL https://aclanthology.org/2021.naacl-main.38.

[Niyongabo et al., 2020] Rubungo Andre Niyongabo, Qu Hong, Julia Kreutzer, and Li Huang. 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, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics. doi: 10.18653/v1/2020.coling-main.480. URL https://aclanthology.org/2020.coling-main.480.

[Nzeyimana and Rubungo, 2022] Antoine Nzeyimana and Andre Niyongabo Rubungo. Kinyabert: a morphology-aware kinyarwanda language model, 2022. URL https://arxiv.org/abs/2203.08459.

[Ogueji et al., 2021] Kelechi Ogueji, Yuxin Zhu, and Jimmy Lin. Small data? no problem! exploring the viability of pretrained multilingual language models for low-resourced languages. In *Proceedings of the 1st Workshop on Multilingual Representation Learning*, pages 116–126, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.mrl-1.11. URL https://aclanthology.org/2021.mrl-1.11.

[Park et al., 2021] Hyunji Hayley Park, Katherine J. Zhang, Coleman Haley, Kenneth Steimel, Han Liu, and Lane Schwartz. Morphology matters: A multilingual language modeling analysis. *Transactions of the Association for Computational Linguistics*, 9:261–276, mar 2021. doi: 10.1162/tacl_a_00365. URL https://doi.org/10.1162%2Ftacl_a_00365.

[Pauw et al., 2012] Guy Pauw, Gilles-Maurice de Schryver, Mikel Forcada, Kepa Sarasola, Francis Tyers, and Peter Wagacha. *Proceedings of the Workshop on Language Technology for Normalisation of Less-Resourced Languages (SaLTMiL 8 - AfLaT 2012).* 05 2012.

[Pires et al., 2019] Telmo Pires, Eva Schlinger, and Dan Garrette. How multilingual is multilingual bert?, 2019. URL https://arxiv.org/abs/1906.01502.

[Pretorius et al., 2009] Rigardt Pretorius, Ansu Berg, Laurette Pretorius, and Biffie Viljoen. Setswana tokenisation and computational verb morphology: Facing the challenge of a disjunctive orthography. In *Proceedings of the First Workshop on Language Technologies for African Languages*, pages 66–73, Athens, Greece, March 2009. Association for Computational Linguistics. URL https://aclanthology.org/W09-0710.

[Raffel et al., 2019] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2019. URL https://arxiv.org/abs/1910.10683.

[Rogers et al., 2020] Anna Rogers, Olga Kovaleva, and Anna Rumshisky. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866, 2020. doi: 10.1162/tacl_a_00349. URL https://aclanthology.org/2020.tacl-1.54.

[Shibata et al., 1999] Yusuke Shibata, Takuya Kida, Shuichi Fukamachi, Masayuki Takeda, Ayumi Shinohara, and Takeshi Shinohara. Byte pair encoding: A text compression scheme that accelerates pattern matching. 09 1999.

[Strubell et al., 2019] Emma Strubell, Ananya Ganesh, and Andrew Mccallum. Energy and policy considerations for deep learning in nlp. pages 3645–3650, 01 2019. doi: 10.18653/v1/P19-1355.

[Taljard and Bosch, 2006] Elsabé Taljard and Sonja E Bosch. A comparison of approaches to word class tagging: Disjunctively vs. conjunctively written bantu languages. *Nordic journal of African studies*, 15(4), 2006.

[Thoma, 2018] Martin Thoma. The wili benchmark dataset for written language identification. 01 2018.

[Tiedemann, 2012] Jörg Tiedemann. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey, May 2012. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2012/pdf/463_Paper.pdf.

[Tjong Kim Sang and De Meulder, 2003] Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147, 2003. URL https://aclanthology.org/W03-0419.

[van Niekerk et al., 2017] Daniel van Niekerk, Charl van Heerden, Marelie Davel, Neil Kleynhans, Oddur Kjartansson, Martin Jansche, and Linne Ha. Rapid development of TTS corpora for four South African languages. In *Proc. Interspeech 2017*, pages 2178–2182, Stockholm, Sweden, August 2017. URL https://dx.doi.org/10.21437/Interspeech.2017-1139.

[Vaswani et al., 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017. URL https://arxiv.org/abs/1706.03762.

[Wang et al., 2020] Zirui Wang, Zachary C. Lipton, and Yulia Tsvetkov. On negative interference in multilingual models: Findings and a meta-learning treatment. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4438–4450, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.359. URL https://aclanthology.org/2020.emnlp-main.359.

[Wenzek et al., 2020] Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://aclanthology.org/2020.lrec-1.494.

[Whiteley et al., 2019] Peter M. Whiteley, Ming Xue, and Ward C. Wheeler. Revising the bantu tree. *Cladistics*, 35, 2019.

[Wolf et al., 2020] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-demos.6. URL https://aclanthology.org/2020.emnlp-demos.6.

[Wu et al., 2016] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144, 2016. URL http://arxiv.org/abs/1609.08144.

[Zhang et al., 2020] Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. Improving massively multilingual neural machine translation and zero-shot translation, 2020.

[Zhuang et al., 2021] Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. A robustly optimized BERT pre-training approach with post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1218–1227, Huhhot, China, August 2021. Chinese Information Processing Society of China. URL https://aclanthology.org/2021.ccl-1.108.