Exploring Cross-Lingual Learning Techniques for advancing Tshivenda NLP coverage

Ndamulelo **Nemakhavhani**

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 Jocelyn **Mazarura**

June 19, 2023

Declaration

I, Ndamulelo Nemakhavhani, 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:

Date: 2023 June 19

Abstract

The information age has been a critical driver in the impressive advancement of Natural Language Processing (NLP) applications in recent years. The benefits of these applications have been prominent in populations with relatively better access to technology and information. On the contrary, low-resourced regions such as South Africa have seen a lag in NLP advancement due to limited high-quality datasets required to build reliable NLP models. To address this challenge, recent studies on NLP research have emphasised advancing language-agnostic models to enable Cross-Lingual Language Understanding (XLU) through cross-lingual transfer learning. Several empirical results have shown that XLU models work well when applied to languages with sufficient morphological or lexical similarity. In this study, we sought to exploit this capability to improve Tshivenda NLP representation using Sepedi and other related Bantu languages with relatively more data resources.

Current state-of-the-art cross-lingual language models such as *XLM-RoBERTa* are trained on hundreds of languages, with most being high-resourced languages from European origins. Although the cross-lingual performance of these models is impressive for popular African languages such as Swahili, there is still plenty of room left for improvement. As the size of such models continues to soar, questions have been raised on whether competitive performance can still be achieved using downsized training data to minimise the environmental impact yielded by ever-increasing computational requirements. Fortunately, practical results from *AfriBERTa*, a multilingual language model trained on a 1GB corpus from eleven African languages, showed that this could be a tenable approach to address the lack of representation for low-resourced languages in a sustainable way.

Inspired by these recent triumphs in studies including *XLM-RoBERTa* and *AfriBERTa*, we present *Zabantu-XLM-R*, a novel fleet of small-scale, cross-lingual, pre-trained language models aimed at enhancing NLP coverage of Tshivenda. Although the study solely focused on Tshivenda, the presented methods can be easily adapted to other least-popular languages in South Africa, such as Xhitsonga and IsiNdebele. The language models have been trained on different sets of South African Bantu languages, with each set chosen heuristically based on the similarity to Tshivenda. We used a novel news headline dataset annotated following the International Press Telecommunications Council(IPTC) standards to conduct an extrinsic evaluation of the language models on a short text classification task.

Our custom language models showed an impressive average weighted F1-score of 60% in few-shot settings with as little as 50 examples per class from the target language. We also found that open-source languages like AfriBERTa and AFroXLMR exhibited similar performance, although they had a minimal representation of Tshivenda and Sepedi in their pre-training corpora. These findings validated our hypothesis that we can leverage the relatedness among Bantu languages to develop state-of-the-art NLP models for Tshivenda. To our knowledge, no similar work has been carried out solely focusing on few-shot performance on Tshivenda.

Keywords: Cross-Lingual Transfer Learning, Tshivenda, Low-resource NLP, XLM-Roberta, Bantu languages

Acknowledgements

I would like to express my gratitude to my supervisor, Dr Vukosi Marivate, and my co-supervisor, Dr Jocelyn Mazarura, for their invaluable guidance and support throughout this project. I am also grateful to the Panlex[1] team for providing access to their database snapshots, which were crucial for the success of this research. Furthermore, I am thankful to the University of Pretoria for providing me with the opportunity to access a vast wealth of resources that facilitated my learning. Finally, I would like to extend a special thanks to NVIDIA, HeartX[2], and Explosion AI[3] for providing academic sponsorships that supported the experiments conducted in this study.

[1] https://dev.panlex.org/api/

[2] https://labelstud.io/

[3] https://explosion.ai/

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

The development of large Pre-trained Language Models(PLM) has grown exponentially recently. As researchers push towards building models that resemble human-like capabilities, adding more data and computing seems viable. This approach has often shown promising results, especially for high-resource languages like English, Spanish, or French. Unfortunately, these benefits cannot be easily achieved for low-resource languages like Tshivenda. As a result, serious questions regarding the systematic marginalisation of under-represented populations from modern technology have been raised. Furthermore, there is growing environmental concern regarding the energy necessary to train these gigantic models. As the world becomes increasingly conscious of the need to address climate change, large corporations building models at these scales will come under increased scrutiny from environmental lawmakers.

Fortunately, there has been some significant progress made in addressing these challenges. One of the most prominent solutions is the movement towards building language-agnostic models by leveraging the power of transfer learning to share the model behaviour learnt from high-resource languages with low-resourced languages. This phenomenon is commonly referred to as cross-lingual transfer modelling. Through this exciting approach, we can enhance the inclusiveness in Natural Language Processing(NLP) applications while simultaneously reducing environmental impact by training a single polyglot model instead of multiple models restricted to only one language.

1.2 Problem statement

Modern NLP-powered applications like chat-bots, personal assistants, and search engines often struggle to understand South African languages [Duvenhage et al., 2017a], which is worrying given our increasing reliance on these tools for daily tasks. A growing number of enterprises and governments are incorporating NLP features into their systems to reduce labour expenses and improve customer satisfaction. However, these interfaces are only available in English, which is not the native language for the majority of South Africans [Lephoko, 2021]. This issue disproportionately affects elderly citizens and disadvantaged communities in rural areas. Consequently, nearly 70% of the South African population is at risk of missing out on the benefits of NLP technology [Lephoko, 2021]. Even advanced NLP models like GPT-3 have yet to convincingly comprehend most South African languages, although they seem to understand popular ones like Zulu and Xhosa.

Tshivenda is the second-least spoken Language in South Africa [Marivate, 2020]. The language is predominantly spoken in the Venda region of the Limpopo province, which the Vhavenda people inhabit. Many people from Venda often migrate to metropolitan areas in South Africa where Tshivenda is not commonly spoken. Despite the widespread use of technology and social media in these regions, there is a lack of sufficient representation of the Tshivenda language on digital platforms. This makes it a challenge to obtain high-quality corpora for NLP research. Fortunately, there has been progress in improving NLP resources for other languages in South Africa that are closely related to Tshivenda, particularly Sepedi, which shares similarities with Tshivenda due to their geographical proximity and shared etymology [Finlayson, 1987; Hellen, 2018]. Moreover, annotated datasets for Tshivenda are limited compared to Sepedi, and cross-lingual transfer learning has not yet been investigated as a means to accelerate the adoption of NLP for Tshivenda.

1.3 Research questions

Considering these hurdles, the following research questions were formulated:

- Is it viable to leverage high NLP resources from Sepedi and other popular Bantu languages in South Africa to improve the coverage of Tshivenda in NLP applications?
- What is the most effective method to develop word representations to maximise few-shot performance between Tshivenda and Sepedi? i.e. monoglot versus polyglot representations
- Are the current Tshivenda data resources sufficient for training state-of-the-art NLP models?

1.4 Contributions

Inspired by the successful cross-lingual transfer learning results from XML-RoBERTa [Conneau et al., 2020] and AfriBERTa [Ogueji et al., 2021], we plan to make the following contributions:

- Use established methods for collecting and pre-processing data to prepare a text corpus of South African languages that can be used for training multilingual language models.
- Develop Tshivenda and Sepedi static word embeddings aligned through a semi-supervised process using VecMap [Artetxe et al., 2017]

- Train a fleet of multi-lingual South African language models from scratch using the XML-RoBERTa [Conneau et al., 2020] architecture.
- Assess the quality of custom-trained language models versus state-of-the-art open-source language models by conducting a topic classification fine-tuning task on a novel news headlines dataset for Tshivenda and Sepedi.
- Review the efficiency of using multilingual models compared to monolingual and bilingual models to improve the NLP performance for Tshivenda.

1.5 Outline

The rest of this paper is organised as follows:

- Chapter 2: Reviews the recent discoveries in advancing NLP research for low-resourced languages.
- Chapter 3: This chapter discusses the methodology used to address the research questions, encompassing the description of the experiment configuration and the criteria for model evaluation.
- Chapter 4: Presents the findings from various experiments described in Chapter 3.
- Chapter 5: This chapter verifies the results' reproducibility through robustness assessments.
- Chapter 6: Analysis of Results and address limitations of the Study
- Chapter 7: Discusses the conclusions that can be drawn from the main findings

Chapter 2

Literature survey

In this chapter, we will briefly review the history of the journey towards truly language-agnostic models through transfer learning. The main focus will be on studies about enhancing Tshivenda representation in NLP applications by leveraging the plethora of resources of Bantu languages closely related to Tshivenda.

The rest of this chapter is organised as follows: Section 2.1 will cover the background and literature review of Transfer learning and how it is a catalyst for the success of cross-lingual language modelling, followed by Section 2.2 where we do a study of Tshivenda language and review existing NLP research and applications for Tshivenda. Finally, in Section 2.3, we will touch on some recent work with a similar objective to ours and highlight the identified gaps that can be addressed in this study.

2.1 Transfer learning

Transfer learning is a widely used machine learning technique that involves initialising model weights in a low-resource setting using a pre-trained model trained on a relatively large and similar dataset to improve performance. The nature of transfer learning can vary depending on the specific knowledge being transferred, with domain transfer and task transfer learning being two common methods identified in the NLP literature [Pikuliak et al., 2021]. For instance, if we train a sentiment classification model using a corpus extracted from Twitter and apply it to predict customer sentiment on e-business product reviews, we are doing domain transfer. We can also adapt this procedure to transfer the model behaviour learned from a high-resourced language to a low-resourced language [Pikuliak et al., 2021]. The high-resourced language is called the "source" language because it is the main source of knowledge which must be transferred to the low-resourced or "target" language. This is known as Cross-Lingual Learning (CLL), where each language represents a unique domain and the domain shift transpires through zero-shot or few-shot settings. Consequently, multilingual learning can be viewed as a special case of CLL in which there is more than one source or target languages [Pikuliak et al., 2021].

Some recent studies contend that instead of creating new models for each of the 6500 known languages, it is more cost-effective to transfer existing knowledge from high-resourced languages [Schuster et al., 2019; Khalid et al., 2021]. However, before this knowledge transfer occurs, the source and target languages must be projected into a shared semantic vector space to ensure comparability [Glavaš et al., 2017]. This presents a challenge of aligning similar word representations from various monolingual vector spaces. This challenge becomes more pronounced as more languages are added to the model and the similarity between the languages and the domains of their training corpora become more diverse [Pikuliak et al., 2021]. It has also been shown that training with too many languages can sometimes lead to saturation which degrades downstream performance [Pikuliak et al., 2021].

2.1.1 Cross-lingual transfer learning

Cross-lingual transfer learning is a sub-field within transfer learning that aims to create a shared semantic vector space between two or more languages to allow models trained on one language to be used on a new language with limited training data. One approach to achieving this is to use established monolingual word embedding training tools like Word2Vec [Mikolov et al., 2013a] to train embeddings for each language. A bilingual sentence-level or word-level dictionary can then be utilised to train a transformation matrix that maps the monolingual embeddings to a shared vector space [Ruder et al., 2019a; Glavaš et al., 2017]. Typically, the embedding space for the source language is frozen, while a projection matrix is learned to transform the target language embeddings into the shared space. For example, we could use a German-English bilingual dictionary to induce a joint vector space for English and German embeddings, with the English embedding space held fixed. However, creating high-quality bilingual dictionaries is difficult for many low-resource language pairs. Furthermore, although a solution like Google Translate[1] can be used, the results may not be reliable, particularly for South African languages. As a result, significant obstacles still need to be addressed to develop shared vector spaces for languages with limited resources.

An additional benefit of inducing a shared embedding space is the automatic generation of word translations, facilitating cross-lingual word meaning comparisons [Ruder et al., 2019a]. However, as with any other supervised problem, obtaining a high-quality parallel corpus makes this hard to attain in low-resource settings. Therefore, researchers have explored alternative methods for obtaining cross-lingual representations, such as semi-supervised and unsupervised approaches. Two notable open-source tools for inducing unified vector spaces between languages are VecMap [Artetxe et al., 2017], and MUSE [Conneau et al., 2017]. While both can function without a bilingual dictionary as a supervision signal, they have limitations as described in [Doval et al., 2018]. To address these shortcomings, a post-tuning method has been proposed to remove the constraint of keeping the initial monolingual embeddings unchanged. One such method, known as the *meet in the middle* technique, has demonstrated success in multilingual settings [Doval et al., 2018]. Alternatively, CLWE can be generated between two languages using an intermediary language similar to both source and target language [Sannigrahi and Read, 2022]. However, it

[1] https://translate.google.com/

should be noted that these techniques are still under active research and development, and their effectiveness may vary depending on the specific languages and domains involved.

2.1.2 Static text embeddings

Word-level embeddings, such as those generated by Word2Vec, have been widely used to represent text in high-dimensional vector spaces. However, in cases where large corpora are available, sentence embeddings can also be employed [Duong et al., 2016]. Sentence embeddings offer improved contextual information than word-level embeddings, which can be beneficial for document classification tasks with substantial word overlap among categories. Nonetheless, aligning vector spaces using parallel corpora in cross-lingual settings is often infeasible, especially in low-resource scenarios. Furthermore, the study of sentence embeddings is relatively limited compared to word-level embeddings, posing challenges for objective evaluation [Mishra and Viradiya, 2019].

Another option is to consider document-level embeddings, which capture the similarity between entire documents rather than focusing solely on words or sentences [Azunre et al., 2021]. This representation is particularly useful for tasks like information retrieval and summarisation of lengthy documents. In the case of short-text applications, n-gram level or Character-level Word Embedding (CWE) techniques are commonly employed. CWE is especially helpful for languages like Chinese, where distinct characters within a sentence carry semantic substance on their own [Chen et al., 2015]. For morphologically rich languages like many indigenous languages in South Africa, subword level embeddings with tokens induced from Byte-Pair encoding [Mesham et al., 2021] and WordPiece [Schuster and Nakajima, 2012] are recommended. These methods can effectively handle the complexities of word structure in such languages.

Established frameworks like Word2Vec [Mikolov et al., 2013a] and FastText [Bojanowski et al., 2017] generate monolingual embeddings that yield fixed global word representations, regardless of the context. FastText, in particular, enhances Word2Vec by operating on the n-gram level, enabling it to learn more accurate representations, including tokens that did not appear in the training lexicon. However, both methods face challenges when dealing with polysemy, which is prevalent in Bantu languages. For example, in the phrase "duvha li kho fhisa" (it is hot) and "linwe duvha" (someday), the word "duvha" refers to "sun" and "day," respectively. In such cases, It is preferable for the term "duvha" to have distinct vector representations based on its surrounding context. This can be easily handled by using dynamic embeddings that are automatically generated in transformer-based models like BERT [Devlin et al., 2018], ELMO [Peters et al., 2018], GPT [Radford et al., 2018], and other similar approaches. These models capture contextual information and allow words to have varying vector representations depending on their contextual usage.

2.1.3 Contextual word vector representation

The capability of contextualised embeddings can be extended to multiple languages through the use of pre-trained multilingual models like mBERT [Devlin et al., 2018] and XML-R [Conneau et al., 2020]. These models generate embeddings that are both contextual and partially aligned across languages. Contextual embeddings capture the representation of a word based on its surrounding sequence, making them well-suited for handling polysemy [Liu et al., 2020]. One of the key benefits of this approach is that the learning process is entirely unsupervised, allowing it to be easily applied in low-resource settings. Supervised objectives like CoVe [McCann et al., 2017] have also shown effectiveness by leveraging a machine translation trained model to generate contextual embeddings. However, it is important to note that CoVe relies on an English-German parallel dataset consisting of approximately 210k sentence pairs, which may be challenging to obtain for most low-resource languages. Furthermore, it should be acknowledged that the performance of multilingual representations may vary across different language pairs due to structural and script disparities [Pires et al., 2019].

Pre-trained polyglot language models have earned significant popularity due to their impressive zero-shot transfer ability [Conneau et al., 2020; Ogueji et al., 2021]. These models are trained on massive amounts of text, ranging from hundreds of gigabytes to terabytes, collected from diverse sources such as Common Crawl, Google Books, and Wikipedia [Devlin et al., 2018]. Under certain conditions, they have demonstrated superior performance to monolingual embeddings when applied to unseen languages [Pires et al., 2019; Wu and Dredze, 2020]. However, mixed results have been observed for languages with non-latin scripts or languages with lower frequency rates on the pre-training datasets [Wu and Dredze, 2020; Muller et al., 2021].

Other reports suggest that the performance of pre-trained multilingual models on truly low-resource languages may be notably inferior to that of equivalent monolingual models [Hedderich et al., 2020]. However, this only applies if there is sufficient low-resource data to train reliable monoglot models. Interestingly, high-resource languages can also be negatively affected by the joint learning approach employed in multingual training scenarios [Wu and Dredze, 2020]. Contrary to the initial promises of these models, some findings indicate that training monolingual models may be more advantageous if a sufficient amount of data is available [Wu and Dredze, 2020; Schuster et al., 2019]. This tendency is particularly prominent in scenarios where the writing style of the target languages significantly differs from that of the source languages, such as variations in the ordering of verbs, subjects, and objects [Pires et al., 2019; Muller et al., 2021].

Additionally, there is a lack of comprehensive research evaluating the applicability of pre-trained multilingual models to indigenous languages in South Africa. The closest studies conducted in this context are AfriBERTa [Ogueji et al., 2021] and [Mesham et al., 2021], which investigated different language modelling techniques for African languages. The empirical results from these studies indicate that multilingual representations may be a promising avenue to explore even with limited training data [Ogueji et al., 2021; Alabi et al., 2022].

AfriBERTa [Ogueji et al., 2021] represents a noteworthy advancement in evaluating the applicability of multilingual pre-trained models for African languages. This research builds upon the foundations established by XML-R [Conneau et al., 2020] and mBERT [Devlin et al., 2018], which do not provide adequate representation for African languages. While XML-R includes five more African languages than mBERT, they constitute only 8% of the corpus, which is insufficient given the immense linguistic diversity in Africa [Marivate, 2020]. Nevertheless, the findings from [Ogueji et al., 2021] demonstrate that even with a corpus size of less than 1GB, it is possible to achieve comparable performance to XLM-R on tasks such as entity recognition and document classification. The authors also emphasise that performance improvements were particularly prominent in zero-shot evaluations conducted on languages that share structural similarities. Similarly, [Khalid et al., 2021] suggest that shared vocabulary and typography play vital roles in enabling effective transfer across languages. These findings align with the fundamental premise of traditional transfer learning, which emphasises the importance of similarity between the source and target domains [Ruder et al., 2019b].

An approach proposed by [Makgatho et al., 2021] trains cross-lingual embeddings for Setswana and Sepedi without using extensive bilingual resources. However, the applicability of this method to distant language pairs remains uncertain, considering that Setswana and Sepedi belong to the same language family. Nevertheless, this approach was fully unsupervised and yielded commendable intrinsic results, as evidenced by the Spearman correlation and *wordsim-553* [Finkelstein et al., 2001] metrics. Similarly, good progress has been made to adapt BERT models to Twi, the most spoken language in Ghana [Azunre et al., 2021]. Unfortunately, the reliability of the results was compromised due to the evaluation being conducted on exceedingly small sentiment analysis datasets, comprising only 20 observations.

While substantial strides have been made in adapting cross-lingual learning techniques for African languages, considerable work remains, especially in constructing robust benchmark datasets and assessing the performance on more intricate tasks like question answering or entailment. Moreover, there is a need for more research to explore the effectiveness of multilingual pre-trained models on distantly related language pairs. We also not that the development of transfer learning techniques for African languages is still in its early stages, and there is a need for more research to address the challenges and limitations in this regard. Continued research and development efforts to address these gaps can lead to substantial advancements in natural language processing for African languages. By bridging these technological divides, we can enable greater linguistic diversity and inclusivity in Artificial Intelligence (AI), paving the way for practical solutions and positive social impact.

2.1.4 Word embedding alignment

Word embedding alignment is highly desirable when working with embeddings within a shared vector space. One of the notable advantages of large pre-trained language models is their ability to automatically generate partial alignment at the token level without the need for supervision [Pan et al., 2021]. However, when dealing with joint multilingual learning scenarios, the training corpus often originate from diverse languages, each with its unique syntax, semantics and word structure. As a result, a significant misalignment is usually observed within the induced shared contextual embedding space [Huang et al., 2021]. Fortunately, numerous approaches have been developed recently to improve these partial alignments through post-pre-training alignment stages. Notably, post-aligned models have reported significant improvements, even in zero-shot scenarios [Pan et al., 2021]. This process primarily involves utilising a Translation Language Modelling objective at both the word and sentence levels [Ruder et al., 2019a].

According to [Ruder et al., 2019a], cross-lingual embedding models may vary in their implementation, but they aim to optimise a common objective. These models typically use mapping techniques to align independently trained embedding spaces into a unified space using a linear projection matrix. While supervised approaches are practical when a parallel corpus is available to train the matrix, there are instances when limited parallel corpora or bilingual dictionaries make this approach infeasible. In such cases, self-learning techniques can be used to align embedding spaces.

Unfortunately, most existing methods for word embedding alignment require large parallel corpora, which poses a challenge for low-resource languages [Pan et al., 2021; Huang et al., 2021]. Additionally, aligning embeddings across different languages with varying sentence structures can be difficult due to the dynamic nature of contextualised embeddings. To address this issue, a potential solution involves utilising the first principal component from the contextualised embedding space as a static representation for words with multiple meanings [Ethayarajh, 2019]. Although this results in static embeddings that lose some contextual information, they can still be a good alternative to traditional static embeddings generated by methods like GloVe or FastText [Ethayarajh, 2019]. Moreover, integrating contextual embeddings with global embeddings has demonstrated promising results in tasks such as bilingual lexicon induction [Zhang et al., 2021].

Recent research has aimed to reduce the reliance on large parallel corpora for post-training alignment of multilingual embeddings [Duong et al., 2016]. One approach to address this challenge is presented in [Artetxe et al., 2017], which proposes a self-learning method that requires only a small number of word pairs (e.g., as little as 25) to generate word alignments. Although this is useful when no parallel data is available, better results are typically achieved with larger dictionaries [Artetxe et al., 2017]. Since post-training alignment is computationally expensive, [Huang et al., 2021] Proposes a robust training methodology that incorporates adversarial training and random smoothing to improve the tolerance of contextualised embeddings to potential word misalignment. The authors argue their approach is preferable to coerced alignments, which may be infeasible to achieve perfectly in most cases. In other studies, human judgement is proposed to fix the alignments using syntactic features; however, this approach may not scale well [Huang et al., 2021].

2.1.5 Model size

Model Size In the era of pre-trained language models, the size of polyglot models has continued to expand in terms of trained parameters, computation time, and coverage of tokens. Several studies have demonstrated that incorporating more training data, epochs, and languages generally improves performance on downstream tasks. For instance, XML-R [Conneau et al., 2020] achieved performance enhancements in cross-lingual topic classification, Entity Recognition (ER), and question answering compared to previous iterations like XLM and mBERT. These improvements were attained by incorporating additional languages and utilizing approximately 2.5TB of text data. These findings indicate significant progress in developing a robust language model capable of transferring knowledge to unseen languages [Khalid et al., 2021]. However, concerns have emerged regarding language models' escalating size and environmental impact due to prolonged computational hours. Subsequently, several studies have suggested that including more languages in these models may result in performance saturation for downstream tasks [Khalid et al., 2021; Artetxe et al., 2017]. Some authors have referred to this phenomenon as the "curse of multilinguality" [Conneau et al., 2020; Khalid et al., 2021].

Some emerging studies in NLP have challenged the notion that machine learning models perform better with larger training data. While large-scale pre-trained models such as GPT [Radford et al., 2018] and RoBERTa [Liu et al., 2019] have achieved impressive results, some researchers have shown that smaller monolingual models trained on limited data [Schuster et al., 2019] can outperform their multilingual counterparts trained on massive datasets. For example, state-of-the-art performance has been achieved for popular African languages using just a small subset of the data used for large pre-trained models [Ogueji et al., 2021]. This discovery is particularly significant for low-resourced languages with limited training data, as it allows for training high-performing polyglot models using just a few megabytes of text data. However, the extent to which we can improve multilingual models without compromising individual language performance is still an open research question that requires further investigation.

New areas of research to reduce model sizes using compression have recently gained traction. A study by [Ogueji et al., 2022] found that contrary to findings by previous studies, compression may help improve the performance of multilingual models. The authors found that compressing multilingual models can improve performance on less represented languages in models such as mBERT. This is an exciting development as it offers a potential solution to the issue of increasing model sizes while maintaining or improving performance, especially for low-resource languages. However, further research is needed to fully understand the impact of compression on multilingual models and determine the best compression techniques for different models and languages.

2.1.6 Model Evaluation

Benchmarking datasets play a critical role in reliably assessing the performance of cross-lingual models in intrinsic and extrinsic settings. They are also valuable in comparing multiple model versions which aim to solve similar problems. Currently, the scarcity of high-quality benchmarking datasets hinders our ability to fully explore the capabilities of cross-lingual language models, particularly in low-resource and high-resource settings [Cruz et al., 2020; Cruz and Cheng, 2019]. Cross-Lingual Natural Language Inference (XNLI) [Conneau et al., 2018] is a widely used dataset for extrinsic evaluation of cross-lingual language models across various Cross-Lingual Understanding (XLU) tasks. While it includes some African languages, such as Swahili and Urdu, support for Tshivenda is currently lacking. Additionally, WordSim-553 [Finkelstein et al., 2001] provides valuable resources for intrinsic evaluation, although Tshivenda support is not yet available. Generally, most high-quality benchmarking datasets focus on languages with higher resource availability. The development of benchmarking datasets for low-resource languages, including local South African languages, is still in the early stages. This is partly due to the prevalence of English or Afrikaans as the dominant languages for information publication [Marivate et al., 2020]. Some authors attribute this limitation to the scarcity of trained linguists who can create reliable bilingual resources for Bantu languages [Mashamaite, 2010]. Traditionally, South Africa has emphasised bilingual dictionaries for translating English or Afrikaans vocabulary to Bantu languages [Mashamaite, 2010]. However, limited efforts have been made to develop similar resources for parallel Bantu-to-Bantu word and sentence pairs. A hub-and-spoke model was proposed in [Mashamaite, 2010] to automate the induction of these bilingual lexicons for Tshivenda and Sesotho. Unfortunately this has not been widely adopted and extended to other Bantu language pairs yet.

However, with recent endeavours to enhance NLP resources for Africa by organisations including Masakhane [Orife et al., 2020], North-West University [Barnard et al., 2014], and Knowledge for All Foundation[2], numerous high-quality datasets have been made publicly available for research. Therefore, the number of local datasets is likely to increase soon. As an interim measure, it is recommended to collect sufficient data from public sources, including local news media, the Bible [Christodouloupoulos and Steedman, 2015], Common Crawl, or social media data [Khalid et al., 2021] to develop baseline models. Valuable guidelines for curating evaluation datasets for two South African languages, Setswana and Sepedi, were published by [Marivate et al., 2020] considering that text processing methods for low-resource languages may differ from traditional approaches used for English. Data augmentation is also a helpful tool to explore when dealing with under-resourced languages given the limited sources for local corpora [Marivate et al., 2020]. Furthermore, new efforts must be made to promote participation from diverse economic sectors in enhancing NLP datasets for South African languages by institutionalising the utilisation of native languages whenever feasible. For example, valuable data can be sourced from government records, public news broadcasts, and technical writings in the private sector [Marivate, 2020].

Another valuable approach to expanding indigenous datasets involves the application of automatic language identification. As internet and social media users continue to grow, significant volumes of indigenous text data are generated daily. Extensive resources like Common Crawl and Wikipedia may also contain data from local languages. We can tap into these vast text resources by leveraging automated scraping tools alongside native language identification models to enrich our limited NLP resources. Bayesian classification models have been utilised with impressive accuracy in identifying South African language families, as demonstrated in previous research by [Duvenhage et al., 2017b]. However, it is worth noting that these models have certain limitations. One prominent limitation is their struggle with high-dimensional feature spaces, which is often encountered in NLP tasks.

Evaluating cross-lingual embeddings can be approached through intrinsic or extrinsic assessments. Intrinsic methods focus on assessing the mathematical relationships between the word vectors of different languages within a unified vector space. The assessment can be conducted using manual evaluation through human judgement or by quantifying quality metrics such as perplexity or cosine similarity. Similarity metrics are often measured between monolingual vectors and a word similarity rating dataset like wordsim-553 [Finkelstein et al., 2001]. On the other hand, extrinsic evaluation is preferred to measure the quality of cross-lingual representations. This involves using the embeddings as inputs on a downstream task, such as bilingual lexical induction or document classification [Ruder et al., 2019a]. For example, in document classification, a model's zero-shot or few-shot performance trained solely on the source language can be

[2] Knowledge For All Home - https://k4all.org/

evaluated on an unseen language. If the downstream model demonstrates good performance in zero-shot or few-shot scenarios, it signifies the efficacy of the cross-lingual embeddings.

Text classification is one of the fundamental tasks found in natural language processing (NLP) that involves categorising text or documents into different topics or classes. It finds applications in various domains, such as news classification, sentiment analysis, website categorisation, and intent detection. However, advancements in text classification have predominantly focused on languages with abundant resources, posing a challenge for low-resource languages. Nonetheless, recent efforts have aimed to overcome this limitation. For example, a multi-class news article classification was performed in a study by [Niyongabo et al., 2020] for two African languages: Kirundi and Kinyarwanda. The study used cross-lingual transfer learning by leveraging the similarity between these low-resource languages, utilising the relatively larger annotated dataset available for Kinyarwanda. It is important to note that this approach may not be readily applicable to inherently different languages, and it relies on the availability of resources in at least one of the languages. Furthermore, the specific metrics used to assess the homogeneity between the languages in the study have not been disclosed.

Traditional classification models like Logistic Regression, Support Vector Machines, and Multivariate Naive Bayes have commonly been employed as baseline models in text classification tasks. Before the advent of transformer-based architectures, LSTM (Long Short-Term Memory) models were popular due to their ability to capture long sequences to some extent. However, their long-term memory capacity is limited compared to transformer models, which have emerged as the state-of-the-art choice for most Natural Language Understanding (NLU) tasks [Wolf et al., 2020]. It is common to encounter classification models that combine sequential models such as Bi-LSTMs, CNNs (Convolutional Neural Networks), or Char-CNN (Character-level CNN). These classic techniques typically utilise monolingual word embeddings, where each language occupies a separate vocabulary space. This approach restricts the potential for cross-lingual transfer unless the source and target languages are highly similar. In contrast, using multilingual embeddings offers a more efficient learning process by training a single model that can be applied to multiple languages. However, this approach requires more training data and a parallel corpus for alignment [Oladipo et al., 2022].

2.2 Tshivenda and Southern Bantu Languages

In this section, we will explore the distinct characteristics of the Tshivenda language and its connections with other Bantu languages used in South Africa.

2.2.1 Tshivenda

Tshivenda is recognised as one of the eleven official languages in the Republic of South Africa(RSA). Its primary concentration of native speakers lies in Limpopo and Gauteng, while reports also indicate the presence of Tshivenda speakers in neighbouring countries like Zimbabwe [Hellen, 2018]. Tshivenda is classified as a Bantu language alongside eight other official languages in

South Africa (excluding English and Afrikaans). In contrast to better-known indigenous South African languages, Tshivenda lacks extensive digital linguistic resources. While numerous bilingual lexicons exist for Tshivenda to English and Tshivenda to Afrikaans, direct translations between Tshivenda and other Bantu languages remain limited. Beyond the challenges posed by data availability, NLP applications for Tshivenda require special consideration due to its distinctive characteristics, including euphemism, diacritics, and homonymy. Code-switching is also prevalent in Tshivenda, similar to the other eight indigenous languages in South Africa, particularly in urban areas with diverse populations. Compared to other native RSA languages such as Zulu, Xhosa, and Sotho, Tshivenda is relatively under-resourced and does not receive coverage from prominent language tools like Google Translate[3].

2.2.2 South African Bantu languages

The native languages in South Africa can be organised into four major family groups: Nguni, Sotho, Tswaronga, and Venda. These language families are widely believed to have derived from a shared ancestral origin [Finlayson, 1987]. It is believed that urbanisation, mixed-race marriages [Hellen, 2018], and industrial activities like farming and mining also play a crucial role in language similarity, often leading to the borrowing of words among different cultures. For example, the word “diphrofesenale” in Sepedi is directly borrowed from the English term “professionals”. Similarly, the word enter is “dzhena” in Tshivenda, “ngena” in Zulu and “kena” in Sepedi. Some compelling evidence by [Finlayson, 1987] also show the possibility of an intermediate ancestor language between Nguni and Sotho languages and between Sotho and Tshivenda.

The connection between Venda and Sepedi is evident through the similarities in word sounds. For example, the words “toropo” (Sepedi) and “dorobo” (Venda) both mean “town,” while “welago” (Sepedi) and “welaho” (Venda) both mean “fell,” and “digwedi” (Sepedi) and “minwedzi” (Venda) both refer to ”months”. Within the Nguni family, Zulu and Xhosa are considered dialects, while Northern Sotho, Southern Sotho, and Setswana exhibit significant mutual intelligibility. These linguistic interconnections make these languages suitable candidates for cross-lingual language modelling. However, it’s important to note that Nguni languages tend to be conjunctive, unlike Sotho and Venda, which are disjunctive [Mesham et al., 2021].

2.2.3 NLP coverage

The growing public awareness of the digital divide, which stems from unequal access to technology, has prompted the NLP community to re-evaluate the benefits of current state-of-the-art methods, particularly in low-resource settings. Most existing methods in NLP focus on Indo-European languages, and it remains uncertain whether these approaches perform equally well on truly marginalised languages [Pikuliak et al., 2021]. Addressing this issue is crucial to mitigate social dilemmas arising from over-generalisation and implicit bias in high-resource datasets lacking diverse cultural representation [Hovy and Spruit, 2016].

[3] https://translate.google.com/

2.2.4 Language similarity

Linguistic features such as language relatedness, typology and grammar play an essential role in selecting the correct NLP tool for different languages. For instance, language modelling for RSA may work well with Byte-pair Encoding tokenisation compared to white space tokenisation, given the morphology richness [Mesham et al., 2021]. Hence, when doing cross-lingual learning, it is useful to explore how similar the source and target languages are to generate a insightful hypotheses. So far, researchers are divided on the hypothesis that multilingual models exhibit commendable zero-shot performance because of shared vocabulary, language similarity, or universal language features [Pikuliak et al., 2021]. However, It has been reported that downstream performance can be enhanced by leveraging the similarity of languages, as cited in [Nyoni and Bassett, 2021].

Language Relationships

Southern Bantu languages share close linguistic ties primarily due to their shared origins. However, it is beneficial to establish a metric system that measures the impact of this relationship in cross-lingual settings to facilitate reliable performance comparisons. For instance, the concept of cognacy was utilised in a study by [Borland, 1986] to assess the homogeneity among Nguni, Venda, Tswaronga, and Shona language families. Cognacy refers to the presence of shared root terms among different languages. One approach involves computing the taxonomic distance coefficient [Sokal, 1966] on a small basic word vocabulary list to determine the percentage of cognacy between two languages [Borland, 1986]. Another convenient method for evaluating similarity is using the World Atlas of Language Structures (WALS) [Dryer and Haspelmath, 2013], which offers valuable information on language families, origins, and countries where the languages are spoken.

According to the findings of [Borland, 1986], Venda exhibits a high degree of cognacy with the Tswaronga and Nguni language families. There was also a significant cognacy observed between Tshivenda and Shona, which is likely attributed to borrowing [Borland, 1986], geographical proximity between Limpopo and Zimbabwe, migration patterns [Finlayson, 1987], or other historical factors. Additionally, it is essential to acknowledge that globalisation has increased interactions and coexistence among individuals from diverse cultures and languages. Moreover, historical migration patterns contribute to the linguistic similarities observed across various African languages, as demonstrated in a study on languages in Cameroon [Tikeng et al., 2021]. Furthermore, some studies suggest that the overall homogeneity among most African languages implies the existence of a common ancestral language until recently [Finlayson, 1987].

2.3 Related work

Cross-lingual language model training, aimed at advancing low-resource language representation, has garnered significant attention in NLP research. A recent development in this area is AfroXLMR [Alabi et al., 2022], a large pre-trained model developed from MLM adaptation on XLMR for 17 African languages. While the model includes three South African Bantu languages in the training set (Sesotho, IsiXhosa, and IsiZulu), it does not yet cover less popular Bantu languages such as Tshivenda. Other similar studies have focused on developing less data-intensive techniques to overcome the challenge of limited access to high-quality training corpora [Lee et al., 2021]. For example, empirical results by [Lee et al., 2021] have shown that utilising domain adaptation on a large language model pre-trained in English yields superior performance in downstream tasks compared to monolingual training or multilingual LM fine-tuning while using fewer data points.

Image-based techniques have also emerged as a promising alternative in cross-lingual language representation learning. For instance, in [Rust et al., 2022], the authors employed an image-based approach that utilises character pixels to learn cross-lingual representations rather than the traditional masked language modelling. This method aims to overcome the challenge of increasing vocabulary size as more languages are included in large language models. Specifically, the approach leverages the script symbol through the co-activation of pixels on text images. However, it is worth noting that this approach underperforms compared to BERT architectures on Latin scripts, although it outperforms BERT in some zero-shot settings.

Although the reasons why LLMs are effective for multilingual tasks are still debated [Pires et al., 2019], numerous studies have suggested that language similarity plays a crucial role. For instance, in another study, it was shown that fine-tuning using multiple Indo-Aryan languages from the same language families produced better results than fine-tuning each language individually [Dhamecha et al., 2021]. It was further indicated that not all related languages benefit from downstream task performance, hence a forward or backward selection process is necessary to get the best combination of languages. In cases where scripts differ, transliteration is used to normalise the text to use the same set of symbols, although this was shown to have an insignificant impact on the performance. Although most of these works show promising results, they focus only on a subset of languages. However, it is plausible to believe that these techniques should also be easily transferable to other language families.

2.4 Summary

This section presented a comprehensive literature review on cross-lingual methods focused on low-resource languages. Additionally, we highlight related work that has been published in the field of Natural Language Processing for the Tshivenda language and other African languages. Our review focuses on techniques, including transfer learning, multilingual embeddings, and other related methods to enhance low-resource language coverage in NLP tools. Furthermore, we discuss the evaluation metrics used to measure the performance of NLP systems. This review provides a foundation for our proposed methodology, which will implement cross-lingual learning techniques to advance Tshivenda NLP applications. The next section will discuss the proposed methodology enacted to achieve the research objectives presented in Section 1.4.

Chapter 3

Research methodology

3.1 Overview

This section will discuss the methods used to address the primary research question of whether cross-lingual language models can improve Tshivenda's coverage in modern NLP applications. It will highlight the process we followed to achieve our objectives, including collecting and analysing data, modelling, designing experiments, and evaluating results. Our study uses a quantitative approach, a widely accepted standard in NLP research, allowing for reasonable comparisons with related works.

3.2 Data collection

Data collection encompasses the tools, platforms, and licenses used to acquire data from public or private sources. To achieve the research objectives, two distinct groups of datasets were prepared. The first group was used for creating word embeddings and training transformer-based language models. In contrast, the second data group was used for training topic classification models to evaluate the quality of the embedding models.

3.2.1 Representation learning corpora

The primary datasets for training the representation models were extracted from publicly available corpora from diverse sources. We downloaded all datasets directly without using web crawlers, as we only needed a subset of the extensive data repositories for our experiments. First, we obtained processed corpora from the South African Centre for Digital Language Resources (SADiLaR) [Eiselen and Puttkammer, 2014] website, which included monolingual and aligned sentences for *9* South African Bantu languages. Using this dataset, we extracted over 330MB of text, comprising approximately 431k sentences, with a unique token to total token ratio of 5%. In addition, we extracted at least 25MB of Nguni and Sotho family texts from the

Flores Multilingual Neural Machine Translation (NMT) dataset [Team et al., 2022; Goyal et al., 2021; Guzmán et al., 2019]. It is worth noting that the aligned text dataset was significantly smaller, revealing how preparing parallel datasets is generally more complex than scraping raw texts from the web.

Furthermore, we extracted an additional 169MB of texts from the *Monolingual Datasets from Web Crawl Data* [Wenzek et al., 2020] (referred to as CC-100) website. Other significant sources of data used in our research include public PDF documents from various South African government websites, the *Massively Multilingual Translation* [Aharoni et al., 2019; Tiedemann, 2012; Zhang et al., 2020] (OPUS-100) website, and the *Leipzig Corpora Collection* [Goldhahn et al., 2012]. A summarised view of these datasets is shown in Table 3.1.

Source	**Language**	**TTR**	**Size (MB)**
SADiLaR	ven, tso, sot, nso, tsn, zul, ssw, xho, nbl	706k/13.2m	330
Flores200	nso, sot, ssw, tsn, tso, cho, zul	68k/356k	2.25
OPUS-100	xho, zul	310k/3.8m	25
CC-100	nso, ssw, tsn, xho, zul	1.6m/27.5m	169
Leipzig	nso, sot, tsn, ven, xho, zul	-	283
Total			809.25

Table 3.1: Primary raw data sources

A total of 1.4 million sentences were extracted from the CC-100 corpora, with a mean Token-Type Ratio (TTR) score of 11.6% for the five languages as shown in Table 3.1. Siswati and isiZulu exhibited the highest token diversity with scores of 31% and 12%, respectively. On the other hand, the Sotho family languages showed relatively lower diversity, with scores of 3% and 5% for Sepedi and Setswana, respectively. The isiZulu corpus from OPUS-100 offered only a 6.5% diversity score, compared to 8.4% from IsiXhosa, resulting in an average diversity score of 7.45%. Although Flores200 had the least amount of text (15.5k sentences), it had the highest TTR score at 28.3%. We observed that Nguni languages tend to have higher TTR scores than Sotho languages. Due to time and computing constraints, we only used a limited amount of South African Bantu texts from the sources mentioned. However, we recognise the potential for improvement in this area and plan to explore it further in the future.

Symbols

Most Bantu languages spoken in South Africa use the same set of symbols consisting of five vowels and 26 alphabets. However, minor differences exist due to the presence of diacritics and accents. For instance, Tshivenda has additional symbols such as [ḓ, ḽ, ṋ, ṱ, ṅ] while Sepedi only has [à, š]. Upon analysing the raw datasets further, foreign symbols were found embedded within the text, including Chinese characters and emojis. This might indicate that the raw data was sorted using a Language Identification (LID) tool to separate the South African languages among a diverse collection of languages. After removing these foreign characters, Tshivenda had the most symbols, with Sepedi coming in second. The character set for other languages in the Sotho family was similar to Sepedi. In contrast, Nguni languages had a slightly different set of characters, including dashes which are commonly used because Nguni languages are conjunctive.

3.2.2 Evaluation corpora: Short-text classification

For extrinsic evaluation of the representation models, we collected news headlines in Sepedi and Tshivenda from various public sources, including local radio stations' Twitter and Facebook pages, namely *Phalaphala FM*, *Lesedi FM*, and *Thobela FM*. The data was acquired using Twint[1], a tool for extracting historical posts from Twitter, and Facebook-scraper[2], a Python package used to retrieve posts from public Facebook profiles. To augment the news corpus, we also incorporated data from the Vukuzenzele newspaper corpora [Marivate et al., 2023]. Additionally, we created synthetic English headlines using Open-AI text-generation service[3], which was subsequently translated to Sepedi. Unfortunately, this capability could not be used on Tshivenda due to the limited availability of reliable English to Tshivenda translators. Table 3.2 provides a summarised view of the news corpus. The documents represented here are in their raw form and will undergo preprocessing to eliminate short, malformed or irrelevant texts.

Source	**Language (ISO639-3)**	**Total articles**
Facebook	nso, ven	3403 / 12043
Twitter	nso, ven	221 / 23
Vukuzensele	nso, ven	883 / 842
Synthetic	nso	6837
Total		11344 / 12908

TABLE 3.2: Raw News corpus summary

3.3 Data annotation

This section will describe how we annotated news headlines in Sepedi and Tshivenda. These headlines were used as a benchmark for comparing models trained through cross-lingual transfer and traditional monolingual learning methods. Additionally, we will detail the criteria and verification steps we used to ensure the accuracy of the annotation results.

3.3.1 News Categories

Instead of synthesising custom news categories, we utilised the International Press Telecommunications Council (IPTC)[1] Media Topic News Codes, which were most recently updated on March 31, 2023. We believe using a standard set of categories enables future work to draw plausible comparisons on this work. The standard specifies 17 topics ranging from politics, crime, sports and economy to lifestyle and leisure. The complete list of all the topics and their respective descriptions can be found on the IPTC website[2].

3.3.2 Annotation platforms

The datasets were first manually annotated using LabelStudio [Tkachenko et al., 2020-2022], an open-source multi-modal data annotation platform. After a significant number of examples

[1] https://www.iptc.org/std/NewsCodes/treeview/mediatopic/mediatopic-en-GB.html

(1000 articles) were annotated, ML-assisted learning was enabled to support automatic labelling. Initially, the model only achieved 61% performance from 1000 samples spanning approximately four topics. At first, only 5/17 categories had enough samples to train a logistic regression model to support auto-labelling. Over time, the performance improved significantly, allowing more annotations to be completed quickly. For example, the initial model was very good at identifying *Health*, *Crime*, *Politics*, *Education* and *disaster, accident and emergency incident* headlines as they were prevalent on the corpus. Since the model was retrained after every batch of annotations, the annotations were then ordered according to the probability score so that low-scoring headlines got a higher chance of being seen at least once by a human annotator. This process was repeated until the model could identify most genres with a confidence score of at least *0.6*.

In theory, one annotator could create ground truth labels for a dataset. However, relying on a single annotator is not recommended because human judgement can be subjective and prone to errors. The reliability and accuracy of the ground truth labels can be improved by using multiple annotators to annotate the same dataset and then resolving any discrepancies through adjudication. Using multiple annotators and adjudication can help identify and correct errors, reduce bias, and increase the consistency and reliability of the annotations. It is a common practice in NLP to use multiple annotators and establish Inter-Annotator Agreement (IAA) score to measure the level of agreement among the annotators. IAA measures can be used to assess the quality of the annotations and the need for further refinement or clarification of the annotation guidelines [Artstein, 2017].

In addition to internally annotated datasets, some of the annotation work was outsourced to external annotators using Doccano [Nakayama et al., 2018]. Similar to LabelStudio, Doccano is an open-source multi-modal annotation platform. A group of volunteers were also recruited to speed up the annotation process. A guideline was provided to the annotators, and examples showed how to deal with ambiguities. Moreover, a link to the IPTC news codes was shared with the annotators for reference. There were two groups of annotators made up of native Tshivenda speakers and Sepedi speakers. The quality of the annotations was appraised using a combination of IAA scores and manual review with a special focus placed on ambiguous news genres.

3.3.3 Challenges

Because the data was collected during the period of Covid-19, we noticed many health-related articles. Sometimes, assigning a single label to a headline was not straightforward. For example, articles about Covid-19-related corruption could fall under health, business or politics. As a result, we decided to treat the annotation process as a multi-label task, although the final models focused on multi-class classification. The criteria used to pick a single label for a headline was the frequency of observations for each topic. For example, if an article could fall under crime or politics, and the number of observations for these categories were 10 and 5, respectively, we assign politics as the label. This was done to minimise the imbalance among the news categories.

Some headlines did not have apparent genres that matched any IPTC topics, for example, headlines related to traffic updates. As a result, an assumption was made that this kind of headlines should fall under the "society" genre, assuming that heavy traffic is an indirect indicator of the problem of overpopulation in cities, which is a major social problem in many countries. Political news headlines were also often difficult to assign to a single label as most reports talked about misconduct or criminal cases against government officials or state-owned entities, for example, the news about the Zondo Commission[3]. As a result, there was a lot of coincidence between this topic and "Crime, Law and Justice" news. Similarly, most news relating to "religion and belief" was about criminal charges against religious leaders.

Notably, we also had a few observations that could not be classified if one did not have context. For example, "Tshiimo tsha Alexandra tsho bva nnda ha tshanda" means "The situation at Alexandra is out of hand". For all we know, this could have been about protests, natural disasters, or an ongoing social problem. As a result, annotators were advised to mark articles like these as "not-applicable" so they could be excluded from the training datasets.

3.3.4 Annotation results

The quality of the annotations was verified through the use of pairwise inter-annotator agreement scores. For Sepedi, the *5* external annotators completed 1.5k annotations with an average pairwise agreement score of *0.427*. Although commendable, a score of 0.427 indicates a moderate agreement, which may have been caused by unclear labelling instructions or the ambiguity on some closely related news categories. We will leave the investigation and possible improvement of this score for future work.

To improve the diversity and class balance of the datasets, we applied data augmentation using a combination of back-translation, the generative capability of Open AI's GPT-3[4] models and zero-shot classification. For instance, given a randomly picked headline in the human-annotated dataset, we translated it to English using Google Translate[5] then sent an API request to get a GPT-3 model to generate ten unique articles on the selected topic while using the selected article as a hint. With this approach, we obtained an additional 6837 headlines for the Sepedi Corpus. We used two methods to verify these machine-generated annotations; firstly, we conducted k-fold cross-validation to select a suitable classifier and ensure that the models trained on this data performed well on the human-annotated dataset. Secondly, we assessed the inter-annotator agreement scores between labels generated by GPT-3 models and those obtained from zero-shot classification results using pre-trained Multi-Genre Natural Language Inference (MNLI) models, including BART (Tang et al., 2020) and DistilBart[6].

For Tshivenda annotations, we obtained an average pairwise agreement score of 0.37 from 3 external annotators. Unfortunately, only 1000 articles were completed, and there was no option to apply the augmentation technique applied for Sepedi since Tshivenda to English machine translation services are not yet readily available. As a result, we had to manually annotate a substantial number of Tshivenda articles without the ability to get the inter-annotator agreement score. Fortunately, the researcher is a Tshivenda-speaking individual; therefore, it was possible

[3] https://www.statecapture.org.za/
[4] https://platform.openai.com/
[5] https://translate.google.com/
[6] https://huggingface.co/valhalla/distilbart-mnli-12-1

to perform rigorous manual verification of the annotations. Some strategies applied to achieve the desired annotation quality involved re-labelling topics that appeared to perform worse on the initial experiments and using keyword filtering to verify that correct labels were assigned as expected. However, we acknowledge that there may be some weaknesses in our findings due to the potential mislabeling of entries. This is an area that we wish to improve on in future works.

3.3.5 Analysis

Despite being in different languages, we observed that the news articles in Tshivenda and Sepedi had comparable content. This is probably because the radio stations that provided the data are overseen by the South African Broadcasting Commission (SABC), which explains why they have access to the same news sources. Out of the 17 topics considered, we note the following prominent topics:

- Crime, law and justice - Unfortunately, this was the most dominant topic across Tshivenda and Sepedi news. It was closely followed by news relating to *Politics*. As mentioned before, it was often difficult to distinguish between crime and politics. This could be because the time span from which the data was collected was not diverse enough, causing our dataset to be biased toward popular issues at that time. Perhaps if we collected data across multiple years, we could have gotten more diversity.
- Society - According to the IPTC[7] specification this genre covers social and human rights issues which are commonly encountered in communities. We often encountered ambiguities for articles that revolved around protests for public service delivery as these could also plausibly fall under “Conflict, War and Peace”. There were also concerning issues around racism, human rights violations and poverty. The issue of poverty often went hand in hand with unemployment which raised ambiguity between the “Society” and ”Labour” genres.
- Health - As noted before, the collected headlines had many instances of news relating to Covid-19 infection rates and vaccination. In addition to ambiguity introduced by Covid-19 related financial crimes, there was often ambiguity with “Science, and Technology” regarding scientific research aimed at developing effective treatment against Covid-19. The travel bans introduced to prevent the spread of Covid-19 also presented significant challenges to annotators since these could be viewed as “disaster, accident and emergency incident”, considering that the disease was declared as a pandemic. Yet, it is also possible to argue that this was a social issue as it involved issuing grants to support those who could not go to work to support their families.
- *Economy, Business and Finance* - Most headlines under this topic talked about failing State-Owned Entities (SOEs), Black Economic Empowerment and Corruption. Corruption news often coincided with the topic of “Crime, Law and Justice” since they often involved theft, bribery and other forms of financial crime. A few instances also referred to government initiatives to combat unemployment and help the economy overcome the

[7] https://www.iptc.org/std/NewsCodes/treeview/mediatopic/mediatopic-en-GB.html

negative impacts caused by Covid-19. As a result, there was sometimes ambiguity between “Labour” and “Health” news.

- *Education* - Education headlines were mostly about the effect of Covid on school programs. Hence we may have mislabelled between education and health. Furthermore, there are several articles about protests related to the lack of learning infrastructure. Multiple cases of arson in rural schools were also encountered, in addition to sexual harassment activities by teachers against female students. This introduced ambiguity between the “education” genre and “conflict, war and peace” and “crime, law and justice”
- *Disaster, accident and emergency incident* - A high number of articles in this genre were about road accidents and floods. There were also a few notable incidents of students drowning which coincided with “Education” headlines.
- *Human Interest* - These headlines often talked about reports of famous people, or any news that affect human emotions such as music legends or political veterans passing away. There were also incidents where a memorial was held for victims of disasters or accidents, which caused ambiguities with the “Disaster, accident and emergency incident” category. Furthermore, it was at times difficult to distinguish whether a headline about a celebrity belonged to the genre of “Arts, Culture, Entertainment and Media” or “Human interests” as most celebrities in the entertainment industry are also famous people.
- *Other topics* - There were very few instances about *Sports*, *Science and Technology*, *Environment* or *Lifestyle and Leisure.* This was unsurprising as specialised topics such as these are often reported exclusively from dedicated channels which we did not utilise as sources in this study.

Cognisant of the observations above, we noticed that for most cases, assigning a single genre to a headline was not enough; hence most articles were assigned multiple labels during the annotation stage. We created a simple algorithm to generate a single-label dataset from the results. If multiple annotators assigned multiple labels to a headline, we chose the most occurring topic. If only one annotator provided a label, we chose the class with the least overall frequency to maintain class balance. We also attempted to merge categories with high cosine similarity scores to ensure all topics had a significant number of training examples.

3.4 Data Analysis

This section will present the exploration steps undertaken to get insights into the various descriptive statistics and topic distributions of the datasets used for language modelling and news topic classification. We aim to understand the relationships between these two data sets to anticipate how this might impact downstream performance.

3.4.1 Raw corpus

The raw datasets comprised unlabelled data in 9 Bantu languages spoken in the Republic of South Africa(RSA). Initially, each language was analysed individually, followed by a cumulative analysis of the full dataset. Our analysis aimed to address several key questions, including the identification of the most frequent words, the determination of the average token length per document, the identification of common entities, and the identification of prominent topics within the dataset. This analysis will provide valuable insights into the linguistic similarities among the languages under investigation, as well as the domains represented in the collected datasets. These insights will enhance our understanding of the results obtained in the downstream news topic classification task.

3.4.1.1 Sepedi

This dataset is made up of a diverse collection of raw texts obtained from various sources, including SADiLAR [Eiselen and Puttkammer, 2014], CC-100 [Wenzek et al., 2020], Flores [Team et al., 2022; Goyal et al., 2021; Guzmán et al., 2019], as detailed in Section 3.2. In total, we gathered more than 179k Sepedii sentences, totalling 24MB in size, as illustrated in Table 3.3 below.

Total documents	Unique tokens	Size in MB
179, 567	744	24.19

Table 3.3: Sepedi Raw Corpus Size

As seen in figure 3.1, each sentence contains about *15* to *40 tokens*, ranging from *70* to *200 characters*. The original text is in UTF-8 format to support accents, containing roughly 744 unique characters. We notice a significant number of outliers with characters over 1000. This might cause issues for the models since most transformer model architectures only support 512 or 1024 characters [Devlin et al., 2018] [Conneau et al., 2020]. To resolve this, we split each outlier sentence into chunks of 1024 smaller sentences.

Popular words

We utilised the Scikit-learn [Pedregosa et al., 2011] TF-IDF vectoriser to extract the most significant tokens from the corpus, employing varying n-gram sizes. Initially, our tests were heavily influenced by stop-words such as “go” (to), “le” (with) and “ka”(by). However, after removing these stop-words using the dynamically generated list, as described in 3.5.1, we obtained more informative phrases that revealed the dominant genres of the data displayed in Figure 3.2.

A brief inspection of the popular uni-grams word cloud depicted in Figure 3.2 reveals a substantial presence of texts originating from the domains of religion, politics, and justice. The prominence of religious texts is likely attributed to the inclusion of data from religious books, such as the Bible. For example, we notice a high occurrence of words like “jehofa” (jehova), “godimo” (heaven) and “modimo” (God).

Entities

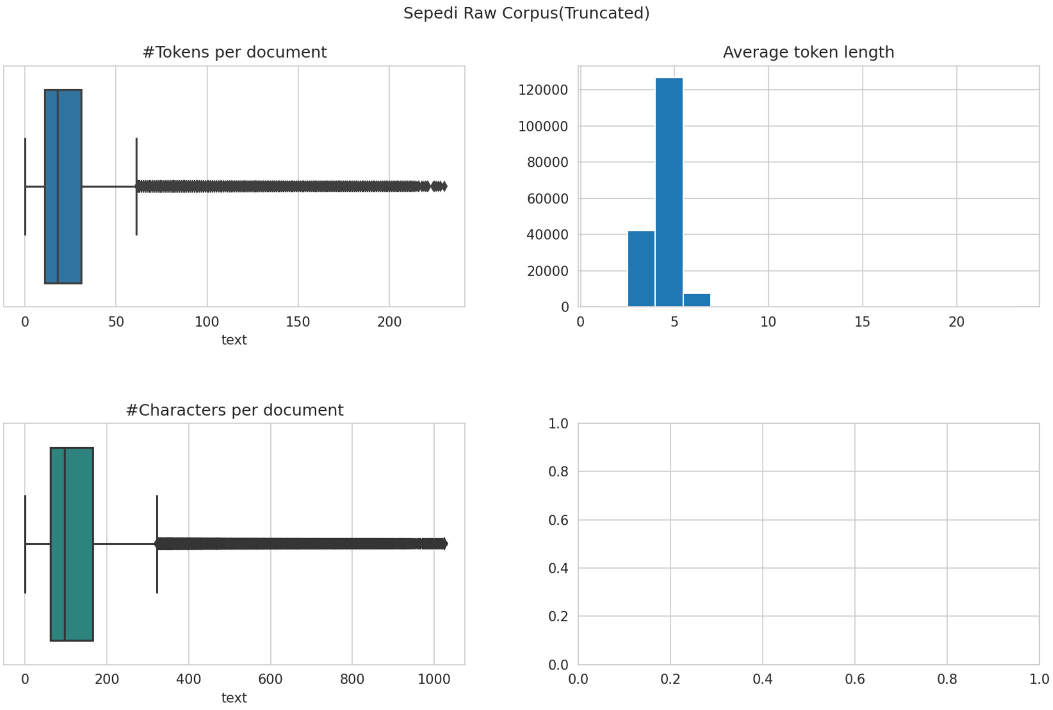


FIGURE 3.1: Sepedi Raw Corpus Size Distributions

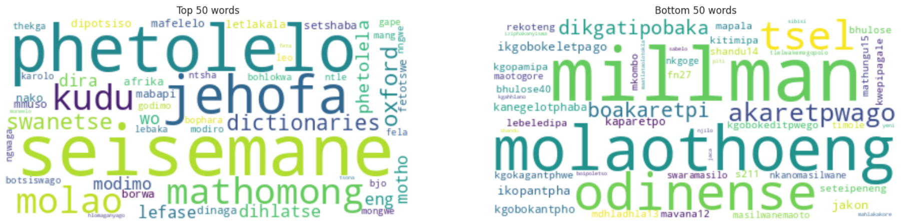


FIGURE 3.2: Sepedi Raw Corpus Popular Uni-grams

The results obtained using an off-the-shelf entity recognition (ER) model developed by Spacy[8] indicated that "Person" (PER) entities are the most common, followed by "Location" (LOC) and "Organisation" (ORG) entities. Figure 3.3 displays the top examples from each of these entity classes. The identified entities for "Location" and "Organisation" are convincing, while "Person" entities appear to be mostly random.

Topics

Upon reviewing the Latent Dirichlet Allocation (LDA) topic modelling results in Table 3.4 again, it becomes apparent that there are several religious tokens present, including "jesu" (Jesus) and "modimo" (God). Additionally, there appear to be dictionary entries and a few legal terms which may have originated from government documents.

[8] https://github.com/explosion/spaCy

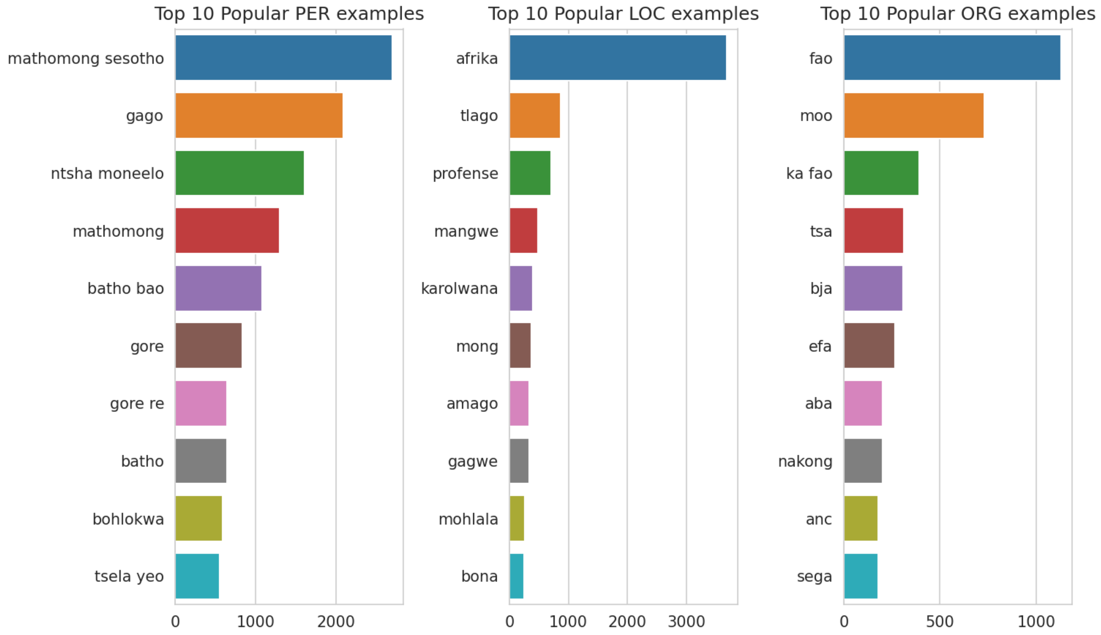


FIGURE 3.3: Sepedi Raw Corpus Popular Entities

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
modimo	kudu	ntle	mathomong	seisemane
motho	bjo	letsatsi	mathomong_seisemane	phetolelo
dira	tsona	bakeng	ngwaga	oxford
jesu	bohlokwa	feela	kapa	dictionaries
ng	feta	hao	morena	oxford_dictionaries
jehofa	dingwe	hau	fihla	phetolela_seisemane
mang	nago	latela	thoma	phetolela
baka	godimo	sebaka	tee	molao
nako	bontsha	fumana	tloga	swanetse
modiro	swana	tattoo	matsatsi	wo

TABLE 3.4: Sepedi Raw corpus popular topic terms. (The English translations for these terms can be found in Appendix C)

3.4.1.2 Tshivenda

Compared to Sepedi, the Tshivenda raw corpus is significantly smaller, containing only 73k sentences, while Sepedi has 179k sentences. We assume that this is because the Venda population is relatively smaller and that Venda people tend to speak less Tshivenda when they migrate to the city for work or tertiary studies, unlike other tribes from Sotho and Nguni families. Another contributing factor could be that we missed some sources more Tshivenda data. However, If we have missed such sources, it suggests that they are less readily available, thereby supporting our initial assumption. Moreover, Sepedi benefits from greater language tool support provided by major corporations like Microsoft and Google, which may have facilitated the collection and processing of Sepedi language data. The size distribution of the collected texts is illustrated in Table 3.5 and Figure 3.4 below.

We observed that the size of the Tshivenda corpus in MB is approximately half the size of

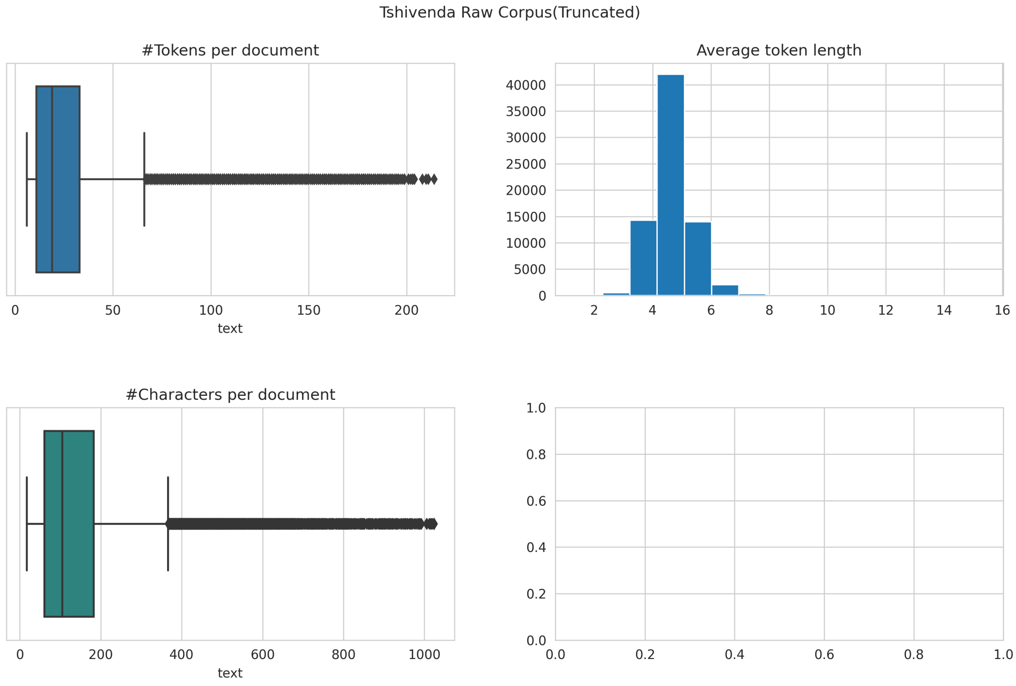


Figure 3.4: Tshivenda Raw Corpus Size Distributions

Total documents	Unique tokens	Size in MB
73, 336	160	10.43

Table 3.5: Tshivenda Raw Corpus Size

the Sepedi corpus, which aligns with our hypothesis that Sepedi has more available resources than Tshivenda. Additionally, we noted that the Tshivenda corpus had only 160 unique tokens compared to 744 in Sepedi. Furthermore, the number of sentences in Tshivenda made up only 42% of the sentences in the Sepedi corpus. The tokens in Tshivenda are also encoded in UTF-8 to support the accents identified in 3.2.1.

Popular words

Like the Sepedi corpus, we identified the need to remove stop words such as “vha” (they), “inwi” (you), “kha” (on) from the Tshivenda corpus before using TF-IDF to extract the highest scoring n-grams to get keywords from each document. The results of this analysis are presented in Figure 3.5 below. Unlike in Sepedi, the Tshivenda corpus appeared to be dominated by political and social issues texts with little to no appearance of religious texts.

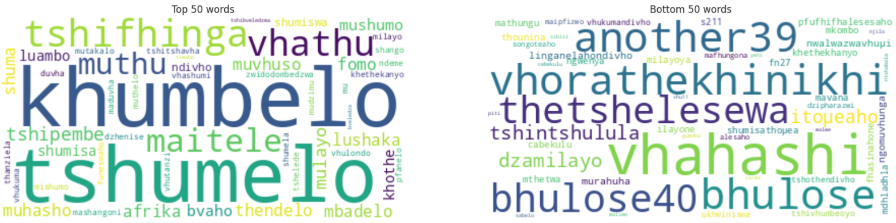


Figure 3.5: Tshivenda Raw Corpus Popular Uni-grams

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
vhathu	muthu	vhathu	khothe	mulayo
muvhuso	tshifhinga	duvha	mulandu	khethekanyo
lushaka	khumbelo	nwana	mveledziso	tshirema
ndeme	tshumelo	dzhena	thaidzo	uyu
mushumo	afrika	bvaho	tsireledzo	komiti
shuma	tshipembe	fhedza	thodisiso	mulayotewa
shumisa	afrika_tshipembe	mbo	maduvha	muhulwane
ndivho	masheleni	pfa	mbuelo	bvelela
shumiswa	thendelo	minwaha	mbudziso	sedzulusa
pfanelo	tshelede	tendelwa	fha	tshiwe

TABLE 3.6: Tshivenda Raw corpus popular topic terms (The English translations for these terms can be found in Appendix C)

Entities

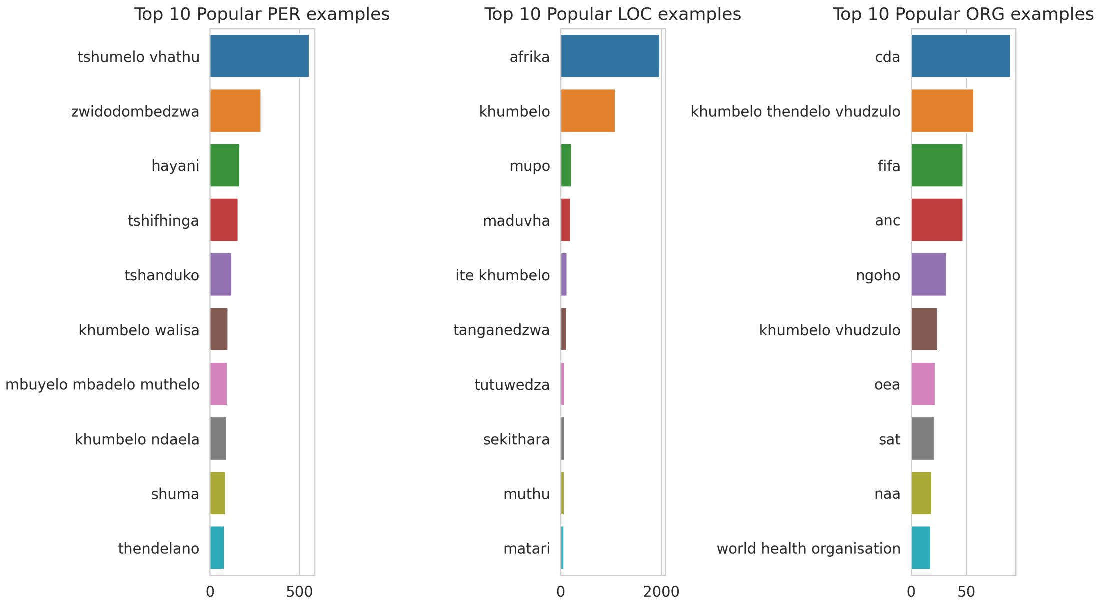


FIGURE 3.6: Tshivenda Raw Corpus Popular Entities

Looking at the entities shown in Figure 3.6, we once again noted the limitations of the pre-trained Spacy[9] model in accurately identifying "Person" entities, as discovered first in Section 3.4.1.1. However, "Organisation" entities appear to be more accurately identified, with examples such as FIFA (Federation of International Football Association), IEC (Independent Electoral Commission - of South Africa), and ANC (African National Congress - A political party in South Africa) appearing in the results. These observations highlight the significance of this study, as capabilities such as entity recognition are crucial building blocks to developing more advanced multi-modal models like Chat-GPT. They also reveal the necessity of constructing specialised models to address the identified limitations of existing models in the South African context.

Topics

[9] https://github.com/explosion/spaCy

Upon reviewing the results of 1000 iterations of LDA topic modelling in Table 3.6, we observe that the top 5 topics primarily revolve around the theme of justice, as indicated by terms such as "khothe" (court), "mulayo" (law), "mulandu" (crime), "sedzulusa" (investigate), and others. Additionally, we notice terms related to government bills and service delivery, including "mulayotewa" (constitution), "pfanelo" (rights), "masheleni" (budget), and "tshumelo" (public service), among others. These findings align with the observations made from the identified entities as seen in Figure 3.5.

3.4.1.3 Other

In addition to working with Sepedi and Tshivenda, we required additional data to train multilingual language models based on all 9 South African Bantu languages. The total amount of data selected for this purpose was approximately 0.19GB, with Sepedi being the language with the largest size of 24.08MB. At the same time, Setswana had the largest number of documents at 340k. Despite Setswana having a larger number of documents, they are generally shorter than Sepedi, with an average of *10* tokens per document compared to *25* in Sepedi. Table 3.7 presents a summarised overview of the full corpus. This dataset will play a pivotal role in addressing one of our secondary research inquiries, which aims to explore the potential of injecting additional training data from closely related languages to achieve enhanced zero-shot performance between Tshivenda and Sepedi.

Lang(ISO 639-3)	#Documents	Size(MB)
nso	179,567	24.08
sot	37,375	2.75
tsn	340,313	20.94
ven	73,336	10.43
tso	34,356	2.48
nbl	30,188	3.25
xho	322,185	21.47
zul	51,250	3.39
ssw	43,113	4.35
Total	1,111,683	186.2

Table 3.7: South African Bantu Corpus Summary

3.4.2 News corpus

The news corpus is a collection of human-annotated and machine-annotated headlines for Sepedi and Tshivenda. In this section, each collection will be explored separately to see if there is consistency between the annotations.

3.4.2.1 Sepedi - Human annotations

This subset of the news corpus was annotated by humans through an adjudication process. It serves as a crucial validation set for the rest of the news corpus, developed using machine-assisted

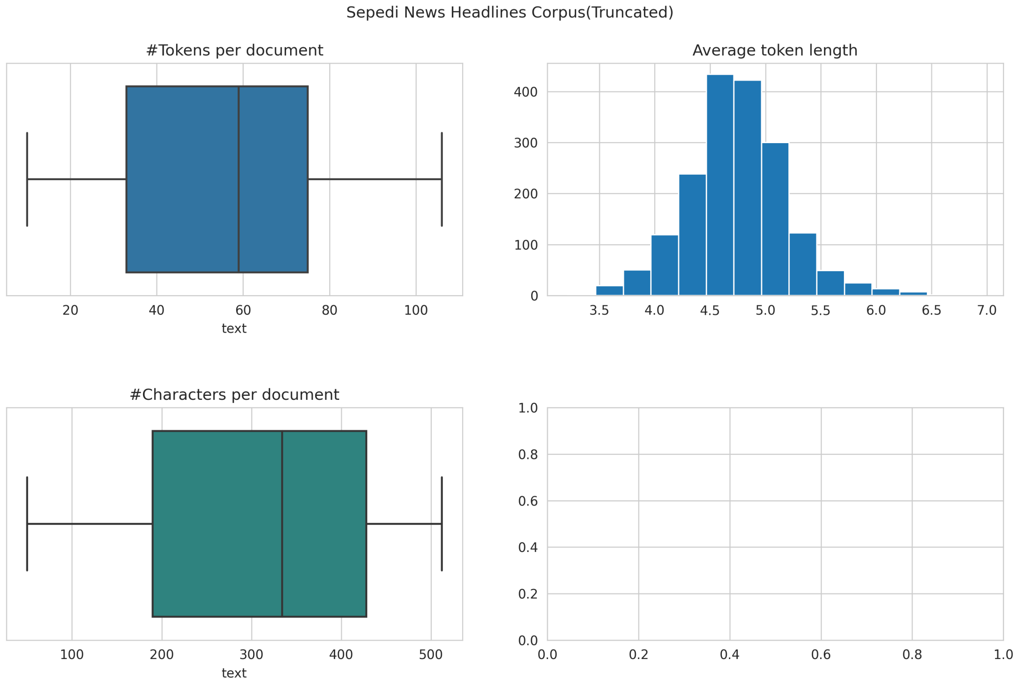


FIGURE 3.7: Sepedi Human Annotated News Corpus

labelling. The corpus comprises 1.8k sentences, which add up to 0.56MB. The average token length as shown in Figure 3.7 is consistent with the observations we made in the raw dataset from Section 3.4.1, with a range between 4 and 6. To ensure the model is not biased towards overly long or short sentences, each headline is truncated to 512 characters, creating a more balanced distribution of sentence lengths.

Popular words

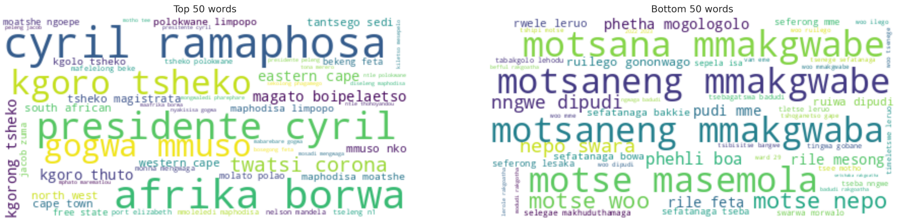


FIGURE 3.8: Sepedi Human Annotated News Corpus - Popular Bi-grams

Entities

Looking at Figure 3.9, we notice that Spacy's multilingual entity recognition (ER) model accurately identifies entities in all three categories we considered. For instance, in the Person

(PER) category, we can see well-known figures in South Africa like “Jacob Zuma” and “Cyril Ramaphosa,” former and current presidents of the republic. These individuals are also prominent in the word cloud of popular bi-grams as shown in Figure 3.8. The Identified Organisations (ORG) category includes South African Airways, a national airline, as well as the DA and ANC, which are prominent political parties in SA. Finally, the identified Location (LOC) entities comprise Johannesburg, Nigeria, Cape Town, and others.

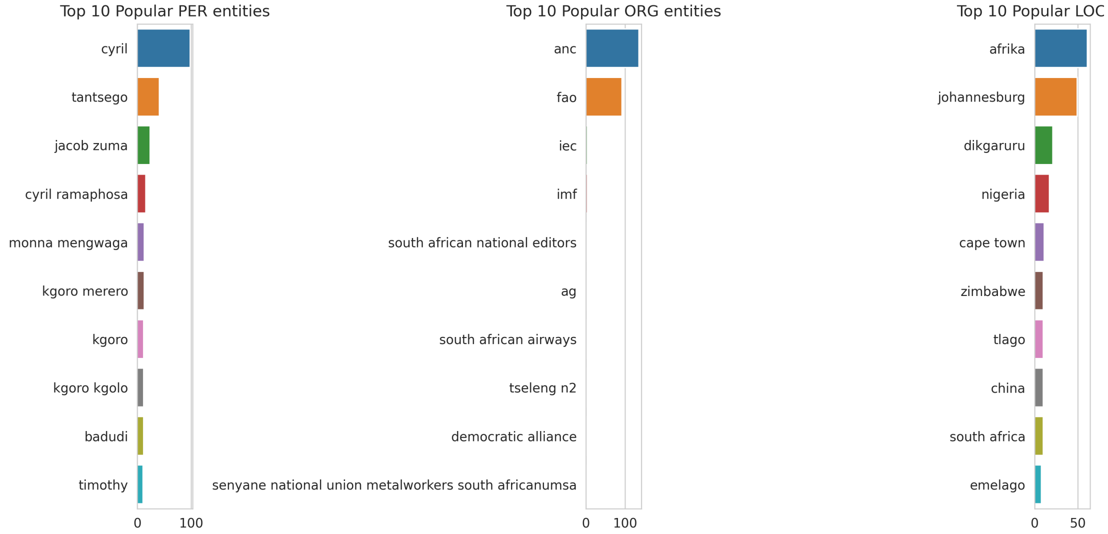


FIGURE 3.9: Sepedi Human Annotated News Corpus - *Spacy* entity examples

Topics

The top 5 topics identified using LDA, as depicted in Table 3.8, show that the corpus may be dominated by the ”Crime, Law, and Justice” news category, evidenced by terms like ”tsheko” (trial), ”maphodisa” (police), and ”molato” (crime or case). We also observe a significant number of unigrams and bi-grams related to education such as “sekolo” (school), “thuto” (education) and “kgoro_thuto” (department of education).

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
limpopo	kgoro	maphodisa	afrika	sekolo
badudi	tsheko	limpopo	afrika_borwa	thuto
magato	mengwaga	tikologong	borwa	south
kgoro	limpopo	ntle	anc	sekolong
boipelaetso	feta	mengwaga	mmuso	barutwana
tikologong	kgorong	monna	tona	african
magato_boipelaetso	wo	bagononelwa	ditshelete	kgoro_thuto
ntle	magistrata	bana	lekala	kgoro
mmasepala	kgoro_tsheko	ngoepe	ditsela	south_african
bya	kgorong_tsheko	fao	merero	phagamego
tikologo	molato	mosadi	nageng	morutwana
barutwana	lapa	moatshe	maloko	limpopo

TABLE 3.8: Sepedi Human Annotated News Corpus popular topic terms (The English translations for these terms can be found in Appendix C)

3.4.2.2 Sepedi - Machine annotations

This dataset contains a subset remaining after removing human-annotated examples. It also contains augmented articles generated using the Open-AI[10] text completion service. As seen in Figure 3.10, the label balance is much better than in the human-annotated subset although we still notice that the top 5 topics dominate with over 1k articles each while the remaining 12 topics have 500 topics or less.

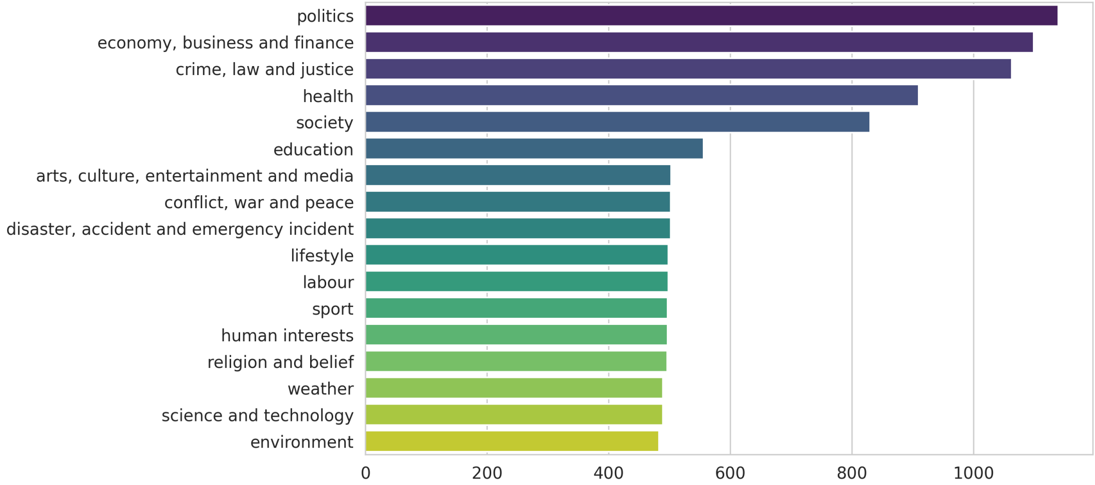


FIGURE 3.10: Sepedi Machine Annotated News Corpus - Label Distribution

Popular words

We observe that the prominent topics in this subset(Figure 3.11) are still dominated by government and crime news, as indicated by terms such as "mmuso" (government), "Cyril Ramaphosa" (South African President), "nyakisiso" (investigation), and so on. This aligns with the previous observation made for the human-annotated subset, as depicted in Figure 3.8. We also see a number of headlines about Covid-19, employment and technology.

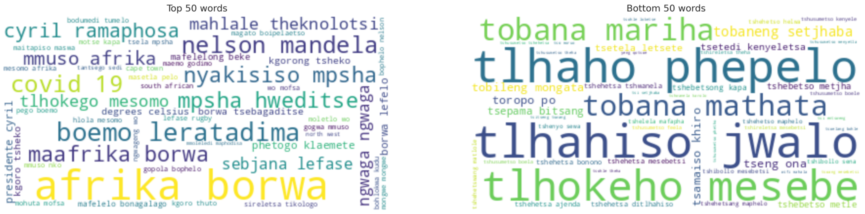


FIGURE 3.11: Sepedi Machine Annotated News Corpus - Popular Bi-grams

Entities

[10] https://platform.openai.com/

The top entities in Figure 3.12 align with those identified in the human-annotated corpus, as shown in Figure 3.9. This result is expected since both datasets were collected from the same sources. Similary, the top 5 topics in Table 3.9 appear to be dominated by issues of public service delivery and crime. We also observe a number of terms about employment and Covid-19 which is likely related to job-cuts that were experienced globally due to the pandemic.

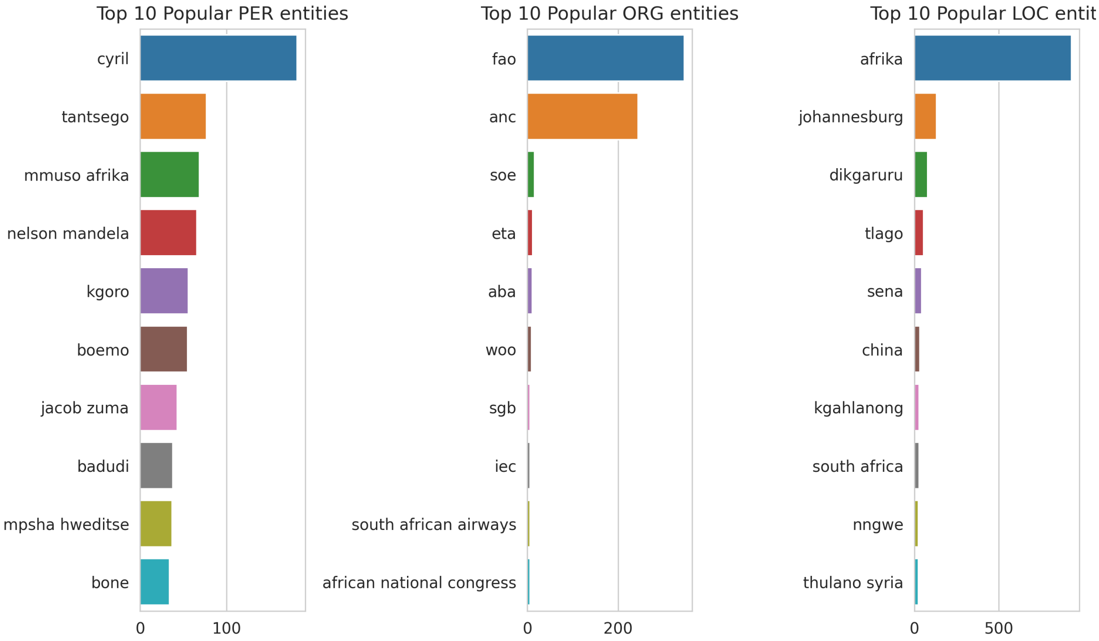


FIGURE 3.12: Sepedi Machine Annotated News Corpus - *Spacy* entity examples

Topics

Topic relationships

We use the average document vectors of articles to analyse the relationship between different topics. This allows us to identify which topics may be incorrectly classified. By doing so, we can merge them for improved performance. For instance, figure 3.13 illustrates a high cosine similarity score between crime and politics, which is unsurprising. However, we also noticed a high score between politics and "economy, business, and finance". This could be due to politically connected companies being involved in Covid-19 tender scandals. Moreover, we found a significant correlation between health and society, which may be attributed to challenges experience by communities during lock-down to prevent the spread of the Covid-19 virus.

3.4.2.3 Tshivenda - Human annotations

Popular words

The top bi-grams depicted in Figure 3.14 suggest that politics and Covid-19 are the dominating categories in this dataset. For example, we see a high occurrence of terms like "Covid 19", "vhulwadze" (disease), "lihoro anc" (ANC political party) and "Cyril Ramaphosa" (South African president).

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
mesomo	limpopo	mmuso	leratadima	mpsha
kudu	maphodisa	basomi	bjo	bana
wo	badudi	maphelo	khutso	bodumedi
dira	ntle	magato	bantsi	dimilione
gape	lefelo	dikgwebo	tlago	feta
setshaba	tsheko	fokotsa	boemo_leratadima	lefaseng
swanetse	molato	theknolotsi	letetswe	ra
thusa	mengwaga	tikologo	dula	matla
bohlokwa	tikologong	melao	maatla	motho
soma	leo	tlhokego_mesomo	pego	neng
hwetsa	fao	sireletsa	mmalwa	boela
nako	pula	tsebagaditse	lefase	bophara
mosomo	polokwane	palo	kudu	dilo
tloga	kgauswi	sego	dithemperetsha	mentsi
leo	feta	mahlale	dutse	ditumelo
covid	bekeng	thusa	beke	thata
fela	polao	kimollo	mafelelong	kotsi
mongwe	mmoleledi	mekgatlo	nago	ditokelo
eupsa	monna	dikhamphani	kgolo	phela
bangwe	bego	boletse	bolwetsi	tumelo

Table 3.9: Sepedi Machine Annotated News Corpus popular topic terms (The English translations for these terms can be found in Appendix C)

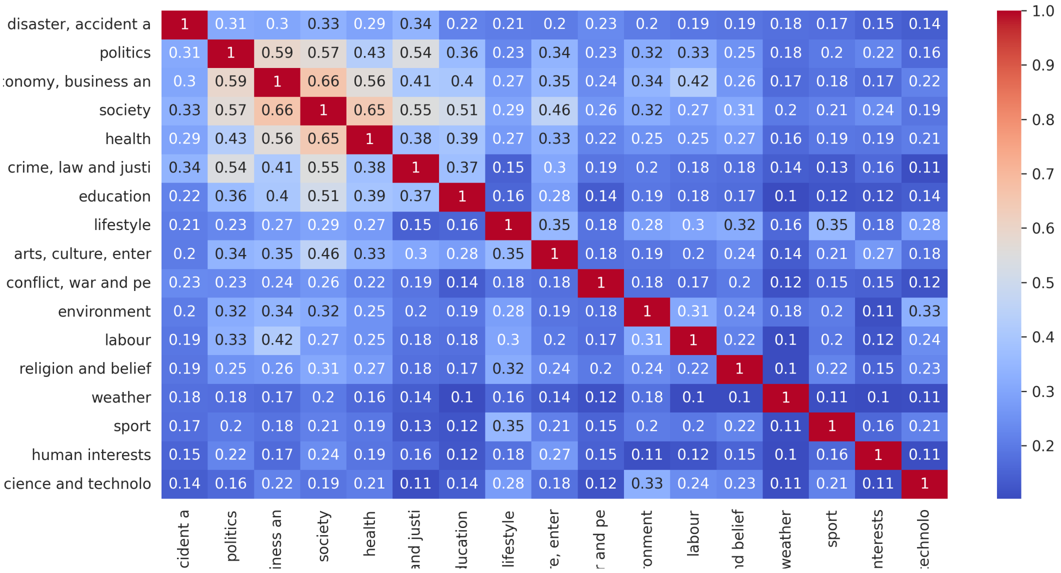


Figure 3.13: Sepedi Machine Annotated News Corpus - Label Similarity heat map

Entities

Surprisingly, we obtained plausible results in entity extraction using Spacy's *xx_ent_wiki_sm*[11] named entity recognition (NER) model, which is not specifically trained on any South African language. The results reveal that the most frequently extracted entities are "Person", followed by "Organisations" and "Locations", as depicted in Figure 3.15. Further analysis shows that most

[11] https://github.com/explosion/spacy-models/releases/tag/xx_ent_wiki_sm-3.5.0

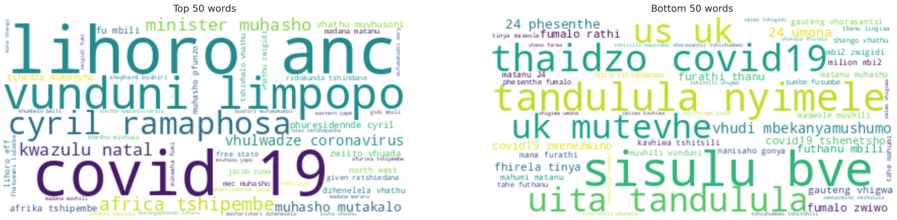


Figure 3.14: Tshivenda Human Annotated News Corpus - Popular Bi-grams

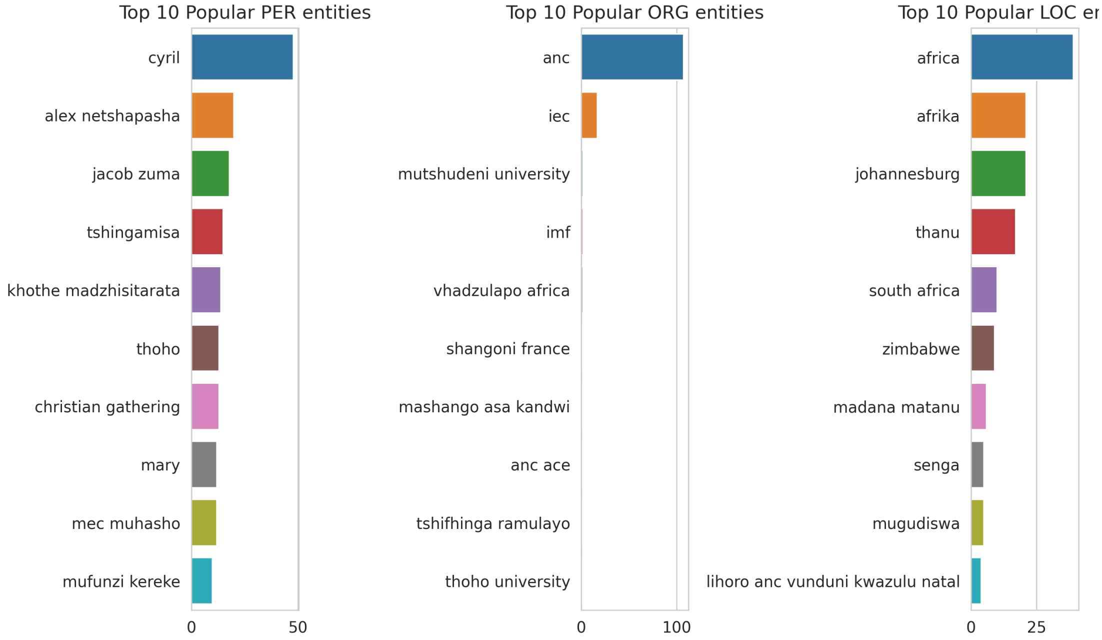


Figure 3.15: Tshivenda Human Annotated News Corpus - *Spacy* entity examples

of the identified individuals are political figures, and similarly, political parties are classified under 'Organisations', such as the ANC. However, some questionable results lead us to believe that the detected entities may only be prominent due to their international fame and frequent appearance in global news, meaning that they are likely to appear in high-resource languages used to train Spacy models.

Topics

We can readily recognise politics, criminal trials, and health-related welfare concerns by examining the top 5 LDA topics in Table 3.10. We also notice that this news corpus may be dominated by the news relating to the South African state capture enquiry procedure, which was active at the time this corpus was collected.

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
vhathu	khomishini	limpopo	vhathu	lihoro
fu	vhathu	vunduni	vhadzulapo	anc
madana	zuma	minwaha	tshifhinga	lihoro_anc
zwigidi	muhulwane	vunduni_limpopo	duvha	khetho
mbili	zwiito	lovha	tshipembe	vunduni
thanu	muvhusoni	mapholisa	pfala	eff
covid	vhutanzi	humbulelwa	mapholisa	masipala
tshipembe	dzhenelela	fumi	tshumelo	mirado
fumi	mavharivhari	khothe	zwavhudi	masipalani
africa	muvhuso	khombo	afrika	lihoro_eff
ina	dzhenelela_vhathu	rathi	tshimbila	mivhuso

TABLE 3.10: Tshivenda Human Annotated News Corpus popular topic terms (The English translations for these terms can be found in Appendix C)

3.4.2.4 Tshivenda - Machine-assisted annotations

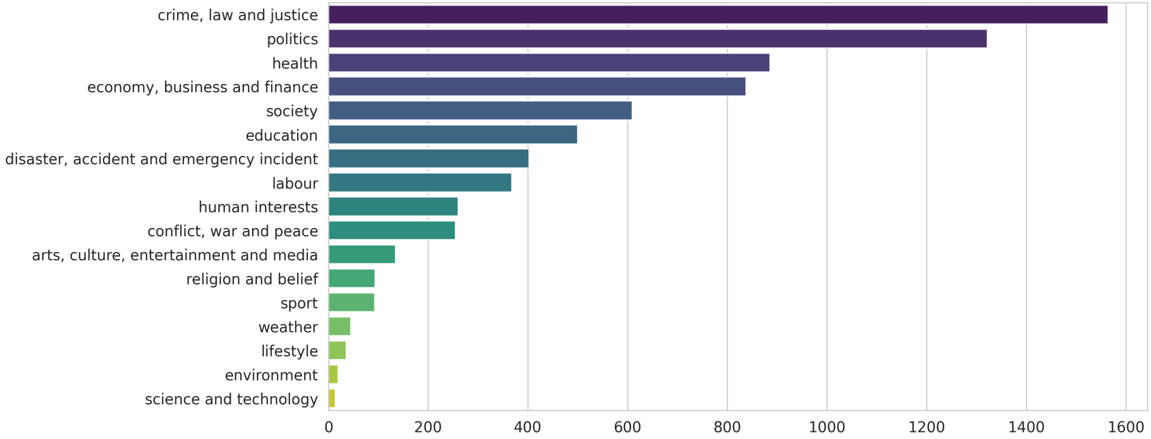


FIGURE 3.16: Tshivenda Machine Annotated News Corpus - Label Distribution

The class imbalance in Tshivenda news, as shown in Figure 3.16, is notably more pronounced. Unfortunately, the limited availability of suitable augmentation tools for Tshivenda, in contrast to Sepedi, hindered our ability to address this issue. For the time being, the only to remedy this is to collect more data for the minority topics. We leave this for future work.

Popular words

As expected the popular words for the machine-annotated subset of the Tshivenda news corpus in Figure 3.17 are similar to the ones observed in the human-annotated from Figure 3.14. In addition to politics and health related news we also observe a significant number of terms related to crime, which can also be observed from the top 5 LDA topics in Table 3.11.

Entities

This time we see worse performance than observed in Figure 3.15 where the pre-trained Spacy model was able to identify numerous locations and organisation entities. This is likely due to the lack of famous entities in this machine-annotated subset of data compared to the human-annotated split.

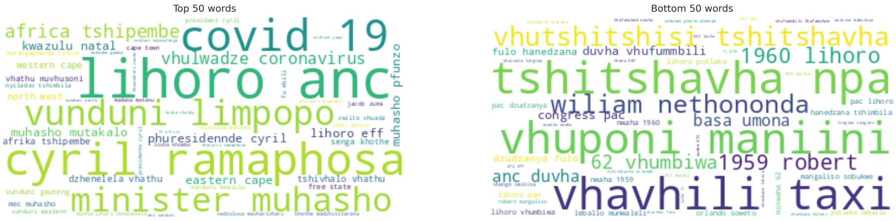


FIGURE 3.17: Tshivenda Machine Annotated News Corpus - Popular Bi-grams

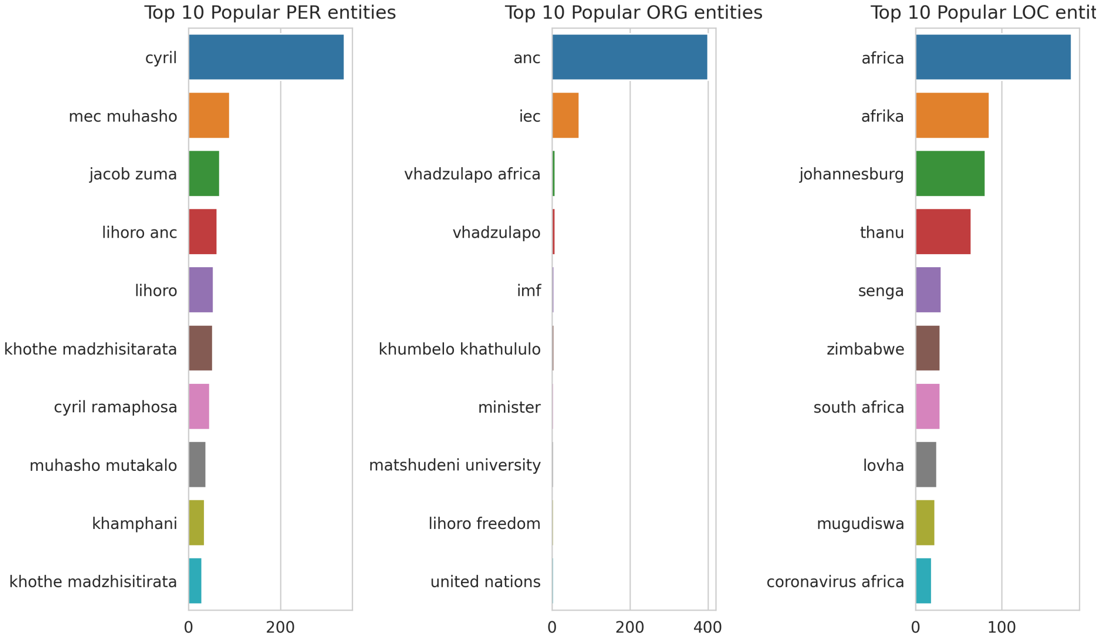


FIGURE 3.18: Tshivenda Machine Annotated News Corpus - *Spacy* entity examples

Topics

The prominent topics identified from LDA topic modelling in Tshivenda (Table 3.11) closely align with the topics identified in Sepedi news (Table 3.8). Notably, Covid-19 emerges as a central theme, influencing other dominant topics like "labour", "society", "economy, business and finance", and "politics". These findings suggest that transfer learning between these two languages is likely to be effective, given that the classification datasets originate from similar domains.

Topic relationships

In contrast to the findings presented in Figure 3.13, we observe a significant correlation between the categories of "society" and "conflict, war and peace" in Tshivenda news as seen on Figure

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
vhathu	khothe	ramaphosa	vunduni	vhashumi
lovha	mulandu	cyril	limpopo	masheleni
fu	mapholisa	tshipembe	mapholisa	dzangano
madana	vhulaha	shango	vunduni_limpopo	rannda
covid	vhahumbulelwa	phuresidennde	muhasho	khamphani
tshivhalo	milandu	coronavirus	gauteng	muvhuso
zwigidi	vunduni	vhulwadze	vhadzulapo	tshifhinga
tshipembe	munna	nyiledzo	natal	nwaha
mbili	senga	muvhuso	kwazulu	mbili
khombo	west	phuresidennde_cyril	vhathu	muhulwane
vhulwadze	north	afrika	vundu	tshelede
africa	muhumbulelwa	vhathu	mec	fhungudza
thanu	minwaha	vhadzulapo	vhuponi	fumi
coronavirus	humbulelwa	afrika_tshipembe	doroboni	muofisi
fumi	senga_khothe	africa	johannesburg	vhuada
rathi	farwa	covid	vhuendi	million

Table 3.11: Tshivenda Machine Annotated News Corpus popular topic terms (The English translations for these terms can be found in Appendix C)

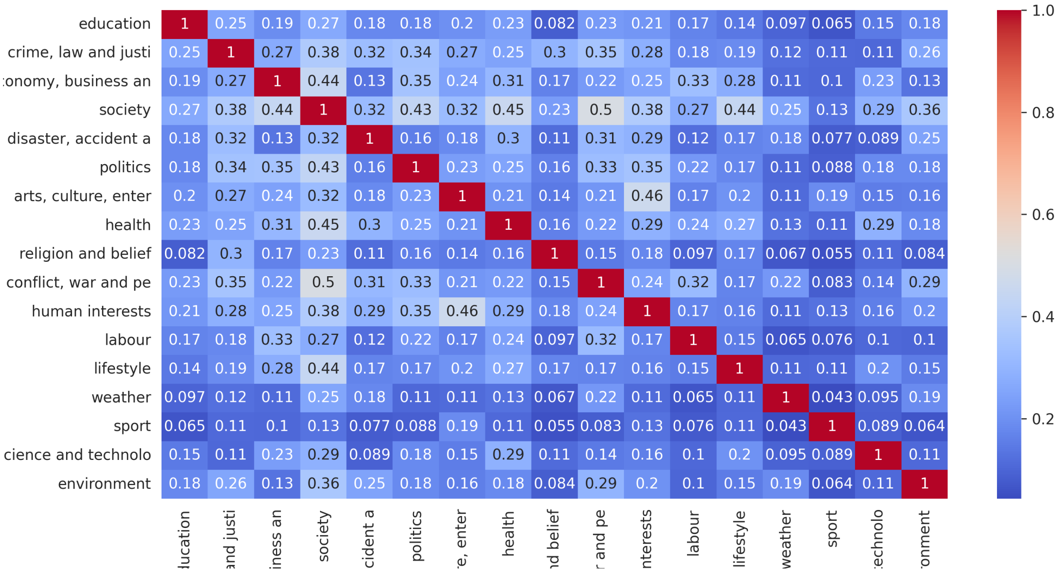


Figure 3.19: Tshivenda Machine Annotated News Corpus - Label Similarity heat map

3.19. This suggests that there is a higher occurrence of instances related to protests, potentially related to service delivery issues or other human rights concerns. Additionally, we identify a strong association between "human interests" and "arts, culture and entertainment" news. This association can be attributed to well-known individuals who also happen to be artists contributing to the entertainment sector. Notably, the similarity score between crime and politics is only 0.34 in this instance, which is lower than the 0.5 observed for Sepedi in Figure 3.13. Finally, we note the relationship between lifestyle and society, which can be attributed to travel restrictions during the period under analysis, leading to society-related news headlines due to the impact of lockdown measures.

3.5 Data Preparation

In this section, we provide an overview of the steps followed to prepare the two groups of datasets used to train the representation models and the downstream task models. We first describe the criteria for selecting the pre-processing steps applied to the raw data, including normalisation, text cleaning and tokenisation. Furthermore, we outline the transformation tasks performed on data collected from PDFs, Database snapshots and HTML pages. Finally, we highlight the challenges encountered when adapting existing text-processing tools to low-resource languages.

3.5.1 Pre-processing

This study followed a delayed pre-processing procedure to prepare our raw datasets for different training pipelines. Traditionally, most NLP datasets go through the same pre-processing steps before they can be used for training. However, in some cases, some of these steps may be optional. For example, it is not required to remove punctuations before training in extensive language modelling. However, when training a classic ML model like Logistic Regression, it is necessary to remove these.

Normalisation

Both Tshivenda and Sepedi texts contain diacritics that require UTF-8 text encoding. Although most NLP tools can handle UTF-8 text without difficulty, we discovered inconsistencies in the usage of diacritics across different data sources. To address this issue, we normalised all texts by removing diacritics and converting each letter to its corresponding ASCII representation using the Unidecode[12] library in Python. Furthermore, all the texts were converted to lowercase as there were inconsistencies in the use of cases. This normalisation process aimed to minimise the occurrence of out-of-vocabulary (OOV) errors by ensuring that the models do not treat the same tokens as different due to writing inconsistencies. In future research, it would be interesting to investigate how different models would perform if this normalisation step was eliminated.

Punctuation and Stop-word removal

As noted above, while punctuation removal is not mandatory in modern model architectures, it is a crucial step for traditional methods that depend on TF-IDF or Word2vec token representation. Similarly, stop-word removal is often used as an additional step to increase the signal-to-noise ratio in a document. However, attention-based representation-based methods can help to reduce the noise-to-signal ratio, which can make some pre-processing steps optional. This is particularly useful when dealing with very large sets of texts, as it can reduce unnecessary compute usage during pre-processing. Additionally, we observed a few texts written in English that could be filtered out using off-the-shelf Language Identification (LID) tools. However, this was not done in this study to avoid the risk of potentially losing information in our limited datasets.

Given the varying pre-processing requirements for different models used in this study, these steps were applied conditionally before training. For traditional machine learning models trained

[12] https://pypi.org/project/Unidecode/1.3.6/

using Sklearn, this was added as an extra transformation layer before vectorisation and estimator layers. Similarly, we applied stop-word and punctuation removal for deep learning-based models that used global word vectors before vectorising the tokens. These steps were not used for training new language models based on the XLMR architecture. Similarly, for fine-tuning the language models, we did not remove stop-words and punctuation; however, the text was normalised and lower-cased.

3.5.2 Lemmatisation and Stemming

Lemmatisation and stemming are beneficial in syntactic and semantic information retrieval tasks. Stemming is a rule-based technique that removes suffixes or stems from words. For example, the stem for the Venda word "tshigayoni" is "tshigayo". Similarly, lemmatisation aims to reduce a word to its atomic form based on its part of speech. Both techniques can be useful in minimising out-of-vocabulary occurrences in token representations by reducing the possible variations of a word. For example, using lemmatisation, we could treat the words "vhatambi" (players) and "mutambi" (player) as one word "mutambi" (player). The NLTK [Bird et al., 2009] library is a popular library used to apply lemmatisation and stemming for English texts. Unfortunately, this library does not currently support most South African languages; hence, these steps are not applied at this time.

3.5.3 Optical Character Recognition

The PDF datasets collected from government websites were converted into text using the PyPDF2[13] library, an open-source PDF text extraction library. Although this library worked well for most PDFs, it struggled with older scanned PDFs. In future work, we will compare these results with OCR results obtained using Tesseract[14], which uses LSTMs to detect characters from images instead of trying to extract the text from the source code of the PDF documents. After extraction, some of the text was misaligned, prompting an additional step to extract meaningful sentences that could be used for training. Heuristic techniques were used to obtain the sentences based on punctuation and other indicators that signal the beginning and end of sentences.

3.6 Tokenisation

Tokenisation is a term commonly used to refer to converting a piece of text into individual words. However, tokens could also refer to individual characters or parts of full words depending on the task. In this study, we consider three methods of tokenisation: Byte-Pair Encoding(BPE) [Gage, 1994; Sennrich et al., 2015], WordPiece [Schuster and Nakajima, 2012] and Word-level tokenisation. This was again influenced by the need to work with multiple models with different

[13] https://github.com/py-pdf/pypdf
[14] https://tesseract-ocr.github.io/tessdoc/

tokenisation requirements. For example, machine learning and deep learning methods traditionally require word-level tokenisation, while attention models like BERT require subword-level tokenisation, such as BPE.

Word level tokenisation

This method of tokenising texts, which involves obtaining tokens by splitting the sentence by white spaces, is probably the easiest. However, it does not work well for conjunctive languages like isiZulu or more complex writing symbols such as Chinese. Despite this limitation, we use this method to train our baseline models as it is the native way to train machine learning models using SKlearn [Pedregosa et al., 2011] and deep learning libraries like Tensorflow [Abadi et al., 2015] or Pytorch [Paszke et al., 2019].

Subword level tokenisation

Subword tokenisation effectively solves traditional NLP models' token coverage problem [Sennrich et al., 2015]. It enables the models to dynamically synthesise word vectors, even for words not present during training. This capability is crucial in low-resource settings. Two popular subword tokenisation methods are Byte-Pair Encoding (BPE) [Gage, 1994; Sennrich et al., 2015] and WordPiece tokenisation, particularly in transformer-based models.

The BPE algorithm induces a vocabulary by iteratively selecting the most frequent subwords on a predetermined set of characters. The resulting vocabulary typically consists of morphemes, which are then used to represent words from the original text in their most atomic form. However, the BPE approach has a disadvantage in that the resulting subwords may not necessarily be meaningful in the context of the given training corpus, which can lead to degraded performance [Bostrom and Durrett, 2020]. To address this, improved algorithms such as WordPiece [Schuster and Nakajima, 2012] create subwords based on the highest probability in the given training data, resulting in meaningful subwords for the context. It is important to mention that Wordpiece takes longer to train than BPE. Consequently, choosing between the two may result in a trade-off between accuracy and speed. On the other hand, Unigram tokenisation adopts a predetermined vocabulary that is constantly reduced until the desired number of tokens is achieved, as noted in the source [Kudo, 2018].

For this study, we used two methods of tokenisation: word-level and sub-word-level. We used whitespace splitting to tokenise data for TF-IDF and word2vec feature extraction and the built-in subword tokeniser for FastText modelling. To train our language model, we utilised BPE and Unigram tokens extracted using the SentencePiece library. We utilised SentencePiece instead of WordPiece, following the procedure outlined in the reference work [Ogueji et al., 2021]. In the future, we will analyse how WordPiece tokenisation works compared to BPE and Unigram methods.

3.6.1 Challenges

As highlighted d in previous sections, the current state of NLP tools suggests a notable gap in pre-processing low-resource language texts. One challenge is the difficulty in removing stop words from the training text for languages such as Sepedi and Tshivenda due to the lack of available stop word collections. In the interim, we used a combination of inverse term frequency, character sizes and word clouds to remove tokens which convey the least meaning.

Additionally, we encountered normalisation challenges, specifically with characters containing diacritics, which needed to be consistently recognised and introduced spelling errors. The lack of advanced spelling checks for Tshivenda and Sepedi compounded this issue. Unfortunately, this problem has no easy solution, and it remains an area for future research. Although there have been efforts to expand NLP tools to low-resource languages recently, such as the development of NLTK Africa wordnet [Bosch and Griesel, 2018], we have observed that there is still much work to be done to create quality model training workflows for South African languages.

3.7 Representation methods

This section provides an overview of the text representation techniques used in this study to extract semantic features from raw texts. We first discuss the conventional approaches based on TF-IDF, which provide a foundation for understanding the advancements made by more recent neural network-based techniques, such as word embeddings and contextual embeddings.

3.7.1 Term Frequency, Inverse Document Frequency(TF-IDF)

TF-IDF is a simple yet powerful technique used to create token embeddings. Its purpose is to determine which tokens in a text have the most significant impact based on their frequency in relation to the total number of documents in a corpus. Each token in the corpus is given a score that reflects its importance in conveying meaning in a sentence or document. Our research used TF-IDF feature extraction to train classic ML models using linear and tree-based algorithms. For specific tasks such as topic classification, this method typically works best with standardisation techniques such as punctuation removal, stemming, lemmatisation, and stop-word removal.

However, due to the lack of mature standardisation tools for Sepedi and Tshivenda, we only apply punctuation removal, accent removal, and stop-word removal using a synthetic list of words dynamically generated using IDF and token length heuristics. We utilised a pre-existing list of stop words specific to African languages available from Github[15]. Additionally, we include extra tokens that are not naturally stopwords but do not convey any useful information, such as “iring” (in this hour), “lehono” (today), “awara” (in this hour), and “ditaba” (news), which usually appear in the introduction of the actual news headline.

3.7.2 Word2Vec

Word2Vec is a popular method for producing dense, high-dimensional word embeddings that capture the semantic similarity between words. Compared to traditional techniques such as

[15] https://github.com/stopwords-iso

TF-IDF or Bag-of-words, which assigns a single weight to each token in a document, Word2Vec represents each token using a 1D vector with 100-500 dimensions, enabling mathematical computation of similarity between words. The vectors are obtained by training a neural network using a technique called Skip-Gram [Mikolov et al., 2013b; Meyer, 2016], which predicts the most likely neighbouring words given a target word. This allows the model to learn representations that capture the meaning of a word based on its surrounding context, resulting in more accurate and contextually rich embeddings. Another less-used method for training Word2Vec embeddings is Continuous Bag-Of-Words (CBOW) [Meyer, 2016]. Unlike Skip-Gram, which predicts surrounding words given a centre word, CBOW predicts the centre word given its surrounding context words. CBOW is generally faster to train than Skip-Gram and works well for smaller datasets or when the focus is on frequent words. However, Skip-Gram tends to perform better in capturing the semantics of less frequent words [Mikolov et al., 2013b].

To induce the Word2Vec embeddings, we use two distinct training strategies. In the first approach, we implement an unsupervised training pipeline, which generates the embeddings from the raw texts. These embeddings are subsequently utilised for training downstream task models by freezing the embedding layers in the LSTM or CNN networks. This strategy ensures that the embedding weights remain fixed during the training process of downstream models, thereby allowing for better optimisation of the subsequent layers. In the second approach, we employ an alternative methodology where the labelled dataset is used to concurrently learn the embeddings during the training on a downstream task. This approach facilitates the production of more optimised embeddings for the specific task at hand, potentially resulting in improved performance over the separately trained embeddings.

Finally, we experiment with different hyperparameters, such as embedding dimensions, window sizes, and negative sampling, to evaluate their impact on the performance of downstream tasks. Furthermore, we compare the performance of our Word2vec embeddings with those obtained from other pre-trained models such as FastText [Bojanowski et al., 2016]. Finally, we analyse the learned embeddings using visualisation techniques such as t-distributed Stochastic Neighbour Embedding (t-SNE) and Principal Component Analysis (PCA) to gain insights into the relationships between words in the semantic space.

3.7.3 Masked language Models

Although Word2vec represented a significant advancement in the development of NLP models, it is limited because it can only generate a single vector representation for each unique token in a corpus. As a result, it cannot fully capture the contextual nuances of word meanings that may vary based on their surroundings. For instance, the word ”bannga” in Tshivenda can have two possible meanings, one referring to a chair and the other to a financial institution. To address this limitation, contextualised embeddings must produce multiple vectors for the same token that can change based on the context. Fortunately, this capability is available by default with the emergence of transformer architectures [Ethayarajh, 2019]. Moreover, with the attention mechanisms used by transformer models, tokens that contribute little to the meaning of a document are automatically ignored, requiring minimal pre-processing of the input text to build language models capable of capturing the subtle variations in meaning.

Language model training can be conducted in two main ways: causal and masked language modelling. Causal language models learn by trying to predict the next token in a given sequence [Aghajanyan et al., 2022]. In contrast, masked language models predict a randomly masked token in the sequence. Language models trained this way are useful for tasks like text generation, where coherence and flow of the text are critical [Devlin et al., 2019]. Causal language models, on the other hand, are more appropriate for tasks like language translation, where the input sequence is known, and the goal is to predict the output sequence. In this study, we chose to use masked language models, as empirical results have shown improved performance in classification and entity recognition tasks [Devlin et al., 2019] from language models trained this way. Moreover, the reference study [Ogueji et al., 2021] we have selected used masked language models. Therefore, we follow the same approach to ensure fair comparisons.

3.7.4 Multilingual representation

Multilingual representations have been proposed to discover a shared semantics vector space supporting multiple languages. In this regard, manually aligned global word embeddings trained using word2vec have demonstrated promising results. To adapt this capability to Tshivenda and Sepedi, we leverage a semi-supervised technique based on VecMap [Artetxe et al., 2017] that enables the projection of monolingual embeddings into a common vector space. It has been established that this approach performs well when sufficient bilingual lexicons are available. However, it is worth highlighting that creating such lexicons for the various South African language pairs may entail significant manual effort.

Alternatively, using multilingual pre-training methods such as XLMR or AfriBERTa can boost the learning of shared representations for multiple languages without the need for explicit mapping between languages [Conneau et al., 2020; Muller et al., 2021]. These methods utilise a masked language modelling objective to learn a shared representation that captures cross-lingual similarity, which can be fine-tuned for specific downstream tasks. Following a similar methodology used to train AfriBERTa [Ogueji et al., 2021] and XLM-RoBERTa [Conneau et al., 2020], our study aims to develop a series of pre-trained language models from scratch using a corpus from nine South African Bantu languages. We employ various combinations of these languages based on their perceived levels of similarity. In addition to intrinsic metrics like perplexity, we evaluate the quality of these language models in downstream tasks to assess their efficacy in improving the coverage of NLP applications for Tshivenda.

3.8 Evaluation

This section outlines the criteria used to validate the hypothesis that cross-lingual language models are a better option to improve the coverage of Tshivenda in NLP applications. We provide an overview of the metrics employed to evaluate the quality of different representation models and their performance in downstream short-text classification and entity recognition tasks. We consider both intrinsic and extrinsic methods to investigate if there is a correlation between intrinsic performance and downstream tasks.

3.8.1 Intrinsic evaluation

Intrinsic evaluation is used to evaluate the inherent quality of the representation methods excluding any downstream performance. In contrast, extrinsic evaluation aims to assess the performance of the representation methods on downstream tasks, such as text classification or named entity recognition. While intrinsic evaluation provides an initial assessment of the quality of the representation methods, it may not always reflect their actual performance in real-world applications. Therefore, it is essential also to conduct extrinsic evaluations to determine the effectiveness of the representation methods in practical use cases.

Visualisations

Unlike modern contextualised embeddings, traditional representation methods generate static word vectors that assign each token in the training corpus with a fixed vector representation. These vectors are usually high-dimensional, ranging from 100 to 500 dimensions. Fortunately, these static word vectors can be projected to a 2D or 3D space using principal component analysis (PCA) to manually evaluate how well semantic similarity is captured. The Tensorflow Projector[16] tool provides a simple online interface to visualise and explore such embeddings. This study used this tool to visually assess the embeddings trained using the Skip-Gram model and the aligned vectors trained using VecMap [Artetxe et al., 2017].

Perplexity

Perplexity is a widely used metric in natural language processing for evaluating the effectiveness of language models. It measures the degree of uncertainty or confusion of the model in predicting the next word in a sequence. A lower perplexity score indicates that the model can better predict the next word suggesting it better understands the underlying language structure [Chen et al., 2008]. In evaluating the performance of our language models, we will use perplexity to quickly measure how well they predict missing words in a sentence. This will provide insight into the language model's potential downstream task performance. Additionally, we can assess if there is a significant increase in accuracy after a drop in perplexity.

3.8.2 Extrinsic evaluation

We will use the annotated news headlines dataset described in Section 3.2.2 to set up different experiments to evaluate the quality of our custom-trained language models. The evaluation will encompass various settings, including monolingual text classification performance and few-shot and zero-shot cross-lingual performance between Sepedi and Tshivenda. These evaluations will allow us to determine the efficacy of our models in different contexts, such as classification

[16] https://projector.tensorflow.org/

tasks within a single language and the ability to transfer knowledge between languages with varying degrees of overlap. Additionally, apart from text classification, we will also conduct these experiments on an entity recognition task by using a pre-annotated dataset from SADiLaR [Eiselen and Puttkammer, 2014]

Prior to conducting experiments using state-of-the-art language models, we employ simpler model architectures such as Logistic Regression, Random Forests, and XGBoost as baseline models. Next, we incorporate more advanced models based on Deep Neural Network architectures like LSTM [Hochreiter and Schmidhuber, 1997] and FastText [Joulin et al., 2017] before evaluating the current performance on off-the-shelf large language models such as AfriBerta [Ogueji et al., 2021] and XLM-R [Conneau et al., 2020]. Using this setup, we aim to draw meaningful conclusions on which compute investments will likely help us achieve the research objectives with minimal data requirements. Given the label imbalance in the dataset, the downstream task metric may be biased towards the majority class. To address this issue, we use the weighted F1-score, which considers both precision and recall to gauge how well our models perform across all classes, not just the majority class.

3.9 Ethical Considerations

This section highlights the ethical considerations anticipated in this research study and how they were addressed. All research steps were conducted ethically and legally to the best of our ability. The following is a summary of potential sources of risks and how they were mitigated.

3.9.1 Informed consent

To prevent exploitation, we made the volunteers sign an ethical consent form before accessing the platform.

3.9.2 Personal Information Protection

Since the research did not require participants to provide personally identifying responses, no risk was identified regarding information leaks. The emails used to create logins were only shared with the platform administrator and immediately removed from chat platforms once the user got on-boarded.

3.9.3 Fairness

We have also taken steps to identify the potential source of bias from the model results. We have also taken steps to understand and describe the data sources and the topics within the data used to train to improve the interpretability of the results. Moreover, the models proposed in this phase of work will not be made public as we cannot guarantee that there is no risk of misinformation or discriminative results from the model.

Chapter 4

Experiment Design

The experiment design section provides a detailed account of the research design, including the study's population, sample, data collection methods, and statistical analyses. Our objective is to make it seamless to produce any results reported in the study. The outline of the chapter is as follows:

- Introduction - This section provides additional context on the subsequent decisions made by restating the study's goals. Moreover, it will emphasise how the decision-making process aligns with the study's objectives.
- Experiments - This section describes the various experimental setups conducted and a comprehensive account of the key parameters used. Furthermore, the training inputs, compute environments and procedures are described in detail.
- Model selection criteria - Finally, a discussion on how the best models are selected will be presented in a way that helps us answer the research questions.

4.1 Introduction

We aim to gather more insights into the current usability of NLP tools and pre-trained models for Tshivenda text. Furthermore, we explore the feasibility of compensating for the lack of training resources by leveraging the available resources from more commonly spoken SA Bantu languages. Furthermore, before training our custom language models, we intend to conduct baseline experiments using state-of-the-art (SOTA) language models, such as AfriBERTa [Ogueji et al., 2021] and XLM-RoBERTa [Conneau et al., 2020], along with established machine learning algorithms, such as Logistic Regression and Decision Trees. Finally, we highlight strategies to overcome limited pre-processing tool support for Tshivenda and Sepedi.

Initially, we will consider monolingual embedding spaces built using FastText. Because it uses subword tokenisation, FastText has an advantage over Word2Vec as it allows us to obtain high coverage of the language vocabulary, including terms that may not be part of the training set.

Additionally, we will train Word2Vec embeddings with comparable dimensions for comparison purposes. Next, we train several classic ML and DNN models as baselines. Furthermore, we will attempt to fine-tune existing pre-trained large language models based on XLM-R using the Sepedi news headline dataset to evaluate its present cross-lingual transferability to Tshivenda. Even though the original XLM-R model training set does not include Sepedi or Tshivenda, three African languages are included in the training set, albeit in small quantities. These languages are Swahili, Xhosa, and Amharic. Therefore, following the approach in [Hedderich et al., 2020], we believe that XLM-R should have some transferability to these unseen languages. Finally, we will resume training on XLM-R [Conneau et al., 2020] and AfriBERTa [Ogueji et al., 2021], on Tshivenda and Sepedi texts to investigate whether we can obtain any performance gains on downstream tasks.

In the second modelling stage, we will combine the existing monolingual embeddings for Tshivenda with monolingual embeddings for Sepedi to generate a unified semantic vector space. We will employ a semi-supervised approach using a small bilingual dictionary to align the embeddings with VecMap [Artetxe et al., 2017] to achieve this. We obtain the bilingual lexicon from parallel texts in Flores using the FastAlign [Dyer et al., 2013] tool. Finally, we will train a classifier to assess the unified semantic vector space in zero-shot and few-shot scenarios, with Sepedi as the source language and Tshivenda as the target language. Following a similar methodology used to train AfriBERTA and XLM-RoBERTa, we aim to build a set of custom pre-trained language models from scratch using a corpus from nine SA Bantu languages. We will experiment with multiple combinations of these languages based on the perceived degrees of similarity. The custom language models will then be used in downstream tasks to evaluate their extrinsic quality. Furthermore, we will compare this performance with perplexity to see if there is any positive correlation between classification metrics and perplexity. We hypothesise that the zero-shot classification performance will be significantly improved by building a representation space using all SA languages. Moreover, we expect that the resulting language models and training strategies can be helpful for future cross-lingual transfer applications for SA Bantu languages.

The upcoming section outlines the experimental setups created to accomplish our research objectives, starting from baseline model training and evaluation, customised language model training, and fine-tuning.

4.2 Experiments

The news classification experiments were set up as single-label multi-class classification problems. In order to achieve a balanced dataset, we assigned single labels based on the frequency of each category. This was necessary as the annotation results often included multiple tags per example. In addition, we encoded all texts in UTF-8 format and saved them using pipe delimiter and Parquet formats to improve portability when working across multiple platforms and avoid issues with commas.

The experiments utilised up to four different training environments based on Debian OS. The primary host was a university-provided lab server running Ubuntu-20.04, which contained two

NVIDIA RTX A6000 GPUs with 96GB VRAM, 64 vCPU cores, and 128GB memory. However, this was shared among several researchers, so its total capacity was only sometimes available. In addition, three cloud VMs with a single Tesla K80, Tesla V100 on GCP, and T4 series from the Azure ML platform were used. The cloud VMs were each fitted with about 4vCPUs and 7.5GB-16GB of RAM. Furthermore, we ran some small-scale experiments on personal PCs. One of the personal PCs was fitted with eight vCPUs, an AMD 4GB GPU, and 16GB RAM, while the other was fitted with NVIDIA T600, 16CPUs, and 32GB RAM.

Throughout all experiments, we aimed to leverage a variety of open-source Machine Learning Operations(MLOps) tools to manage our experiment pipelines and track metrics. In particular, we used Comet ML[1] for experiment logging and performance comparisons of results throughout the training cycles. Additionally, we used Data Version Control (DVC) [de la Iglesia Castro, 2023] to create repeatable training pipelines and schedule different experiments to run consecutively. This made it seamless for us to retrain all models as more data became available. It was also easy to read off important metrics emitted during training, with the ability to easily compare different runs to see the effect of manipulating different parameters. Finally, we utilised Hydra [Yadan, 2019] as a hierarchical configuration tool to manage model architecture and standard hyperparameter settings.

4.2.1 Baseline models

4.2.1.1 Pre-trained Multilingual BERT models

This experiment involves fine-tuning a series of state-of-the-art multilingual models on the downstream Tshivenda and Sepedi tasks. The selected models are AfriBERTa and XLM-RoBERTa, both of which aim to improve multilingual performance in large language models. XLM-RoBERTa is currently the top-performing multilingual language model, surpassing mBERT. Meanwhile, AfriBERTa is tailored to improve performance for 11 popular African languages.

Datasets

To prepare our news headline dataset for supervised machine learning, we divided it into three sets: 80% for training, 10% for development, and 10% for testing. We did not experiment with different ratios but may consider doing so in future work. We used the stratify option on the Sklearn *train_test_split* function to ensure that each news category was represented in each set. For Sepedi, we had approximately 5,000 headlines in the training set and 2,000 and 3,000 in the validation and testing sets, respectively. Similarly, for Tshivenda, we used 3.6k examples for training and 1.5k and 2.2k examples for validation and testing respectively.

Before passing the datasets to the training stage, we run a preparation pipeline to ensure the data is in a format supported by HuggingFace [Wolf et al., 2019] datasets library. All normalisation steps are performed in this pipeline, and the dataset is saved in a staging directory exclusively used for fine-tuning language models. Our normalisation process for language model fine-tuning includes removing accents and whitespaces and converting the text to lowercase.

[1] https://www.comet.com/site/

Additionally, we have configured this normalisation through Hydra to enable future performance evaluation using different settings.

Training environment

All the experiments were conducted in a GPU environment, using the HuggingFace library to fine-tune a pre-trained model. The training scripts and other supporting Python modules were made available as a reusable Python package that can be installed and run in any environment with *Python3.9.x* installed.

4.2.1.2 Classic ML algorithms

In this stage, we trained several linear and non-linear models, including Logistic Regression and Support Vector Machines, using default settings on Sklearn. Additionally, we trained tree-based models, such as Random Forests and XGBoost models, using TF-IDF feature extraction. We then conducted a series of hyper-parameter optimisation runs to reduce over-fitting.

Datasets

To prepare the data for training Sklearn models, a similar process was followed as outlined in Section 4.2.1.1. The dataset was normalised and staged in a dedicated folder for training ML models with sklearn. The data was divided into two sets, an 80% training set and a 20% test set. Text columns were normalised by removing punctuation, extra whitespaces, accents and changing all tokens to lowercase. To improve the learning process of the model, we removed any rows with missing or insufficient text (less than 10 characters). We also limited the length of longer texts to 512 characters. The training sets for Sepedi and Tshivenda had 7317 and 5967 examples respectively for training, and 1839 and 1492 examples respectively for testing.

Training environment

To train our machine learning models for different languages, we used Python3.9.x and Scikit-learn version 1.2.1 on a local CPU environment since GPU training is not supported or required by native Sklearn. We trained one Sklearn model at a time and utilised the *n_jobs=-1* option to leverage multi-core training.

Hyper-parameter optimisation

We utilised Optuna [Akiba et al., 2019], an open-source optimisation library with extensive support for various ML frameworks, to select the hyper-parameters for the best models. For both Sepedi and Tshivenda, we implemented the same pipeline using DVC's [de la Iglesia Castro, 2023] *foreach* functionality. As the annotated data arrived in small increments, this approach helped us to conveniently incorporate additional data into the pipelines without modifying the optimisation flow. Table 4.1 provides an overview of the settings used to set up the optimisation jobs.

Model Type	Parameter	Param Type	Search Space
tfidf	ngram_max	int	[1, 5]
	max_df	float	[0.1, 1.0]
	min_df	int	[1, 10]
	max_features	int	[1000, 50000]
logit	loss	string	log_loss
	class_weight	categorical	[None, ”balanced”]
	penalty	categorical	[”l2”, ”l1”]
	max_iter	int	[50, 500]
svm	loss	categorical	[”hinge”]
	penalty	categorical	[”l2”, ”l1”]
	max_iter	int	[50, 2000]
	alpha	float	[1e-5, 1e10] (log scale)
	class_weight	categorical	[None, ”balanced”]
svc	C	float	[1e-10, 1e10] (log scale)
	gamma	float	[1e-10, 1e10] (log scale)
	kernel	categorical	[”rbf”, ”sigmoid”]
	class_weight	categorical	[None, ”balanced”]
random_forests	n_estimators	int	[20, 320] (step=20)
	max_depth	int	[3, 10] (step=1)
	criterion	categorical	[”gini”, ”entropy”]
	class_weight	categorical	[None, ”balanced”]
xgboost	n_estimators	int	[20, 320] (step=20)
	max_depth	int	[5, 10] (step=1)
	learning_rate	float	[1e-5, 0.3]
	objective	categorical	[”binary:logistic”, ”binary:logitraw”]

Table 4.1: Hyper-parameter search space for Bayesian optimisation on classic ML models

4.2.1.3 Deep Learning models

In this stage, we have evaluated the performance of these traditional models to determine the extent to which transformers can improve on their predecessors. We use Tensorflow [Abadi et al., 2015] version 2.8 to train different sets of models with embedding dimensions ranging from 128 to 500.

LSTM

Before the advent of the transformer [Vaswani et al., 2017], traditional LSTM (Long Short-Term Memory) [Hochreiter and Schmidhuber, 1997] models were considered state-of-the-art for processing text. These models are a type of recurrent neural network capable of processing sequential data, such as text, and capturing the context of words in a sentence. They have been widely used in various NLP tasks, including sentiment analysis, machine translation, and text classification. However, traditional LSTM models have limitations in capturing long-term dependencies and handling vanishing and exploding gradient problems. These limitations were addressed by the transformer model, which utilizes self-attention mechanisms to capture global dependencies and has demonstrated superior performance in a variety of NLP tasks. As a result, transformer models have emerged as the new state-of-the-art in NLP, surpassing traditional LSTM models. Despite this, we still include traditional LSTM models as baselines for our study.

CNN

Although CNNs (Convolutional Neural Networks) were originally developed for image classification and object recognition tasks, they have also shown promising results when applied to NLP tasks [Lei et al., 2015]. CNNs can extract features from images by applying filters or convolutions to small patches of the image, but the same principle can be applied to text by considering it as a 1-dimensional sequence of vectors. In NLP, CNNs have been used for various tasks, including sentiment analysis, text classification, and named entity recognition, and they have demonstrated competitive performance compared to traditional NLP models like LSTM while being computationally efficient [Lei et al., 2015]. While CNNs are not as versatile as transformer models, they can be beneficial in situations where computational resources are limited, and the text input has a fixed length or can be padded to a fixed length.

Datasets

Because Bi-LSTMs and Text-CNNs are primarily based on non-contextual word embeddings, they can be sensitive to non-textual symbols, such as punctuation and accents, similar to Sklearn models. Therefore, we configured the cleaning transformations to remove punctuation, accents, extra whitespace and convert all text to lowercase. This pre-processing pipeline was almost standard before transformers came and made some of the phases obsolete. Once all the text is cleaned and empty rows are removed, the data is saved in a specific DNN staging folder, with each class category saved as a subdirectory containing individual headlines saved as text files. Finally, the datasets are loaded into the training pipeline using the Keras *text_dataset_from_directory* utility function. The dataset is divided into train, validation, and test sets, split using a 70%:10%:10% ratio, respectively.

Model Architecture

To train the Bi-LSTM models, we used a base model with two BI-LSTM layers containing 64 neural units, followed by a 64-unit dense layer fed to the output layer containing a softmax activation. The training loop utilised the standard Adam optimiser with sparse categorical entropy loss in Tensorflow-Keras [Abadi et al., 2015; Chollet et al., 2015]. For CNN models, we utilised a 1D convolutional layer followed by a max pooling layer. A dense layer with 32 units and a dropout was then added before feeding into the output layer with softmax activation.

Hyperparameter optimisation

Parameter	**Type**	**Value**
Vocabulary size	Integer	$1000 \leq vocab_size \leq 100000$
Embedding dimensions	Integer	16, 32, 48, 64, 100, 128, 256, 300, 500
Maximum sequence length	Integer	$32 \leq input_length \leq 512$
Number of LSTM units	Integer	$4 \leq lstm_units \leq 512$
Number of dense units	Integer	$4 \leq dense_units \leq 512$
Use dropout	Boolean	use_dropout
Dropout rate	Float	$0.1 \leq dropout_rate \leq 0.7$
Learning rate	Float	$1e-4 \leq lr \leq 1e-2$

Table 4.2: Hyper-parameter search space for Bayesian optimisation on LSTM models

In the case of deep learning, we opted for keras_tuner [Chollet et al., 2015] instead of Optuna [Chollet et al., 2015] since our models were trained using the Keras [Chollet et al., 2015] API

Parameter	**Type**	**Value**
Vocabulary size	Integer	$1000 \leq vocab_{s}ize \leq 100000$
Embedding dimensions	Integer	16, 32, 48, 64, 100, 128, 256, 300, 500
Maximum sequence length	Integer	32, 64, 128, 256, 512
Number of 1D Conv units	Integer	$4 \leq conv1d_{u}nits \leq 512$
Number of dense units	Integer	$4 \leq dense_{u}nits \leq 512$
Kernel size	Integer	$4 \leq kernel_{s}ize \leq 32$
Pool size	Integer	$4 \leq pool_{s}ize \leq 32$
Use dropout	Boolean	use_dropout
Dropout rate	Float	$0.1 \leq dropout_{r}ate \leq 0.7$
Learning rate	Float	$1e^{-5} \leq lr \leq 1e^{-2}$

Table 4.3: Hyper-parameter search space for Bayesian optimisation on CNN models

which is built into TensorFlow [Abadi et al., 2015]. Our experiments were set up to optimise the validation loss to ensure the model did not over-fit the training data. For each hyperparameter configuration, we train the model for 10 epochs with an early stop condition enabled in case the loss does not increase for 3 consecutive epochs. A table summarising the configuration for the auto-tuning jobs for LSTM and CNN experiments is shown in Table 4.2 and Table 4.3, respectively.

4.2.1.4 FastText models

In this experiment, we use the FastText library to train a multi-class topic classification model. Initially, we train monolingual models for Sepedi and Tshivenda, using default hyperparameters. Later, we use FastText's built-in hyperparameter optimisation functionality to enhance the performance of the models.

Datasets

FastText datasets typically require special processing since the training data labels must be prefixed and suffixed with two underscores. As a result, we built a transformation pipeline to load data from the finished annotations and convert the rows to a format required by FastText. The processed data was staged in a dedicated directory for training and optimising any FastText model. The same pre-processing steps as in Section 4.2.1.1 were applied. The train and validation set are saved using a pattern that identifies the language, prefixed by the data split name, such as "train.nso" for Sepedi training set or "valid.ven" for Tshivenda validation set. This pattern allows us to easily load the appropriate data split during the model training and evaluation phases for any language code.

Training environment

To train FastText models, we use a CPU environment with 4 CPUs and 16GB of RAM, as GPUs are not necessary. During the hyperparameter tuning stage, we limit the model size to 2M parameters to avoid running out of memory.

Hyperparameter optimisations

Fortunately, the FastText library has a built-in auto-optimisation function that produces the best parameters for the given dataset. We restricted the model to 2 million parameters to prevent resource exhaustion, as recommended in the library documentation [Joulin et al., 2017]. We also limited the training duration to 10 minutes to prevent over-fitting. This allowed us to get the optimal settings for the learning rate, training epochs and dimensions, among others.

4.2.2 Multilingual representation models

4.2.2.1 Word2Vec

In addition to the embeddings automatically learnt during classification model training, we train a set of monolingual embeddings with varying dimensions for Sepedi and Tshivenda. These embeddings are trained using the same raw data to train our custom language models. Finally, the resulting embeddings are projected to a common semantic vector space using VecMap.

Datasets

For this task, we used a collection of raw datasets from Sepedi and Tshivenda languages to train FastText and Word2vec embeddings with dimensions of 128, 300, and 500. We also utilised a small, bilingual lexicon of 300 entries to align embeddings with VecMap. We obtained a subset of parallel sentences from SADiLar [Eiselen and Puttkammer, 2014], Flores [Goyal et al., 2021; Guzmán et al., 2019; Team et al., 2022], and multilingual terminology datasets scraped from various websites to generate the lexicon. We leveraged the FastAlign [Dyer et al., 2013] utility to accomplish this. A total of 179k documents were used to train the Sepedi embeddings and 73k samples of Tshivenda documents were used to train the embeddings, resulting in 20392 and 9083 tokens, respectively. Standard text cleaning steps, including punctuation removal, accent removal, lower casing, and extra white space removal, were also applied before running the training scripts.

Model Architecture

The Word2Vec embeddings are trained for 50 epochs using the skip-gram method provided by Gensim [Rehurek and Sojka, 2011]. We use a window of 10 tokens, ignoring tokens with less than two occurrences. Meanwhile, the FastText models are trained using the built-in FastText trainer with automatic hyperparameter optimisations. VecMap is set to use Cupy[2], which allows it to train faster on GPU-powered hosts. The training of the embeddings was conducted on local PCs with 16 CPUs, while the alignment was done on a host fitted with a Tesla K80 GPU running on the Google Cloud platform. We used VecMap to align vector spaces in a semi-supervised way, utilising a small dictionary obtained from parallel sentences for Tshivenda and Sepedi.

VecMap uses the orthogonal Procrustes objective score to align embeddings of two languages by minimising the difference between them. The orthogonal Procrustes objective score measures the similarity between two matrices by transforming one matrix to minimise the Frobenius norm of the difference between them while ensuring that the transformed matrix remains orthogonal [Artetxe et al., 2017]. The drop probability prevents over-fitting by ”dropping out” individual

[2] https://github.com/cupy/cupy/

neurons in the neural network during training. The drop rate gradually decreases during training until a desired objective is reached or the set number of iterations is exceeded.

4.2.2.2 Zabantu

The centre stage of the study involves a series of experiments where we train language models from scratch using a combination of South African Bantu languages. We call these collections of models *Zabantu* with ”ZA” representing Southern Africa and ”Bantu” representing Bantu languages. In total, we train four groups of language models;

- Zabantu-VEN : A monolingual language model trained on 73k raw sentences in Tshivenda
- Zabantu-NSO : A monolingual language model trained on 179547 raw sentences in Sepedi
- Zabantu-NSO+VEN: A bilingual language model trained on 179547 raw sentences in Sepedi and 73k sentences in Tshivenda
- Zabantu-SOT+VEN: A multilingual language model trained on 479k raw sentences from Sesotho, sepedi, Setswana, and Tshivenda
- Zabantu-BANTU: A multilingual language model trained on 1.4M raw sentences from 8 South African Bantu languages

The language models were first tested and verified on Google Colab before being deployed to the primary GPU-powered server for training. The server had 2 NVIDIA RTX A600 GPUs with 95GB VRAM and 64 compute-optimized vCPUs.

Datasets

The raw datasets used for training word embeddings in Section 4.2.2.1 are also used to train new language models from scratch. The only text-cleaning steps performed are removing extra white spaces, accents, and lower casing. The data is split into training and evaluation sets using a 75%:25% split ratio for each model family. The evaluation split is used to compute perplexity in each individual language after every epoch. We follow the same strategy used by AfriBERTa to sample the training batch from the multilingual corpora, which involves randomly selecting a language and then sampling a fixed number of examples from that language’s corpus. In addition, we use a masking probability of 0.15 for masked language modelling (MLM) during pre-training, which involves randomly masking a certain percentage of tokens in the input sequence and predicting their original values based on the surrounding context. By setting the MLM probability to 0.15, we ensure that the model is exposed to sufficient masked tokens during pre-training, which helps it learn to effectively handle missing information and fill in the gaps in the input sequence.

Model Architecture

Each family of models is trained following the XLM-RoBERTa architecture with dedicated tokenizers obtained using SentencePiece [Kudo and Richardson, 2018]. XLM-R is a variation of BERT that incorporates cross-lingual pre-training and a larger model size to improve its performance on multilingual NLP tasks. Furthermore, XLM-R uses a larger model size than BERT, with up to 550 million parameters, allowing it to capture more complex linguistic patterns and relationships in the input data [Conneau et al., 2020]. We use this as a reference architecture since it achieves state-of-the-art results on various cross-lingual tasks, as shown in [Ogueji et al., 2021; Alabi et al., 2022].

Observing a similar process to AfriBERTa [Ogueji et al., 2021] and XLM-RoBERTa [Conneau et al., 2020], we train our models using a stream of sequences from different languages. Given a list of L languages, we sample a batch of size B from a randomly chosen language L for training. The languages are chosen to maximise the diversity of the languages in each epoch. If all the examples from a single language are exhausted, we choose a second language to sample. If we have sampled all the available languages, we reset the data and start from scratch. This way, the model can theoretically train for an unbounded number of epochs. During training, the language model is evaluated using the Masked Language Modelling (MLM) objective. The Next Sentence Prediction (NSP) objective is not part of this architecture since it was found that the performance of MLM alone is good enough for RoBERTa [Liu et al., 2019].

Each model is trained using BPE and Unigram tokens, with vocabulary sizes ranging between 30k and 250k tokens. Like AfriBERTa, the base models contain 70k tokens and are trained for ten epochs, except for Tshivenda monolingual models, which were trained on 30k tokens due to limited training examples. In addition to the base models, we have larger models trained on 150k and 250k tokens. Furthermore, we repeat each training configuration for 20, 50, and 100 epochs. We use the HuggingFace [Wolf et al., 2019] trainer API for PyTorch [Paszke et al., 2019] to train the models, with Comet ML[3] used to track training metrics. The training batch size was capped at 8 for all experiments and 16 for evaluation, with 16-bit floating-point precision enabled. The loss was evaluated every 1000 steps, and a checkpoint was saved in case of intermittent deallocation when training in spot cloud VMs. The learning rate is initially set to 1×10^{-4} and allowed to gradually decrease over time using the default linear schedular in HuggingFace.

Training environment

Most of the training is conducted on the main server with 2xNVIDA RTX A600 GPUs. In addition, we run supplementary experiments on Tesla K80, T4 and V100 GPUS at different stages of the study to ensure all models were sufficiently trained. We use DVC [4] to queue experiments so that the training continues running in the background consecutively. Furthermore, we track the training progress on Comet ML [5], which has native support for models trained using HuggingFace [Wolf et al., 2019].

[3] https://www.comet.com/site/
[4] https://dvc.org/
[5] https://www.comet.com/site/

4.2.3 News Topic Classification models

4.2.3.1 Monolingual performance

In this stage, we tested the Zabantu custom language models on a news topic classification task using Tshivenda and Sepedi news headlines datasets described in 3.3. Due to the imbalance in class labels, we enacted two correction strategies to balance the label distribution. Firstly, we consolidated all classes with less than a set frequency into a default label called 'other'. Secondly, Second, we manually grouped classes with similar topics, such as ”human interests”, ”arts, culture, entertainment, and media”, and ”conflict, war, and peace”, into a single label called ”Society”. This approach proved practical as these topics all relate to societal issues, like protests for service delivery and other human rights or welfare matters.

Datasets

Similar to the dataset used in Section 4.2.1.1. We create 80:10:10 training, validation, and test splits to fine-tune each family of *Zabantu* language models on a text classification task. We apply the same preprocessing steps used in the training stage of the language models. This means that, unlike in classic ML or DNN models, we do not remove punctuation or stopwords before training a classifier.

Model Architecture

By attaching a classification layer to the language models, we are able to obtain the baseline monolingual score for each language model trained from scratch. We set the training batch size to 4 and trained for 5 epochs, using the weighted F1 score to evaluate the fine-tuned model.

Training Environment

We use the same training environment described in Section 4.2.1.1. The experiments are organised using DVC pipeline definitions, allowing us to use YAML to describe fine-tuning recipes that make evaluating different variations of language models easy.

4.2.3.2 Zero-shot performance

In this stage, we evaluate the zero-shot and few-shot performance of the language models. To achieve this, we consider four different zero-shot strategies. These include using 10, 50, and 100 labelled examples from the target language, in addition to the source language, to train a linear classifier for each target language. In the fourth strategy, named ”Use both,” we concatenate the labelled examples from the source language with those from the target language and train a single classifier. This evaluation aims to determine the effectiveness of the language models in adapting to new languages with limited labelled data.

We test Zabantu models against benchmarks set by Deep learning zero shot models trained on the vectors aligned using VecMap. Since VecMap returns output embeddings for each language in separate vector files, we combine the vectors into one file before loading them as an embedding matrix. This matrix is then used to initialise the embedding layer in Keras training. We then train the model as before but in a multilingual setting. The same data used to train baseline deep learning models is used to get zero and few-shot performance between Sepedi and Tshivenda. We first load each language's dataset separately and then combine them into a single dataset depending on the selected zero-shot strategy. For example, if the strategy is ten shots, we load the Sepedi dataset into a TensorFlow dataset, then sample 10 examples from each class in the Tshivenda dataset and add them to the train and validation splits. Finally, we evaluate the performance of the target language using the weighted F1-score as the primary metric.

4.3 Model selection criteria

This section outlines the criteria used for systematic model selection. Our main objective is to select models that perform best in the target tasks taking into cognisance other factors such as computational efficiency and data requirements. We start by outlining the key evaluation metrics used to assess model performance in the different tasks. We then discuss how we balance the trade-off between model complexity and accuracy.

The objective of this study is to develop high-performing models that can compete with state-of-the-art results while utilising minimal resources. Typically, achieving this goal requires vast amounts of data and expensive computing hardware, making it challenging to achieve for languages from less-developed regions. Therefore, our selection criteria prioritises the development of models that closely match SOTA benchmarks while remaining computationally efficient, making these models more accessible to a broader audience.

Given the imbalanced nature of our large dataset, we have chosen F1-score and validation loss as our primary performance metrics. Validation loss is an essential indicator of over-fitting or the presence of outliers in our news headline datasets. Moreover, we use a weighted F1-score to evaluate each class according to its frequency. In addition to these metrics, we consider the relationship between training time, epoch count, and GPU utilisation to evaluate whether complex models are worth the additional computational cost. Specifically, we identify the model that achieves the highest F1 score while converging quickly on the same compute target as the superior choice.

4.4 Expectations

Considering the small size of the news headlines dataset, it is understandable if deep learning models need to perform better. However, using pre-trained embeddings can enhance performance by increasing token coverage. FastText should also overcome this because it can synthesise word vectors for tokens that do not appear during training. This capability is absent in typical deep learning models which use Word2Vec embeddings. While Sklearn models may perform well on small datasets, BERT models are expected to outshine them. Additionally, models trained on the specific SA languages (Tshivenda and Sepedi) used in the news corpus will likely perform better than larger multilingual models such as XLM-R and AfriBERTa. Nonetheless, we expect that the current size and quality of the annotations may distort these expectations.

4.5 Summary

This section provided a detailed scope and plan for conducting experiments to confirm our hypothesis. In the next section, we will share the outcomes of various empirical experiments.

Chapter 5

Results

This results section summarises the outcomes of our experiments using numerical metrics and graphical or tabular figures. Our study explored cross-lingual learning techniques to enhance NLP coverage, which is currently limited for many low-resource languages like Tshivenda. To achieve this, we utilised traditional machine learning baseline models, monolingual deep learning models, and deep learning models with cross-lingual embeddings to identify gaps in the current NLP ecosystem that must be addressed to improve the representation of Tshivenda.

In addition to traditional ML approaches, we trained a series of custom large language models and fine-tuned them on a news topic classification task. We evaluated monolingual and cross-lingual zero-shot cases to identify how to bridge the gap in Tshivenda NLP coverage. Our goal was to see how effective multilingual models can be in improving low-resource languages like Tshivenda. The expected outcome of the study is to contribute to developing more accurate and efficient NLP systems for low-resource languages like Tshivenda, with potential implications for communication, education, and socio-economic development in Tshivenda-speaking communities. The upcoming sections will showcase the results from the experiments that were conducted as described in Section 4.2.

5.1 Baseline model performance

5.1.1 Classical Machine Learning models

In this section, we will share the outcomes of training classic machine learning models from three families: tree-based, linear, and non-linear. As previously mentioned in Section 4.2.1.2, all models underwent the same pre-processing steps and were trained on the same dataset using the Scikit-learn [Pedregosa et al., 2011] library. First, we will present the results from training using the default hyper-parameters, followed by the outcomes from optimised parameters achieved through Bayesian hyperparameter search.

5.1.1.1 Default hyper-parameters

Inputs	**Baseline**					
	Logit	**SVM**	**SVC**	**RF**	**Xgboost**	**Average**
Tshivenda	**79**	77.6	62.6	67	70	71
Sepedi	**74**	74.2	59.7	45.1	55.4	61.6
Average	76.5	75.9	61.15	56.05	62.7	-

Table 5.1: Weighted F1-scores(%) for the news topic classification task using classic ML models

According to the results presented in Table 5.1, it can be observed that even simple models such as logistic regression, without any hyper-parameter tuning, were able to achieve almost 80% f1-scores for Tshivenda and 74% for Sepedi. Although Sepedi had more than 7.3k training examples compared to Tshivenda's 5.9k, it had inferior average performance across all classic ML models, particularly in tree-based models. This could be due to the fixed vocabulary size of 5k which might have been insufficient for vectorising Sepedi text. It is also possible that some information was lost during the augmentation process through back-translation for Sepedi headlines.

A closer examination of the confusion matrix results generated from the Tshivenda test set in Figure 5.1 reveals that the Logistic regression model, which performs the best, is adept at identifying most topics such as Crime, Health, Labour, and Sports. However, there are numerous instances where the model wrongly predicts Crime, especially on categories such as Politics, Business, and Society. As per the analysis conducted in Section 3.4.2, the training data was dominated by Crime and Political news, which could explain the model's tendency to prioritise the crime category.

We also observe a decline in the recall score as the number of examples per category drops. For example, the sports category had the lowest occurrence and obtained the lowest recall score of 58%. On the contrary, crime had the highest number of occurrences and achieved the highest recall score of 87%. These findings highlight the importance of having enough examples per label to achieve good performance.

5.1.1.2 Optimised hyper-parameters

The optimised hyper-parameters were obtained using the Optuna [Akiba et al., 2019] library. Efforts were made to balance the need to get the best accuracy results without over-fitting the training data. This was achieved by shifting the focus from maximising accuracy to minimising the validation loss as the primary objective. The parameter search scopes used for each trained model can be viewed in Table 4.1.

For the most part, we do not observe any significant changes in performance across all model after hyper-parameter optimisation. This suggests that our progress may be hindered by the relatively low complexity of the chosen models or the limited number of training examples. We still observe a surprisingly low score for the random forest model on Sepedi News, which could point to potential data quality issues or a sub-optimal hyperparameter setting. We leave the

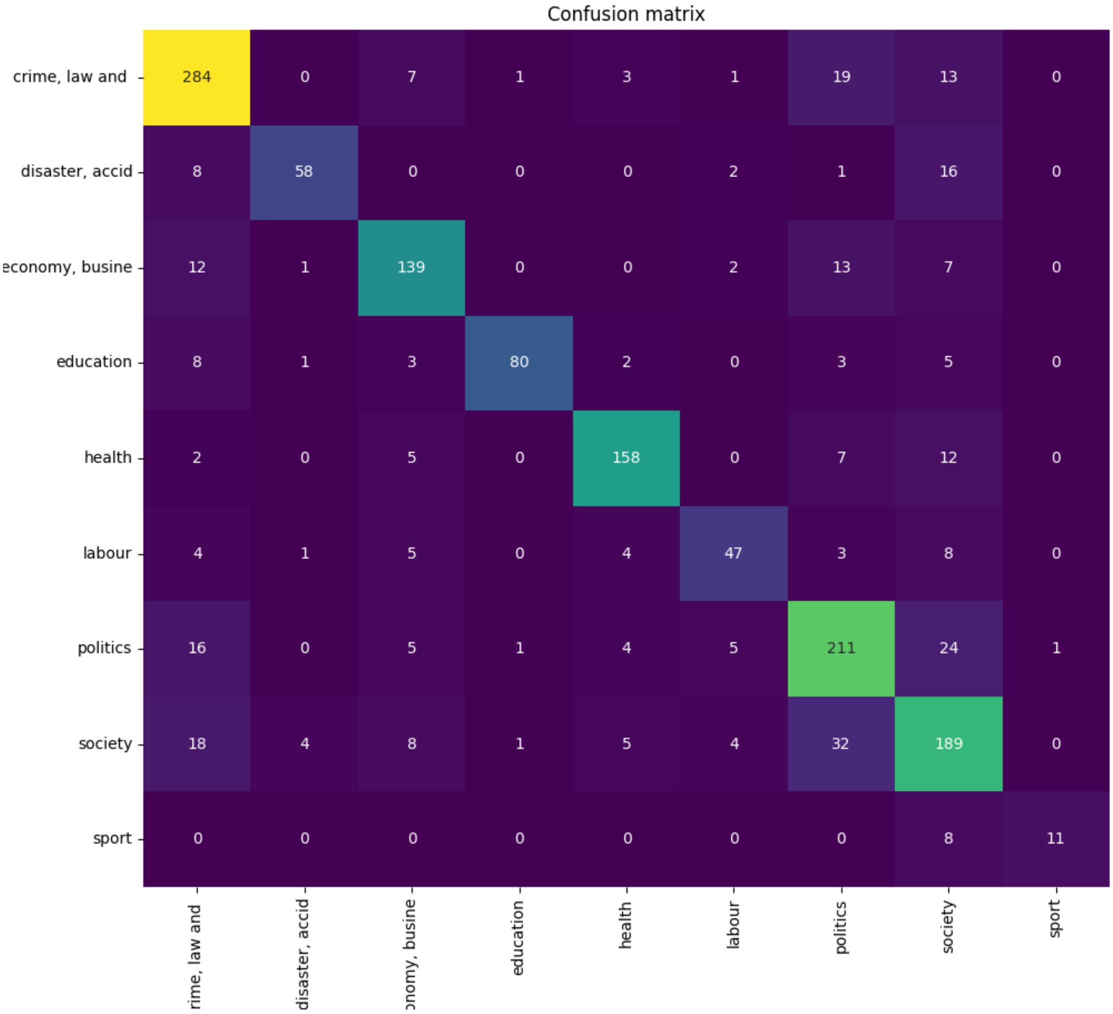


Figure 5.1: Logistic regression news topic classifier confusion matrix on Tshivenda news test set

Inputs	**Optimised**					
	Logit	**SVM**	**SVC**	**RF**	**Xgboost**	**Average**
Tshivenda	**79**	78.4	77.4	67	70.1	74.2
Sepedi	**75.4**	73.9	73.7	45	55.4	64.6
Average	77.2	76.15	75.55	56	62.75	-

Table 5.2: Weighted F1-scores(%) for the news topic classification task using Optimised ML models

investigation of this observation as a possible task for future work. Overall, we see commendable performance from these simplistic models, which are very quick to train.

Worryingly, Sepedi performance still lags behind Tshivenda, even though the best classifier's(Logistic regression) F1 score has now improved by 1%. However, looking at Figure 5.2, we still observe a notable number of wrong predictions which probably point to some inherent difficulty in distinguishing the selected news genres in this study. Better representations with higher vocab coverage can likely improve these results. Moreover, this time the *society* category seems to be best classified compared to the previous iteration for Tshivenda. We see similar

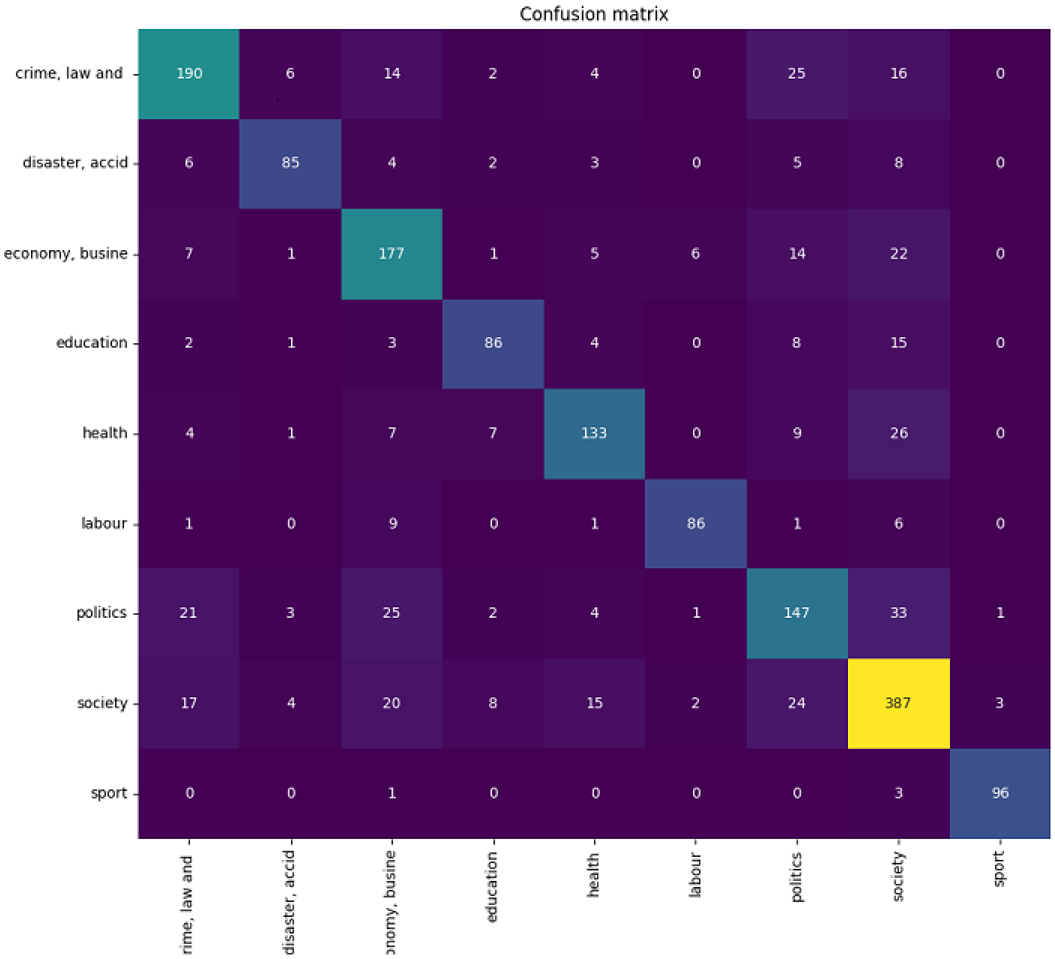


FIGURE 5.2: Logistic regression news topic classifier confusion matrix on Sepedi news test set

misclassification patterns between crime, society, and politics as in Tshivenda in Figure 5.1. This time sports seem to have a good recall, benefiting from augmentation, which increased the frequency from 1% seen in Tshivenda texts to 5%.

5.2 Deep neural network model performance

In this section, we investigate whether neural network-based models can enhance the baseline performance of the classification ML models presented in Section 5.1.1. First, we examine a scenario where the embeddings are learned jointly with the classification weights. Subsequently, we explore a scenario where embeddings are pre-trained using a larger unlabelled corpus that is not a part of the news headlines. We use Keras Tuner [Chollet et al., 2015] to tune the architectures of the Bi-LSTM and CNN models to attain the best possible hyper-parameter values. Our experiments are designed to minimise validation loss, which will help the models avoid overfitting, a likely issue considering we have less than 10k training points.

5.2.1 Monolingual news classification performance

The initial results were obtained using default training parameters which were chosen using heuristics as described in Section 4.2.1.3. The presented score is the weighted F1 score obtained on a test set after training for seven epochs. For these experiments, we used a fixed vocabulary size of 10,000, a learning rate of 1×10^{-3}, and a batch size of 32. The LSTM model architecture consists of two bidirectional LSTM layers with 64 units each, followed by a dense layer with 64 units, and a softmax layer with the same number of units as the number of news genres. The CNN model includes an embedding layer with 128 dimensions and a sequence length of 64. In each experiment, the data was split into train, validation, and test sets with a split ratio of 0.1.

Inputs	**Default performance** [%]			**Optimised performance** [%]		
	Bi-LSTM	**CNN**	**Average**	**Bi-LSTM**	**CNN**	**Average**
Tshivenda	69.8	**75.2**	72.5	69.7	68	68.85
Sepedi	67.6	**72.3**	69.95	70.9	66	68.45
Average	68.7	73.75	-	70.3	67	-

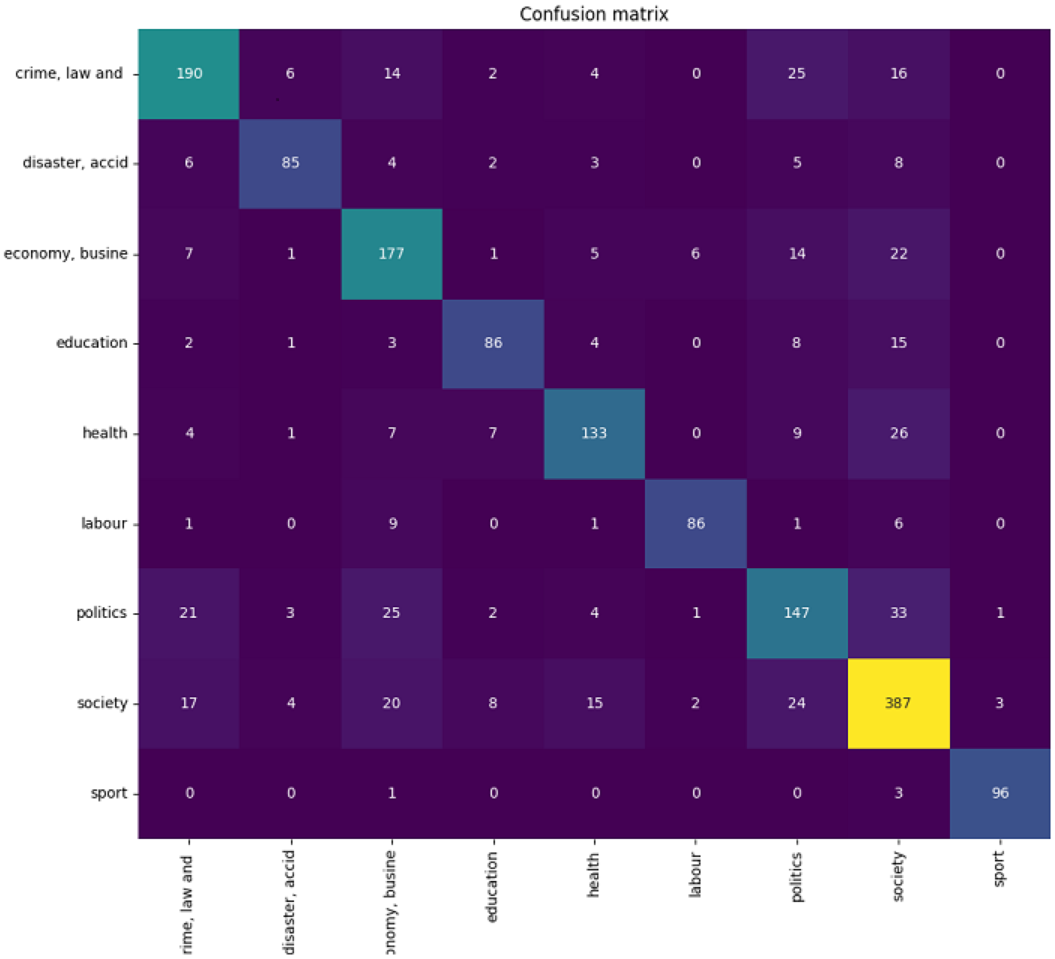
TABLE 5.3: Weighted F1-scores[%] for the news topic classification task obtained from baseline Deep Learning models

The optimised results were obtained using Keras Tuner, as described in Section 4.2.1.3. Once the best hyperparameters were obtained, we retrained the model using the same train, validation, and test split and recorded the new performance. Surprisingly, we observed inferior performance compared to the classic ML methods presented in Section 5.1.1. One possible cause of this is that the dataset is too small, as deep neural models have been shown to be most effective for larger datasets. Hence, for small-scale settings such as the current one, classic ML models may be sufficient. Interestingly, the models trained with tuned parameters perform worse than the default models. This could be an issue with the optimisation algorithm terminating too early or with the parameters forcing the model to overfit the training data, causing it to perform worse on unseen data.

We also noticed that the un-optimised CNN-based models outperformed the Bi-LSTM-based models, even though CNNs were originally developed for image understanding rather than text. Moreover, despite having more training examples, Sepedi's performance still lags behind Tshivenda. The confusion matrix in Figure 5.3 exposes weaknesses in Sepedi predictions arising from the misclassification of business and economy news with politics and society. It could be that these wrongly predicted articles were about corruption related to Covid-19 funds by politically connected companies, which negatively affected people's livelihoods. This is to be expected as it was revealed in Section 3.4 that the collected articles were biased towards these topics because they were collected during the period of the Covid-19 pandemic.

5.2.2 Multilingual news classification performance

We also investigate the ability of DNN models to represent multiple languages using a shared single embedding space. The results presented here were obtained after aligning monolingual vector spaces developed from Tshivenda and Sepedi texts using VecMap. To determine the



Confusion matrix

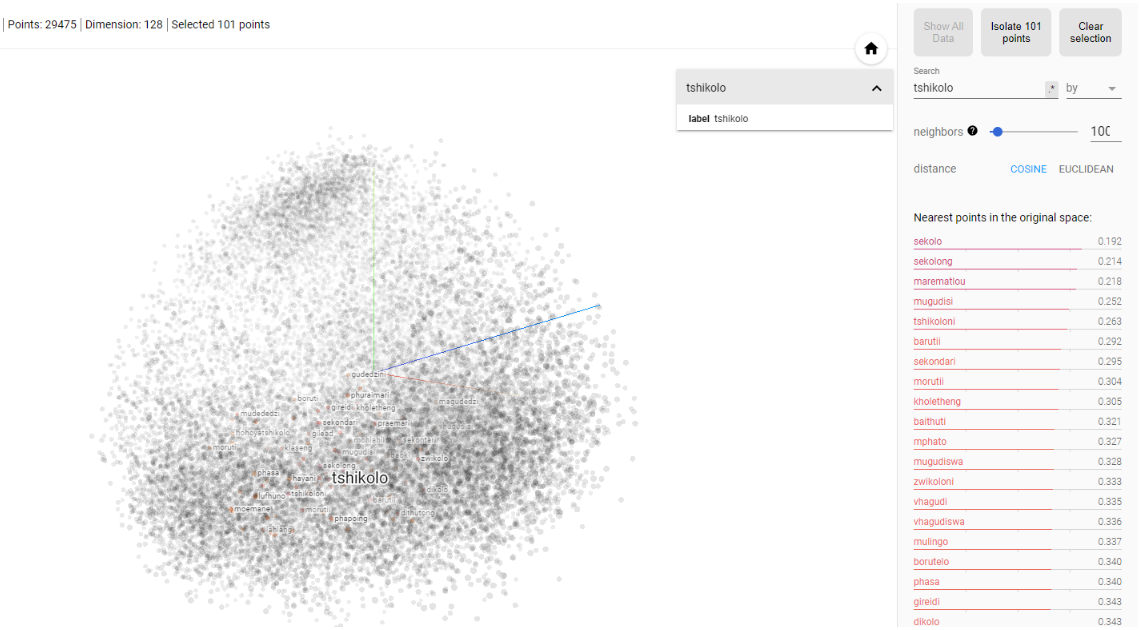
	rime, law and	disaster, accid	onomy, busine	education	health	labour	politics	society	sport
crime, law and	190	6	14	2	4	0	25	16	0
disaster, accid	6	85	4	2	3	0	5	8	0
economy, busine	7	1	177	1	5	6	14	22	0
education	2	1	3	86	4	0	8	15	0
health	4	1	7	7	133	0	9	26	0
labour	1	0	9	0	1	86	1	6	0
politics	21	3	25	2	4	1	147	33	1
society	17	4	20	8	15	2	24	387	3
sport	0	0	1	0	0	0	0	3	96

Figure 5.3: CNN news topic classifier confusion matrix on Sepedi news test set

effectiveness of the alignment, we first retrain the classifier models using the new shared vector space and compare the results to the initial findings presented in Section 5.2.1. Additionally, we evaluate different few-shot cases to determine if we can achieve good performance even with small training examples in the low-resource language(Tshivenda).

A snapshot of the new vector space obtained by aligning vectors using a small bilingual Sepedi to Tshivenda dictionary is shown in Figure 5.4. In this case, the word “Tshikolo” (school) is perfectly aligned with its Sepedi translation “sekolo”. We observe a similar trend for “hayani” (home), which is mapped to “gae”. With these promising results in mind, we proceeded to evaluate whether this would translate to improved news topic classification performance in monolingual and few-shot settings.

The results presented in Table 5.4 are obtained using custom-trained embeddings of varying sizes (128, 300, and 500) generated from both FastText and Word2Vec. Furthermore, the vectors have been projected to the same vector space using VecMap. These static vectors are used to initialise a frozen Embedding layer in the CNN and Bi-LSTM model architectures. Comparing these results to those obtained in Section 5.2.1, we find no significant change in downstream task performance across all dimensions when using multilingual pre-trained vectors. This suggests



Points: 29475 | Dimension: 128 | Selected 101 points

tshikolo

label tshikolo

Search: tshikolo

neighbors: 100

distance: COSINE | EUCLIDEAN

Nearest points in the original space:

sekolo	0.192
sekolong	0.214
marematlou	0.218
mugudisi	0.252
tshikoloni	0.263
barutii	0.292
sekondari	0.295
morutii	0.304
kholetheng	0.305
baithuti	0.321
mphato	0.327
mugudiswa	0.328
zwikoloni	0.333
vhagudi	0.335
vhagudiswa	0.336
mulingo	0.337
borutelo	0.340
phasa	0.340
gireidi	0.343
dikolo	0.343

FIGURE 5.4: Semantic similarity example from the shared embedding space developed using VecMap

Inputs	Word2Vec [%]			FastText [%]		
	Bi-LSTM	CNN	Average	Bi-LSTM	CNN	Average
Mono						
Tshivenda-128	65.8	78	71.9	53.9	76.2	65.05
Tshivenda-300	64.7	77.5	71.1	67.4	76.3	71.85
Tshivenda-500	71.1	**77.8**	74.45	66.4	77.7	72.05
Mono						
Sepedi-128	49	73.9	61.45	60.2	74.5	67.35
Sepedi-300	50.2	73.5	61.85	63.3	74.6	68.95
Sepedi-500	51.3	**74.9**	63.1	65.4	74.6	70
Zero-shot 10						
Tshivenda-128	55	59.9	57.45	57	**60**	58.5
Tshivenda-300	52.8	57.6	55.2	52	54.4	53.2
Tshivenda-500	56.9	56.6	56.75	59.1	56.7	57.9
Zero-shot 100						
Tshivenda-128	66.6	**72.9**	69.75	64.9	72.0	68.45
Tshivenda-300	60.7	72.4	66.55	67	72.2	69.6
Tshivenda-500	64.4	71.3	67.85	67	72.2	69.6

TABLE 5.4: Weighted F1-scores for the news classification task obtained from frozen bilingual word embeddings trained with Word2Vec and FastText

that we are able to project to a multilingual vector space while maintaining the individual performance of each input language.

The F1 scores tend to increase as the number of vector dimensions increases from 128 to 500 in monolingual-settings. Interestingly, we observe the opposite effect for the few-shot cases for both CNN and Bi-LSTM models. Furthermore, we observe that few-shot performance on Tshivenda tends to increase with 10, 50, or 100 examples per label in the training set. With 100 examples, we achieve a test score of approximately 60%, which is promising given the use of just a few hundred Sepedi to Tshivenda word pairs to align the embeddings.

5.2.3 FastText

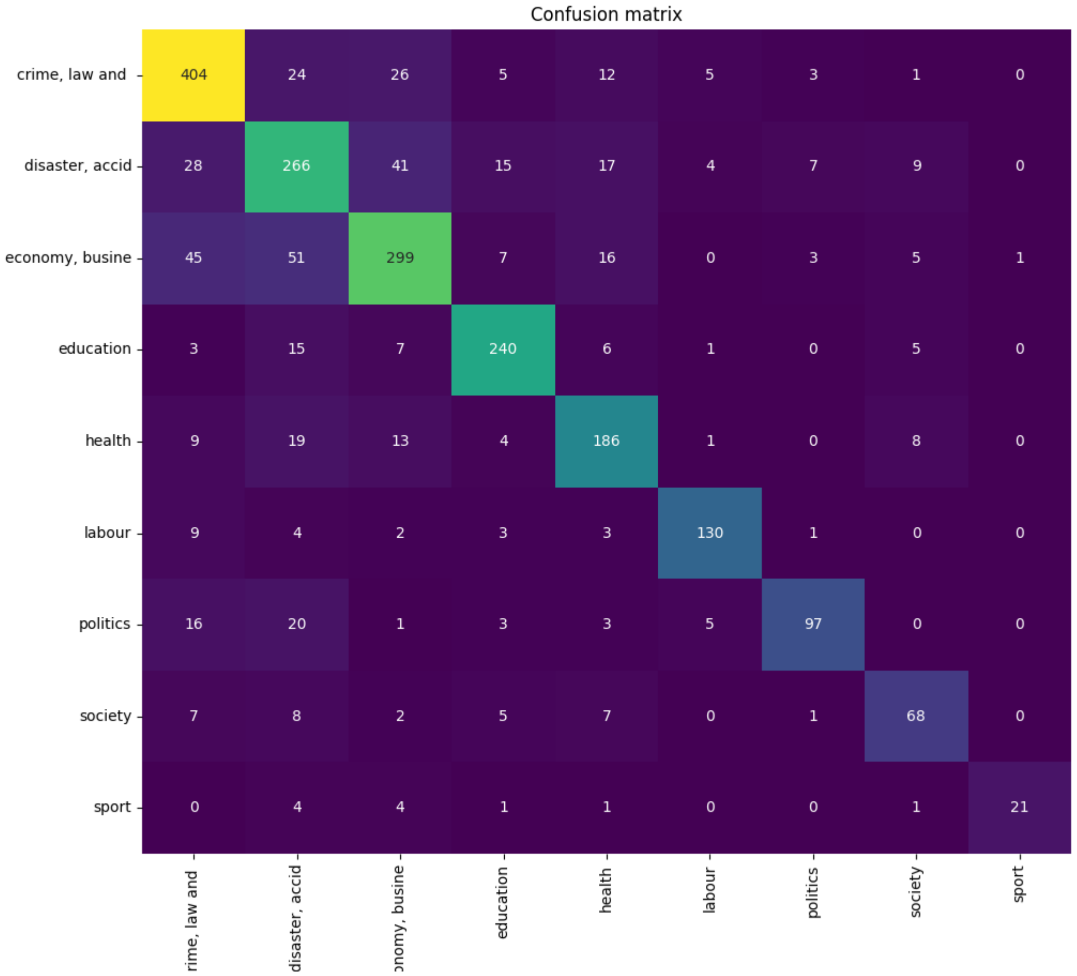
FastText [Bojanowski et al., 2017] is a DNN-based technique used for various natural language processing (NLP) tasks. It learns how to represent words by analysing information about their subwords, otherwise known as morphemes. This approach allows FastText to handle out-of-vocabulary words and morphologically rich languages more effectively than other models. Despite its relatively simple architecture compared to other deep learning models like Bi-LSTM, FastText exhibits impressive performance on NLP tasks like text classification, sentiment analysis, and language modelling. As a result, it is widely used in industry and research and has become a popular alternative to traditional sequence models like recurrent neural networks which are slower to train.

Inputs	**Fast-Text baseline performance** [%]	
	Default	**Optimised**
Tshivenda	75.6	76.4
Sepedi	74.6	72.5
Average	75.1	74.45

Table 5.5: Weighted F1-scores for the news classification task obtained from FastText models

After optimising the hyperparameters for Sepedi, we obtained a slightly worse performance than the original. This could be due to the optimised model overfitting the training data, making it less effective in generalisation. We also noticed that the optimised model has a small dimension of 36 compared to 128 in the default model. Perhaps placing a constraint on the maximum and minimum dimensions could help improve the score, we leave this as a future improvement area. The confusion matrix in Figure 5.5 shows that FastText struggles with distinguishing between crime and business, disasters, and politics.

Based on the observations from the confusion matrices of FastText and Logistic regression models in Figures 5.5 and 5.1 respectively, we notice that the overlap between the categories is minimal in Logistic regression, particularly between crime and society. However, for FastText, we observe a higher overlap between crime and society, and confusion between disasters and various other categories such as business, education, health, and politics. These observations raise questions about the effectiveness of FastText compared to Logistic regression in low-data settings. Another possibility for this behaviour could be issued with labelling, given that the COVID-19 pandemic would have generated a lot of news in the health, government, and business sectors, which could have confused the annotators. The overlap of ”disaster, accident and emergency incidents” with ”education” could be due to incidents where there was damage to learning infrastructure due to weather or protest action. Further investigation may be necessary to determine the underlying causes of these observations.



	rime, law and	disaster, accid	onomy, busine	education	health	labour	politics	society	sport
crime, law and	404	24	26	5	12	5	3	1	0
disaster, accid	28	266	41	15	17	4	7	9	0
economy, busine	45	51	299	7	16	0	3	5	1
education	3	15	7	240	6	1	0	5	0
health	9	19	13	4	186	1	0	8	0
labour	9	4	2	3	3	130	1	0	0
politics	16	20	1	3	3	5	97	0	0
society	7	8	2	5	7	0	1	68	0
sport	0	4	4	1	1	0	0	1	21

Figure 5.5: FastText news topic classifier confusion matrix on Tshivenda news test set

5.3 Pre-trained Language models

In this section, we will present the performance of fine-tuned pretrained language models, considering both few-shot classification performance and monolingual performance for Sepedi and Tshivenda news headlines. We will start by fine-tuning existing state-of-the-art models such as XLMR, Afro-XLMR and AfriBERTa. Ultimately, we will repeat this process on the Zabantu fleet of models which we trained from scratch.

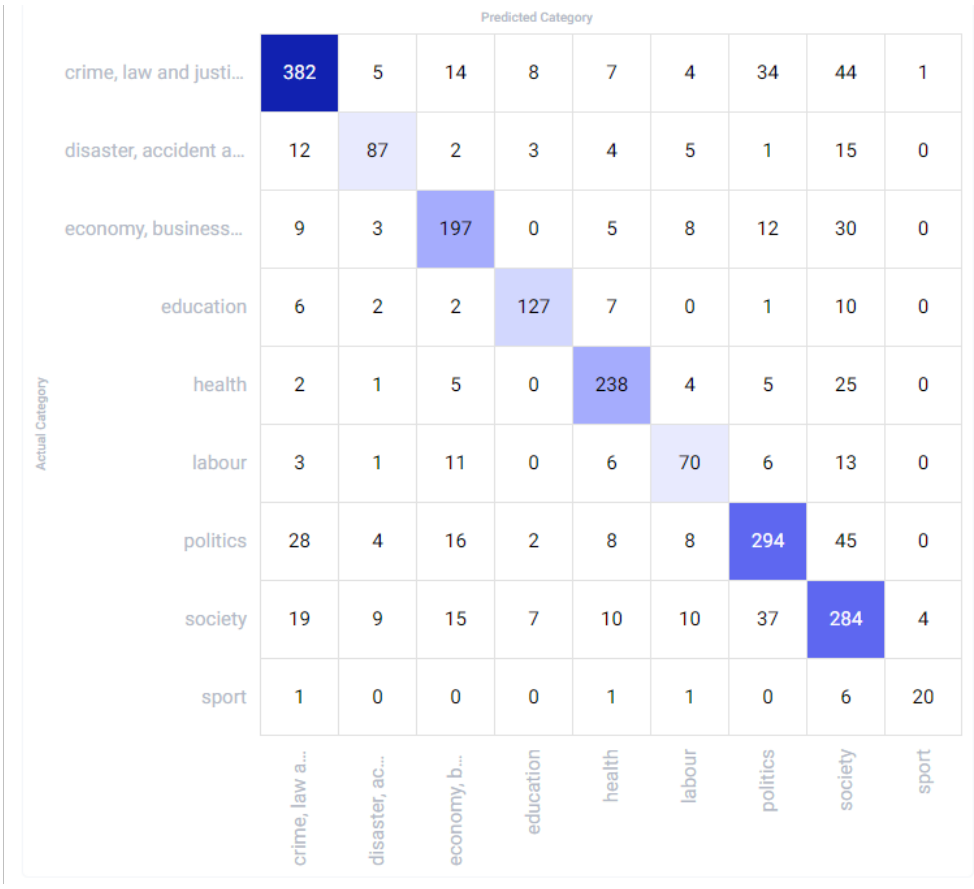
5.3.1 Monolingual news classification performance

We report the monolingual performance of plug-and-play large language models from Hugging Face. All models are based on the XLM-RoBERTa architecture and serve as a good baseline for our custom language models, which will be discussed in the next section. We fine-tune each model on the labelled monolingual news corpus on multiple servers provisioned with Tesla K80, T4, and V100 GPUs for ten epochs. We also implement early stopping to terminate training if the performance does not increase for three consecutive epochs.

Inputs	**Weighted F1-score [%]**				
	XLM-R	**AfriBERTa-base**	**AfriBERTa-large**	**Afro-XLMR-base**	**Average**
Tshivenda	70.6	74.3	**75.2**	71.6	72.93
Sepedi	66	71.4	72.4	**74.1**	70.42
Average	68.3	72.85	73.8	72.85	-

Table 5.6: Weighted F1-scores for the news classification task obtained from fine-tuning open-source language models based on the XLM-Roberta architecture

XLMR generally performs worse than AfriBERTa in classifying both Sepedi and Tshivenda news headlines. Although AfriBERTa was not trained on Tshivenda text, it still produces results on par with deep monolingual learning and classic ML models, which were trained explicitly on Tshivenda data. A technique like Language Adaptive Fine-tuning (LAF) could be adopted on XLMR to improve its performance by adapting it to Tshivenda and Sepedi texts before downstream task fine-tuning. Similarly, Afro-XLMR did not have Tshivenda or Sepedi in the pre-training set [Alabi et al., 2022], yet they still achieved a good F1 score. We also note that the F1 score for Sepedi is higher than Tshivenda on Afro-XLMR compared to AfriBERTa, likely because Afro-XLMR had Sesotho texts in its pre-training set which come from the same language family as Sepedi.



	crime, law a...	disaster, ac...	economy, b...	education	health	labour	politics	society	sport
crime, law and justi...	382	5	14	8	7	4	34	44	1
disaster, accident a...	12	87	2	3	4	5	1	15	0
economy, business...	9	3	197	0	5	8	12	30	0
education	6	2	2	127	7	0	1	10	0
health	2	1	5	0	238	4	5	25	0
labour	3	1	11	0	6	70	6	13	0
politics	28	4	16	2	8	8	294	45	0
society	19	9	15	7	10	10	37	284	4
sport	1	0	0	0	1	1	0	6	20

Figure 5.6: Confusion matrix on Tshivenda news test set on a fine-tuned AfriBERTa-large model

Evaluation of the confusion matrix obtained from AfriBERTA-large predictions on Tshivenda news in Figure 5.6 shows a significant overlap between crime with politics, society, and disaster reports headlines. This is the same overlap observed with other models trained so far. We appear to have a labelling issue that cannot be overcome using advanced models only. Perhaps an independent benchmarking dataset would be helpful to confirm that the trends observed so far are valid. Some of the misclassified headlines are shown in Table 5.7. It is clear from these predictions that some headlines are ambiguous and can fit into multiple news genres.

Text	Actual	Predicted
mukalaha wa eastern cape o mangala a tshi wana zwa uri o nwaliswa sa muthu o no lovhaho muhashoni wa muno (a man from eastern cape was alarmed to discover that he was registered as a diseased man at the department of education)	Society	Crime
duduzane zuma uri vhomcebisi jonas vho vha vha tshi divha nga ha u ri mutangano wavho wo sudzuluselwa saxonwold (duduzane zuma says mr mcebisi jonas was aware that their meeting was rescheduled to saxon-world)	Politics	Crime

Table 5.7: Mis-classified examples in the Tshivenda news test set by AfriBERTa-large classifier

The loss curve in Figure 5.7 shows that training loss gradually decreases with increasing steps as expected. However, the evaluation loss is increasing due to some outliers not being predicted correctly. We assume these to be outliers because although the loss increases over time, the validation accuracy tends to increase. The possible reason for this is rare headlines or just signs of a few mislabelled data. We noticed that this is a recurring trend across all pre-trained models, including XLM-R, AfroXLMR and AfriBERTa. Nonetheless, the models seem to get it right most of the time.

5.3.2 Multilingual news classification performance

Multilingual fine-tuning has been shown to benefit low-resource languages compared to monolingual fine-tuning [Pires et al., 2019]. In this section, we present the results obtained by fine-tuning the various open-source state-of-the-art language models on a classification task using a combination of Sepedi and Tshivenda news headlines. We start with a simple case where we merge the two corpora and fine-tune the model to leverage more labelled points in Sepedi to achieve better performance for both languages. Secondly, we evaluate the zero and few shot cases where we train using the full Sepedi texts and a subset of Tshivenda texts to simulate a truly low-resource setting. We start with a case where we limit each label from Tshivenda to only 10 examples (which we call shots), all the way to about 100 examples per news category. For the evaluation environment, we used a virtual machine hosted on the Google Cloud Platform which was endowed with a NVIDIA Tesla V100 GPU with 16GB Memory and four vCPUs.

Looking at Table 5.11, XLMR performed very well on Tshivenda after joint fine-tuning with Sepedi. However, the performance in Sepedi remains relatively unchanged from the monolingual

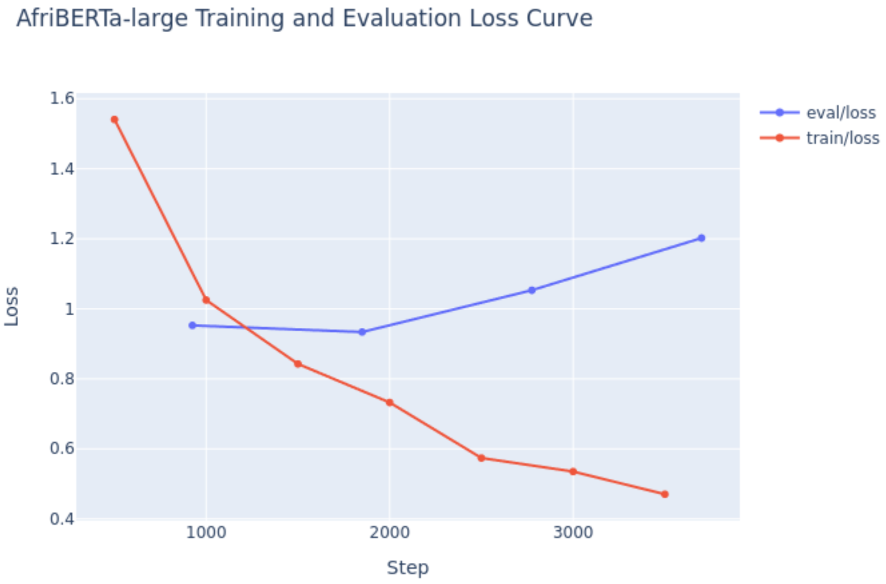


Figure 5.7: AfriBERTa large training loss on Tshivenda news headlines

Inputs	**Weighted F1-score [%] - 10 shots**				
	XLM-R	**AfriBERTa-base**	**AfriBERTa-large**	**Afro-XLMR-base**	**Average**
Tshivenda	6	48	49	50	38.25

Table 5.8: Weighted F1-scores for the *few-shot* News Classification task with 10 shots. Source language = Sepedi, Target language = Tshivenda

Inputs	**Weighted F1-score [%] - 50 shots**				
	XLM-R	**AfriBERTa-base**	**AfriBERTa-large**	**Afro-XLMR-base**	**Average**
Tshivenda	47	60	56	**62**	56.25

Table 5.9: Weighted F1-scores for the *few-shot* News Classification task with 50 shots. Source language = Sepedi, Target language = Tshivenda

Inputs	**Weighted F1-score [%] - 100 shots**				
	XLM-R	**AfriBERTa-base**	**AfriBERTa-large**	**Afro-XLMR-base**	**Average**
Tshivenda	8	62	63	**65**	49.5

Table 5.10: Weighted F1-scores for the *few-shot* News Classification task with 100 shots. Source language = Sepedi, Target language = Tshivenda

Inputs	**Weighted F1-score [%] - Multilingual Finetuning**				
	XLM-R	**AfriBERTa-base**	**AfriBERTa-large**	**Afro-XLMR-base**	**Average**
Tshivenda	70.0	**73.0**	73	**74**	72.5
Sepedi	66.6	**71**	69	**72**	69.65
Average	68.3	72	71	73	

Table 5.11: Weighted F1-scores for the *multilingual fine-tuning* on the News Classification task

score. Similarly, it appears that AfriBERTa performs equally in multilingual fine-tuning setting as the monolingual setting for Tshivenda news.

5.4 Zabantu models

In addition to fine-tuning existing open-source models, we have developed a novel set of language models trained on various combinations of SA Bantu texts. We began with a set of base models that trained for only ten epochs. Then, we continued training for up to 100 epochs, using early stopping to terminate when the training loss no longer decreased. Next, we split each language corpus into training and validation sets with ratios ranging from 0.2 to 0.3. After training, we calculated perplexity on the validation set, first for each language individually and then for all languages collectively. In this section, we will outline the results of language model training and the performance scores achieved by fine-tuning these models for various downstream tasks.

5.4.1 Language model training results

Table 5.12 shows a comprehensive view of the experiment configurations used to train different language models with a subset of Bantu languages spoken in South Africa. We show the intrinsic metrics, including perplexity and loss, used to identify possible good language model candidates which might perform well in cross-lingual news topic classification.

The reported vocabulary sizes exclude two special tokens used in the XLMR architecture that represent the beginning and end of a sentence. These special tokens, known as the start-of-sentence (SOS) and end-of-sentence (EOS) tokens, are added to the input sequences during pre-processing to indicate the start and end of each sentence. During training, each experiment is named using the pattern ”zabantu-[language-code]-[vocabulary-size]k-[tokenisation-method]-[number-of-epochs]”. For example, *zabantu-ven-30k-unigram-10epochs* represents a language model trained on a corpus of Tshivenda sentences using a vocabulary of 30k obtained from Unigram tokenisation and trained for ten epochs. Similarly, *zabantu-sot_ven-50k-bpe-50epochs* represents another experiment using a vocabulary of 50k obtained from byte-pair-encoding (BPE) on a corpus of sentences derived from a combination of Sesotho, Sepedi, Setswana and Tshivenda, trained for 50 epochs.

Based on Table 5.12 we have observed that models trained with BPE tokenisation perform slightly worse than those with Unigram tokenised vocabulary, based on both loss and perplexity results. However, in most cases, we found that the perplexity and loss performance remains comparable when considering the same number of tokens and training epochs. It is worth noting that we were able to train tokenisers with significantly larger vocabularies using BPE, although this does not necessarily guarantee improved performance. We discovered that when the number of tokens exceeded a certain threshold, the performance started to decline, potentially due to the inclusion of tokens with no semantic meaning in relation to the training data, a common problem when using BPE tokenisation Bostrom and Durrett [2020].

5.4.2 Monolingual news classification performance

Table 5.13 shows the results of fine-tuning various Zabantu language models on a news topic classification task. Each row displays the test F1 score for a language along with the properties

Model	**Settings**				**Metrics**	
	#Tokens	**#Examples**	**#Params**	**Epochs**	**Perplexity**	**Loss**
Zabantu-ven	30k	58k	80M	10	34.1	3.53
	30k	58k	80M	50	10.68	2.37
	30k-bpe	58k	80M	10	35.6	3.57
	30k-bpe	58k	80M	50	6.8	1.91
	50k-bpe	58k	96M	10	36.73	3.6
	70k-bpe	58k	110M	10	36.37	3.59
	70k-bpe	58k	110M	20	19.39	2.59
	85k-bpe	58k	123M	50	8.35	2.12
Zabantu-nso	30k	125k	80M	10	22.15	3.09
	30k-bpe	125k	80M	10	24.1	3.18
	50k	125k	96M	10	23.44	3.15
	50k-bpe	125k	96M	10	26.02	3.35
	70k	125k	110M	10	24.62	3.2
	70k-bpe	125k	110M	10	26.67	3.27
	85k-bpe	125k	123M	50	10.08	2.31
Zabantu-nso-ven	30k	189k	80M	10	8.18	6.7
	30k-bpe	189k	80M	10	1212.72	7.1
	50k	189k	96M	10	16.0	2.77
	50k-bpe	189k	96M	10	913.8	6.81
	70k	189k	110M	10	36.3	3.59
	70k-bpe	189k	110M	10	18.73	2.93
	150k-bpe	189k	172M	50	8.98	2.19
Zabantu-sot-ven	30k	479k	80M	10	37.93	3.63
	30k-bpe	479k	80M	50	7.56	2.02
	50k	479k	96M	10	39.2	3.67
	50k-bpe	479k	96M	20	12.0	2.49
	50k-bpe	479k	96M	50	7.98	2.08
	70k	479k	110M	10	14.98	2.71
	70k-bpe	479k	110M	10	1560	7.35
	85k	479k	110M	20	12.01	2.48
	85k-bpe	479k	110M	20	12.91	2.56
	150k	479k	110M	20	12.81	2.54
Zabantu-bantu	30k	1.4M	80M	10	7102.18	8.87
	50k	1.4M	80M	10	9187.65	9.13
	70k	1.4M	110M	10	8884.927	9.09
	250k	1.4M	250M	10	3.47	3.47
	250k-bpe	1.4M	250M	10	112.9	4.72
	250k-bpe	1.4M	250M	50	17.3	2.85

Table 5.12: Training results for Zabantu Language Models

of the fine-tuned language model. We observe that good scores are obtained for the monolingual Zabantu-VEN on Tshivenda and similarly for Zabantu-NSO on Sepedi news. However, although these scores are generally higher, they are still fairly comparable to scores from open-source language models, including AfriBERTa and Afro-XLMR.

5.4.3 Multilingual news classification performance

This section shares the results obtained from cross-lingual news topic classification using Sepedi as the source language and Tshivenda as the target language. We start with 10 examples per

Model	#Tokens	#Epochs	Tshivenda news	Sepedi news
Zabantu-ven	30k	10	73	-
	30k	20	74	-
	30k-bpe	50	76	-
	50k-bpe	100	74	-
	70k-bpe	10	76	-
	70k-bpe	20	76	-
	85k-bpe	50	75	-
Zabantu-nso	30k	10	-	73
	30k-bpe	10	-	71
	50k	10	-	71
	50k-bpe	10	-	73
	70k	10	-	72
	85k-bpe	50	-	68
Zabantu-nso-ven	30k	10	21.2	26.1
	50k	10	75.9	74.1
	50k-bpe	10	21.2	26.1
	70k	10	77	74.3
	150k-bpe	50	73.9	67.1
Zabantu-sot-ven	30k	10	76	73
	30k-bpe	50	72.4	68.2
	50k	10	21.2	26.1
	50k-bpe	20	74.2	69
	50k-bpe	50	72.4	67.9
	70k	10	59	67.6
Zabantu-bantu	70k	10	5	10
	250k	10	72	67
	250k-bpe	50	75.6	70.6

Table 5.13: Weighted F1-scores[%] from finetuning Zabantu language models on Tshivenda and Sepedi News headlines

category in the target language and gradually increase it to 100. Finally, we test the joint fine-tuning setting.

The results reveal the challenges faced by all models in the few-shot scenario, where only 10 Tshivenda examples per category are available for training. Despite obtaining scores above 50 in Zabantu-SOT+VEN models, there appears to be a consistent struggle across the entire model ensemble, as evidenced in Table 5.14. However, a significant improvement in performance is observed when the number of few-shot examples is increased to 50, and subsequently to 100. This finding holds significant implications for Tshivenda NLP research, as it highlights the potential of leveraging a limited annotated Tshivenda dataset in conjunction with abundant resources available in Sepedi and other related languages from the Sotho language family. Such an approach can enable the development of state-of-the-art NLP models even with constrained annotated datasets, therefore addressing the data scarcity challenge in low-resource language settings.

Our results obtained from Zabantu models are comparable to those of open-source models reported in Table 5.6. For most language model variants, the performance is also proportional to the monolingual setting but noticeably lower in few-shot cases. Our experiments show that we begin to see good performance when we have at least 50 examples for each news category

Zero-shot - 10			
Model	**#Tokens**	**#Epochs**	**Tshivenda news**
Zabantu-nso-ven	30k	10	3
	50k	10	**52**
	70k-bpe	10	45
	150k-bpe	50	33
Zabantu-sot-ven	30k	10	36
	30k-bpe	50	31
	50k	10	5
	50k-bpe	20	34
	50k-bpe	50	33
	70k	10	33
	85k	20	48
	85k-bpe	20	**56**
	150k	20	**56**
Zabantu-bantu	70k	10	5
	250k	10	34
	250k-bpe	50	38

Table 5.14: Weighted F1-scores[%] on Tshivenda and Sepedi News headlines few-shot settings with 10 target examples

Zero-shot - 50			
Model	**#Tokens**	**#Epochs**	**Tshivenda news**
Zabantu-nso-ven	30k	10	2
	50k	10	63
	50k-bpe	10	1
	70k	10	65
	70k-bpe	10	66
Zabantu-sot-ven	30k	10	57
	30k-bpe	50	49
	50k	10	6
	50k-bpe	20	53
	50k-bpe	50	51
	70k	10	49
	85k	20	69
	85k-bpe	20	66
	150k	20	67
Zabantu-bantu	70k	10	6
	250k	10	52
	250k-bpe	50	55

Table 5.15: Weighted F1-scores[%] on Tshivenda and Sepedi News headlines few-shot settings with 50 target examples

in the target language, although this performance is only high enough for Zabantu-NSO+VEN models. Bigger models like Zabantu-SOT+VEN and Zabantu-BANTU seem to work better with at least 100 examples per class. To our surprise, we did not observe higher scores when doing joint fine-tuning. Instead, we got scores slightly lower than the monolingual fine-tuning case. This is the opposite effect of what we expected, as we expected fine-tuning with more data to have a significant positive effect.

Zero-shot - 100			
Model	**#Tokens**	**#Epochs**	**Tshivenda news**
Zabantu-nso-ven	30k	10	4
	50k	10	67
	50k-bpe	10	5
	70k	10	68
	150k-bpe	50	58
Zabantu-sot-ven	30k	10	62
	50k	10	7
	50k-bpe	20	60
	50k-bpe	50	57
	70k	10	59
	85k	20	69
	85k-bpe	20	68
	150k	20	69
Zabantu-bantu	70k	10	7
	250k	10	59
	250k-bpe	50	64

Table 5.16: Weighted F1-scores[%] on Tshivenda and Sepedi News headlines few-shot settings with 100 target examples

Multilingual Finetuning				
Model	**#Tokens**	**#Epochs**	**Tshivenda news**	**Sepedi news**
Zabantu-nso-ven	30k	10	10	16
	50k	10	74	72
	50k-bpe	10	13	11
	70k-bpe	10	74	72
	150k-bpe	50	68	73
Zabantu-sot-ven	30k	10	-	-
	30k-bpe	50	-	-
	50k	10	5	10
	50k-bpe	20	71	66
	50k-bpe	50	71	67
	70k	10	72	68
	85k	20	75	75
	85k-bpe	20	75	75
	150k	20	74	76
Zabantu-bantu	70k	10	5	10
	250k	10	71	68
	250k-bpe	50	72	70

Table 5.17: Weighted F1-scores[%] on Tshivenda and Sepedi News headlines using multilingual fine-tuning

5.5 Summary

In this section, we shared the results of our experiments aimed at improving NLP coverage for Tshivenda. We focused on comparing the performance of basic machine learning models and more advanced pretrained language models to determine the most suitable approach for small data scenarios. The summary of Zabantu models versus pre-trained XLMR models is shown in Figure 5.8. Similarly 5.9 shows the summary of the performance of Zabantu models compared to DNN models while Figure 5.10 compares the final performance in Zabantu models with classic

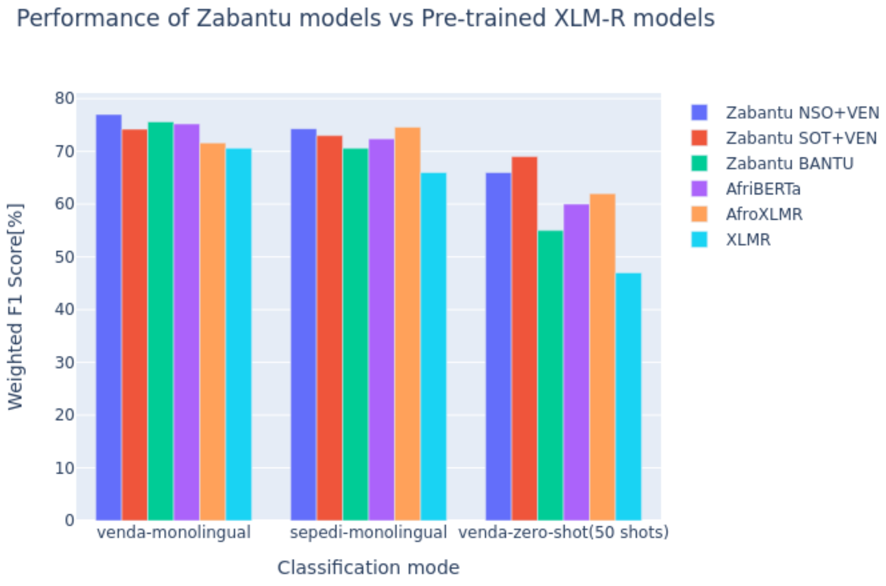


FIGURE 5.8: Summarised performance of Zabantu models vs Pre-trained XLM-R models

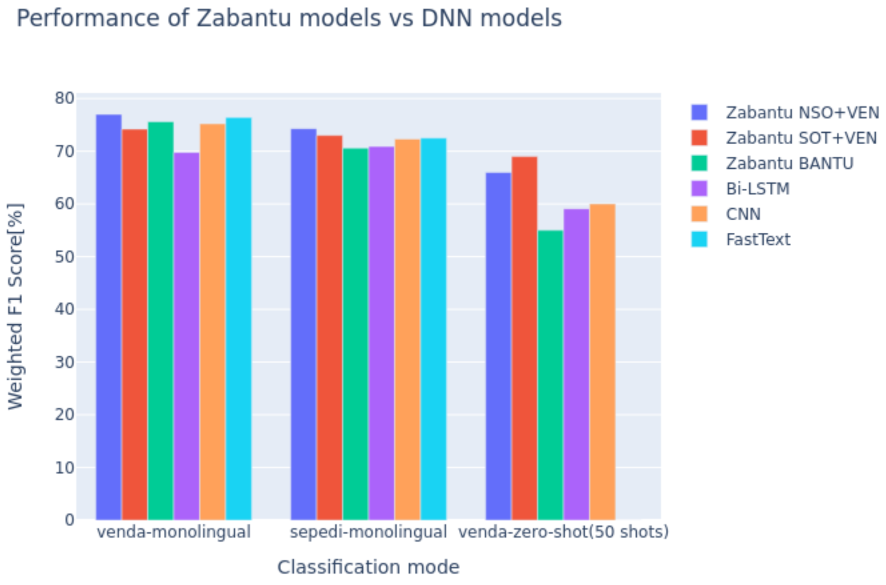


FIGURE 5.9: Summarised performance of Zabantu models vs Deep Neural Network (DNN-based) models

ML models. In Chapter 7, we will provide the implementation of these results and explore the implications of these findings on our research questions.

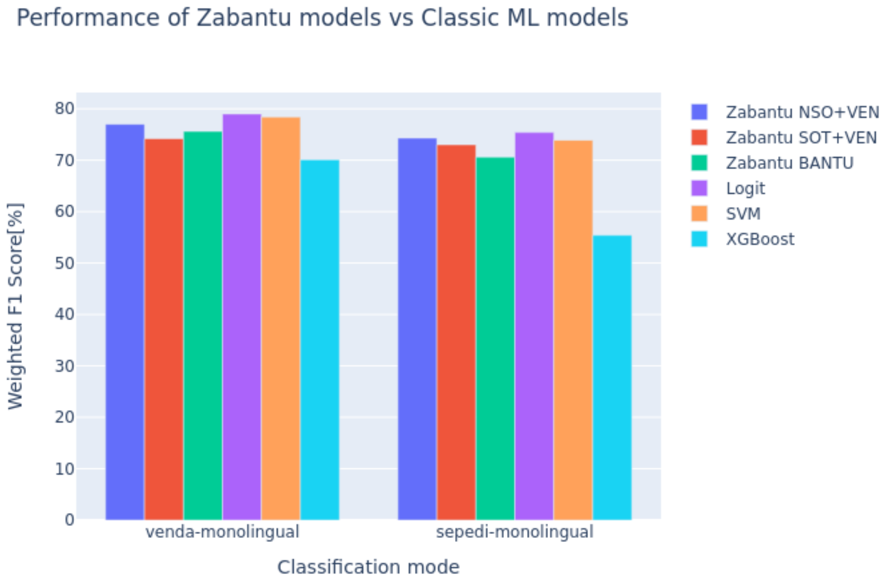


FIGURE 5.10: Summarised performance of Zabantu models vs Classic Machine Learning models

Chapter 6

Predictability, computability, and stability (PCS)

In this section, we will highlight the results of testing the reliability of our top-performing models by subjecting them to various types of adversarial inputs. Initially, we introduce perturbations into the training data and analyse how it affects the model's performance. We present this section alongside the methodology and experiment design sections in Section 3 to emphasise our commitment to transparency throughout the entire life-cycle of the study, including data collection, processing, and modelling stages. We believe that transparency is crucial for building trust and supporting future research efforts in Tshivenda and Sepedi NLP.

Given that one of our primary objectives is to maximise reproducibility, the findings from these observations serve as a significant contribution to the establishment of a robust baseline for future comparisons. This holds particular importance within the context of the developing NLP landscape in South Africa, where independent verification of research results is crucial to avoid overestimating our current capabilities. By ensuring the reliability and accuracy of our findings, we can effectively focus research efforts in the areas that really matter.

We introduced irregularities in the training datasets by randomly introducing spelling errors on Tshivenda news headlines with a probability of 20%. The spelling errors occurred in two forms: missing characters constituted 50% of the errors, while the remaining 50% involved the injection of random characters. The results summarised in Table 6.1 suggest that our models exhibit significant susceptibility to adversarial inputs, as the average performance drops significantly for perturbed inputs. These results highlight the need for further research and development to enhance the models' robustness against adversarial attacks and improve their generalisation capabilities in real-world scenarios. Despite these shortcomings, we still observe promising performance across the different model families with an average F1 score close to 65%.

Unfortunately, it appears that we lose the zero-shot capability with the introduced perturbations, which suggests that the introduced perturbations negatively impact the model's ability to generalise across languages and handle out-of-domain data effectively. While the model demonstrates robustness within the scope of Tshivenda and Sepedi, further investigation is needed

Model	**Mono score**	**50-shot score**	**NEW Mono score**	**NEW 50-shot score**
Zabantu-ven-70kbpe-20	76	-	69	-
Zabantu-nso-ven-70k-10	77	66	63.1	33
Zabantu-sot-ven-150k-20	76	57	70.4	35
Zabantu-bantu-250kbpe-50	75.6	55	63.6	25

Table 6.1: Weighted F1-scores[%] on Tshivenda news topic classification task after random pertubations

to address the challenges and limitations associated with zero-shot capabilities in multilingual settings.

Chapter 7

Discussion

In this section, we will discuss the implications of the findings from Chapter 5. We will also assess whether the research questions were addressed and compare the results with existing related work where possible. Lastly, we will describe potential areas of improvement to assist future researchers in solving the crucial problems to help advance Tshivenda NLP applications.

7.1 Key findings

The research findings reveal interesting observations regarding the performance of news topic classification models in monolingual and cross-lingual settings, as indicated by the F1 scores in Figures 5.8, 5.9 and 5.10. In terms of monolingual performance, classic machine learning models, such as logistic regression and support vector machines, demonstrated strong results. For Tshivenda, logistic regression achieved a monolingual weighted F1 score of 0.79, while for Sepedi, it achieved a score of 0.75 across nine popular news topics. The results were peculiar because we expected Sepedi models to outperform Tshivenda in monolingual Settings. We suspect that the augmented entries in the Sepedi dataset may not have been diverse enough to help the model generalise better.

Surprisingly, text-CNN models performed better than Bi-LSTM models in monolingual and cross-lingual settings. In the monolingual scenario, the best-performing deep learning model, a text-CNN, achieved an F1 score of 0.75 for Tshivenda News and 0.72 for Sepedi. This is an unexpected result as Bi-LSTM models are known to capture sequential dependencies found in text data more effectively than CNNs. We suspect that our choice of the Bi-LSTM network architecture might have caused this discrepancy resulting in over-fitting or under-fitting the datasets at hand. We were also shocked to discover that certain classic ML models outperformed text-CNN and Bi-LSTM models. This suggests that, with limited datasets, classic ML models may be more suitable than DNN models, which typically require extensive training data to achieve better generalisation.

Despite not being pretrained on Tshivenda or Sepedi, open-source pre-trained language models like AfriBERTa, AfroXLMR, and XLMR achieved impressive monolingual news topic classification results. For instance, AfriBERTa-large obtained a weighted F1 test score of 75.4 for Tshivenda and 72.4 for Sepedi, while AfroXLMR achieved 71.6 for Tshivenda and 74.1 for Sepedi. Similarly, XLMR attained scores of 0.7 for Tshivenda and 0.66 for Sepedi. Although not significantly higher scores than classic ML or DNN models, these scores illustrate the capability of leveraging pretrained models to achieve state-of-the-art performance in Tshivenda NLP tasks. This approach eliminates the need to train models from scratch, reducing the data requirements necessary for achieving competitive results.

When training language models from scratch using different combinations of South African Bantu languages, it was observed that languages within the Sotho family, such as Sepedi, Sesotho, and Setswana, yielded similar performances. The model trained on a combination of Tshivenda and Sepedi, named Zabantu-NSO+VEN, achieved the best performance with a test F1 score of 0.77 for Tshivenda and 0.74 for Sepedi. However, training on all South African Bantu languages led to a slight drop in performance. The model trained on all languages achieved a test F1 score of 0.76 for Tshivenda and 0.71 for Sepedi in a monolingual setting. Similarly, this model achieved the lowest test F1 score of 0.55 in few-shot settings with 50 examples per class, while Zabantu-NSO+VEN and Zabantu-SOT+VEN attained 0.66 and 0.69, respectively. This discrepancy suggests that Tshivenda may be less closely related to languages outside the Sotho family, necessitating further investigation to confirm this observation.

We, therefore, make the following deductions with regard to our research questions:

1. Is it viable to leverage high NLP resources from Sepedi and other popular Bantu languages in South Africa to improve the coverage of Tshivenda in NLP applications?

The results support the viability of leveraging NLP resources from related South African Bantu languages to improve the performance of Tshivenda NLP tasks. Classic ML, deep learning, and pretrained models showed promising performance in monolingual and cross-lingual settings. Despite not being pretrained on Tshivenda or Sepedi, pretrained models such as AfriBERTa, AfroXLMR, and XLMR demonstrated impressive performance which was comparable to the performance obtained form languages trained from scratch using SA languages only. This suggests that we can further leverage the potential of other widely spoken Bantu languages like Swahili and Shona, which share common origins with Tshivenda, to enhance cross-lingual transfer capabilities.

2. What is the most effective method to develop word representations to maximise few-shot performance between Tshivenda and Sepedi? i.e. monoglot versus polyglot representations

The research compared different approaches to building word representations and their impact on news topic classification performance. Classic ML models performed well on limited Tshivenda datasets, often outperforming deep neural network models based on text-CNN, FastText and Bi-LSTM architectures. The pretrained transformer-based models, despite not being specifically trained on Tshivenda or Sepedi, achieved F1 scores close to Zabantu language models. Even with as few as 50 examples per news category in Tshivenda, we achieved up to 0.6 few-shot weighted F1 score using AfriBERTa. This suggests that leveraging pretrained models with contextual embeddings from transformer architectures effectively maximises few-shot performance. This is also supported by the results obtained using Zabantu language models trained from scratch, where we achieved few-shot F1 score of 0.69 using Zabantu-SOT+VEN which was pre-trained on Tshivenda, Sepedi, Sesotho and Setswana texts.

Therefore, it appears that it is more viable to use exisiting pre-trained models to develop fine-tuned models for specialised Tshivenda tasks. This significantly reduces the need to collect extensive training datasets which is often expensive.

3. Are the current Tshivenda data resources sufficient for training state-of-the-art NLP models?

Despite the limited availability of Tshivenda data resources, this research demonstrates the possibility of developing cutting-edge NLP models for the language. The findings highlight the potential of cross-lingual transfer learning, utilising both static and contextual embeddings from pretrained transformer models. However, to further advance NLP for African languages, it is crucial to create high-quality benchmark datasets encompassing a wide range of NLP tasks, such as machine translation, entity recognition, and other complex language understanding tasks. The absence of such datasets continues to hinder the development of NLP applications for South African languages. To address this challenge, an effective strategy might involve leveraging fine-tuning of pretrained models on small datasets, enabling the augmentation of available training resources and the creation of specialised models tailored to specific tasks in Tshivenda and other South African Bantu languages.

7.2 Hypothesis validation

Our research hypothesis was that we could improve the inadequate coverage of Tshivenda in NLP research by leveraging cross-lingual transfer learning. To achieve this, we used a closely related language, Sepedi, which has more resources than Tshivenda. In Chapter 5, we presented our findings that confirmed the feasibility of this approach. We also discovered that models trained with languages outside of South Africa still performed well due to the commonalities among Bantu languages across Africa. Furthermore, we found that even less advanced models could be beneficial in low-resource scenarios and could aid in developing supplementary capabilities such as Language Identification models. These can help build high-quality datasets, which, in turn, can be used to build more specialised language models using advanced transformer architectures.

We also observe that multilingual fine-tuning often produces better results than monolingual fine-tuning, confirming findings in related studies pertaining to cross-lingual capabilities of large language models [Pires et al., 2019; Conneau et al., 2020]. Furthermore, our observations confirm that even with relatively small data, we can still use large language models to build NLP applications for low-resource languages, as reported by [Ogueji et al., 2021]. This is an important finding as it opens up the possibility of using pre-trained language models to improve the accuracy of natural language processing tasks for low-resource languages without the need for large amounts of input data.

7.3 Implications of the results

Efficacy of cross-lingual transfer with open-source pre-trained language models

Even though Tshivenda was not included in the languages used to train AfriBERTa or Afro-XLMR, we have noticed impressive performance in news classification tasks under different learning scenarios. Furthermore, these models have sometimes outperformed custom models exclusively trained in Tshivenda and related SA Bantu languages. Based on this observation, it may be wise to temporarily halt training new monolingual language models for Tshivenda and instead focus on adapting existing state-of-the-art models using the currently available datasets. This could help us generate higher-quality datasets that can be used for specialised tasks in Tshivenda in the future.

Foundation for future work in Tshivenda NLP applications

This project has achieved an important milestone by establishing a foundation for future studies on improving Tshivenda coverage in NLP applications through cross-lingual transfer learning. Our results indicate that while current NLP tools may not be optimised for Tshivenda, they can still be effective for various applications if adequate seed data is available. We have also made our experiments public to enable further exploration of this topic using more data or advanced training setups.

7.4 Limitations

- Not all evaluation datasets utilised in this study have been assessed by an independent body or other related experiments, as a result these results require further verification to confirm the viability of cross-lingual transfer between Sepedi and Tshivenda.
- It is worth noting that the evaluation tasks(e.g.topic classification) used in our study may have been too trivial, allowing even traditional machine learning models to handle them. Therefore, assuming there is enough data available, we suggest exploring more complex tasks like entailment and question answering to further validate our findings.
- The model(s) used limited datasets for training, which may result in bias against minority communities. For instance, most crime headlines were related to regions like Alexandra and other low-income communities. This could lead the model to wrongly associate Alexandra with crime, even though there are affluent regions with comparable crime rates that were not part of the training dataset.

7.5 Recommendations and Future work

We identify the following crucial areas for future improvement:

- Reducing subjectivity and annotation bias by getting more annotators and following a strict adjudication process to ensure accurate evaluation of benchmarking datasets.

- We also acknowledge that using a multi-class classification approach may not be the most suitable choice for news categorisation based on IPTC topics, as news headlines often cover multiple relevant topics simultaneously. Therefore, it is worth considering the utilisation of a multi-label classification setting, which would allow for the assignment of multiple relevant labels to each headline.

- As proposed by [Duvenhage et al., 2017a], it is also essential to advance supplementary tools like language identifiers, spell checkers, entity recognition, and machine translation models. These tools can aid in developing robust datasets that can be leveraged to create more sophisticated applications, such as instruction-following agents like Chat-GPT.

- The primary challenge in incorporating Tshivenda into the NLP field appears to be data curation and annotation. As previously emphasised by [Marivate, 2020], addressing this challenge will necessitate novel collaborations between academic, governmental, and commercial institutions to produce high-quality and diverse datasets required to build world-class AI tools.

- A more detailed study of the factors influencing downstream performance across South African Bantu languages is required. For example, to investigate if any transfer can occur between languages with different writing styles, such as Nguni languages and Tshivenda.

Chapter 8

Conclusions

This study used an evaluation-based approach to investigate feasible methods to help boost Tshivenda coverage in NLP applications. First, we examined various state-of-the-art representation models based on the XLM-RoBERTa architecture to produce contextualised embeddings. We then compared these to classic approaches that rely on global word representations such as Word2Vec and TFIDF. We also explored tokenisation techniques, including Byte-Pair Encoding (BPE), and Unigram to support the rich morphology of Tshivenda and other South African Bantu languages. Finally, we evaluated the quality of the different embedding techniques on a short text classification task using a new dataset of news headlines collected from Tshivenda and Sepedi local radio stations.

Empirical results show that classic ML models work well for small datasets on topic classification in the monolingual case. While deep neural network models perform well, their higher computational requirements make them less suitable for small datasets where ML models are faster to train and produce similar results. Pre-trained language models like AfriBERTa, and AfroXLMR demonstrate exceptional performance in multilingual scenarios, even when not explicitly trained on Tshivenda or Sepedi texts. To our surprise, we have discovered that the original XLMR model, which had previously faced criticism for its inadequate representation of African languages within its training corpus, exhibits respectable monolingual performance for both Tshivenda and Sepedi texts.

From the Zabantu models, our highest performing model was trained bilingually on Tshivenda and Sepedi texts. It achieved an impressive 77% weighted F1 score on previously unseen news headlines in Tshivenda and 74% in Sepedi. This performance slightly surpasses pre-trained models such as AfriBERTA and AFRO-XLMR, which attained maximum scores of 75.2% and 74% for Tshivenda and Sepedi, respectively. These findings substantiate previous observations that pre-training language models on smaller yet related datasets can effectively enhance the performance in low-resource scenarios [Ogueji et al., 2021]. However, we believe there is still potential for further improvement by employing techniques such as parameter-efficient fine-tuning on existing advanced large models like XLMR and LLMA. Additionally, we observed encouraging few-shot F1 scores, reaching approximately 70% for Tshivenda in the Zabantu models with as few as 50 examples per news category. Pre-trained open-source models demonstrated a similar trend, averaging around 60% in few-shot news topic classification with Tshivenda as the target language and Sepedi as the source language.

From our findings, most existing NLP tools can already be used for Tshivenda. Using cross-lingual embeddings and few-shot learning, we observe that commendable performance can be achieved even for small datasets, provided we have a larger dataset of a closely related language. However, despite recent efforts to develop NLP resources for Bantu languages, we notice a significant gap in the availability of high-quality benchmarking datasets for South African languages, indicating a need to curate data from different fragmented sources.

We hereby release our newly curated news headlines topic classification dataset to the public, which will serve as a valuable contribution to the existing benchmark datasets for African languages. In addition, we provide our trained models as baselines on the HuggingFace platform, enabling researchers to leverage and build upon our work in future investigations. We expect that the methodologies employed in this study will inspire further advancements in research, ultimately bridging the gap in NLP tool capabilities for South African languages. The dataset and the model cards of the released artefacts are available in the appendix section of this report.

Bibliography

[Duvenhage et al., 2017a] Bernardt Duvenhage, Mfundo Ntini, and Phala Ramonyai. Improved text language identification for the South African languages. In *2017 Pattern Recognition Association of South Africa and Robotics and Mechatronics (PRASA-RobMech)*, pages 214–218. IEEE, 2017a.

[Lephoko, 2021] Kgothatso Lephoko. Natural language processing in the context of indigenous South Aafrican languages, 06 2021.

[Marivate, 2020] Vukosi Marivate. Why african natural language processing now? a view from South Africa# africanlp, 2020.

[Finlayson, 1987] Rosalie Finlayson. Southern-bantu origins. *null*, 7(2):50–57, 1987. URL `https://doi.org/10.1080/02572117.1987.10586684`. doi: 10.1080/02572117.1987.10586684.

[Hellen, 2018] Tlou Prosper Hellen. An analysis of how language contact influenced changes in address norms: The case of Tshivenda in Zimbabwe. *The Journal of Pan-African Studies*, 12:32, 2018.

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

[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. URL `https://aclanthology.org/2021.mrl-1.11`.

[Artetxe et al., 2017] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1042. URL `https://aclanthology.org/P17-1042`.

[Pikuliak et al., 2021] Matúš Pikuliak, Marián Šimko, and Mária Bieliková. Cross-lingual learning for text processing: A survey. *Expert Systems with Applications*, 165:113765, 2021. URL `https://www.sciencedirect.com/science/article/pii/S0957417420305893`. ID: 271506.

[Schuster et al., 2019] Sebastian Schuster, Sonal Gupta, Rushin Shah, and Mike Lewis. Cross-lingual transfer learning for multilingual task oriented dialog. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3795–3805, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1380. URL `https://aclanthology.org/N19-1380`.

[Khalid et al., 2021] Usama Khalid, Mirza Omer Beg, and Muhammad Umair Arshad. Rubert:a bilingual roman Urdu bert using cross lingual transfer learning. *CoRR*, abs/2102.11278, 2021. URL `https://arxiv.org/abs/2102.11278`.

[Glavaš et al., 2017] Goran Glavaš, Federico Nanni, and Simone Paolo Ponzetto. Cross-lingual classification of topics in political texts. In *Proceedings of the Second Workshop on NLP and Computational Social Science*, pages 42–46, Vancouver, Canada, August 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-2906. URL `https://aclanthology.org/W17-2906`.

[Mikolov et al., 2013a] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013a.

[Ruder et al., 2019a] Sebastian Ruder, Ivan Vulić, and Anders Søgaard. A survey of cross-lingual word embedding models. *J. Artif. Int. Res.*, 65(1):569–630, may 2019a. ISSN 1076-9757. doi: 10.1613/jair.1.11640. URL `https://doi.org/10.1613/jair.1.11640`.

[Conneau et al., 2017] Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. Word translation without parallel data. *arXiv preprint arXiv:1710.04087*, 2017.

[Doval et al., 2018] Yerai Doval, Jose Camacho-Collados, Luis Espinosa-Anke, and Steven Schockaert. Improving cross-lingual word embeddings by meeting in the middle. In *Proceedings of EMNLP*. Association for Computational Linguistics, 2018.

[Sannigrahi and Read, 2022] Sonal Sannigrahi and Jesse Read. Isomorphic cross-lingual embeddings for low-resource languages. In *Proceedings of the 7th Workshop on Representation Learning for NLP*, pages 133–142, 2022.

[Duong et al., 2016] Long Duong, Hiroshi Kanayama, Tengfei Ma, Steven Bird, and Trevor Cohn. Learning crosslingual word embeddings without bilingual corpora. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1285–1295, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1136. URL `https://aclanthology.org/D16-1136`.

[Mishra and Viradiya, 2019] Mridul K Mishra and Jaydeep Viradiya. Survey of sentence embedding methods. *International Journal of Applied Science and Computations*, 6(3):592–592, 2019.

[Azunre et al., 2021] Paul Azunre, Salomey Osei, Salomey Addo, Lawrence Asamoah Adu-Gyamfi, Stephen Moore, Bernard Adabankah, Bernard Opoku, Clara Asare-Nyarko, Samuel Nyarko, Cynthia Amoaba, et al. Contextual text embeddings for twi. *arXiv e-prints*, pages arXiv–2103, 2021.

[Chen et al., 2015] Xinxiong Chen, Lei Xu, Zhiyuan Liu, Maosong Sun, and Huanbo Luan. Joint learning of character and word embeddings. In *Twenty-fourth international joint conference on artificial intelligence*, 2015.

[Mesham et al., 2021] Stuart Mesham, Luc Hayward, Jared Shapiro, and Jan Buys. Low-resource language modelling of South Aafrican languages. Apr 1, 2021. URL https://arxiv.org/abs/2104.00772.

[Schuster and Nakajima, 2012] Mike Schuster and Kaisuke Nakajima. Japanese and korean voice search. *2012 ieee international conference on acoustics, speech and signal processing (icassp)*, 2012-march:5149–5152, 2012.

[Bojanowski et al., 2017] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5:135–146, 2017.

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

[Peters et al., 2018] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1202. URL https://aclanthology.org/N18-1202.

[Radford et al., 2018] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018.

[Liu et al., 2020] Qi Liu, Matt J. Kusner, and Phil Blunsom. A survey on contextual embeddings, 2020. URL https://arxiv.org/abs/2003.07278.

[McCann et al., 2017] Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. Learned in translation: Contextualized word vectors. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/20c86a628232a67e7bd46f76fba7ce12-Paper.pdf.

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

[Wu and Dredze, 2020] Shijie Wu and Mark Dredze. Are all languages created equal in multilingual BERT? In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 120–130, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.repl4nlp-1.16. URL https://aclanthology.org/2020.repl4nlp-1.16.

[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, 2021.

[Hedderich et al., 2020] Michael A Hedderich, David Adelani, Dawei Zhu, Jesujoba Alabi, Udia Markus, and Dietrich Klakow. Transfer learning and distant supervision for multilingual transformer models: A study on african languages. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2580–2591, 2020.

[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`.

[Ruder et al., 2019b] Sebastian Ruder, Anders Søgaard, and Ivan Vulić. Unsupervised cross-lingual representation learning. In *Proceedings of ACL 2019, Tutorial Abstracts*, pages 31–38, 2019b.

[Makgatho et al., 2021] Mack Makgatho, Vukosi Marivate, Tshephisho Sefara, and Valencia Wagner. Training cross-lingual embeddings for Setswana and Sepedi. *CoRR*, abs/2111.06230, 2021. URL `https://arxiv.org/abs/2111.06230`.

[Finkelstein et al., 2001] Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414, 2001.

[Pan et al., 2021] Lin Pan, Chung-Wei Hang, Haode Qi, Abhishek Shah, Saloni Potdar, and Mo Yu. Multilingual bert post-pretraining alignment. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 210–219, 2021.

[Huang et al., 2021] Kuan-Hao Huang, Wasi Uddin Ahmad, Nanyun Peng, and Kai-Wei Chang. Improving zero-shot cross-lingual transfer learning via robust training. *CoRR*, abs/2104.08645, 2021. URL `https://arxiv.org/abs/2104.08645`.

[Ethayarajh, 2019] Kawin Ethayarajh. How contextual are contextualized word representations. *Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings. arXiv [cs. CL]*, 2019.

[Zhang et al., 2021] Jinpeng Zhang, Baijun Ji, Nini Xiao, Xiangyu Duan, Min Zhang, Yangbin Shi, and Weihua Luo. Combining static word embeddings and contextual representations for bilingual lexicon induction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2943–2955, 2021.

[Liu et al., 2019] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[Ogueji et al., 2022] Kelechi Ogueji, Orevaoghene Ahia, Gbemileke Onilude, Sebastian Gehrmann, Sara Hooker, and Julia Kreutzer. Intriguing properties of compression on multilingual models. *arXiv preprint arXiv: Arxiv-2211.02738*, 2022.

[Cruz et al., 2020] Jan Cruz, Jose Kristian Resabal, James Lin, Dan Velasco, and Charibeth Cheng. Exploiting news article structure for automatic corpus generation of entailment datasets. 10 2020.

[Cruz and Cheng, 2019] Jan Christian Blaise Cruz and Charibeth Cheng. Evaluating language model finetuning techniques for low-resource languages. *arXiv preprint arXiv:1907.00409*, 2019.

[Conneau et al., 2018] Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. Xnli: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2018.

[Marivate et al., 2020] Vukosi Marivate, Tshephisho Sefara, Vongani Chabalala, Keamogetswe Makhaya, Tumisho Mokgonyane, Rethabile Mokoena, and Abiodun Modupe. Investigating an approach for low resource language dataset creation, curation and classification: Setswana and Sepedi. In *Proceedings of the first workshop on Resources for African Indigenous Languages*, pages 15–20, Marseille, France, May 2020. European Language Resources Association (ELRA). ISBN 979-10-95546-60-3. URL https://aclanthology.org/2020.rail-1.3.

[Mashamaite, 2010] Kwena Mashamaite. The compilation of bilingual dictionaries between african languages in South Africa: The case of northern sotho and Tshivenda*. *Lexikos*, 11, 02 2010. doi: 10.4314/lex.v11i1.51301.

[Orife et al., 2020] Iroro Orife, Julia Kreutzer, Blessing Sibanda, Daniel Whitenack, Kathleen Siminyu, Laura Martinus, Jamiil Toure Ali, Jade Abbott, Vukosi Marivate, Salomon Kabongo, et al. Masakhane–machine translation for africa. *arXiv preprint arXiv:2003.11529*, 2020.

[Barnard et al., 2014] Etienne Barnard, Marelie H Davel, Charl van Heerden, Febe De Wet, and Jaco Badenhorst. The nchlt speech corpus of the South Aafrican languages. Workshop Spoken Language Technologies for Under-resourced Languages (SLTU), 2014.

[Christodouloupoulos and Steedman, 2015] Christos Christodouloupoulos and Mark Steedman. A massively parallel corpus: the bible in 100 languages. *Language resources and evaluation*, 49(2):375–395, 2015.

[Duvenhage et al., 2017b] Bernardt Duvenhage, Mfundo Ntini, and Phala Ramonyai. Improved text language identification for the south african languages. In *2017 Pattern Recognition Association of South Africa and Robotics and Mechatronics (PRASA-RobMech)*, pages 214–218. IEEE, 2017b.

[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, 2020.

[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.

[Oladipo et al., 2022] Akintunde Oladipo, Odunayo Ogundepo, Kelechi Ogueji, and Jimmy Lin. An exploration of vocabulary size and transfer effects in multilingual language models for african languages. In *3rd Workshop on African Natural Language Processing*, 2022.

[Hovy and Spruit, 2016] Dirk Hovy and Shannon L Spruit. The social impact of natural language processing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 591–598, 2016.

[Nyoni and Bassett, 2021] Evander Nyoni and Bruce A. Bassett. Apr 1, 2021. URL https://arxiv.org/abs/2104.00366.

[Borland, 1986] C. H. Borland. Internal relationships in southern bantu. *null*, 6(4): 139–141, 1986. URL https://doi.org/10.1080/02572117.1986.10586665. doi: 10.1080/02572117.1986.10586665.

[Sokal, 1966] Robert R Sokal. Numerical taxonomy. *Scientific American*, 215(6):106–117, 1966.

[Dryer and Haspelmath, 2013] Matthew S. Dryer and Martin Haspelmath, editors. *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig, 2013. URL https://wals.info/.

[Tikeng et al., 2021] Pascal Tikeng, Brice Nanda, and James Assiene. On the use of linguistic similarities to improve neural machine translation for african languages. 2021.

[Lee et al., 2021] Chanhee Lee, Kisu Yang, Taesun Whang, Chanjun Park, Andrew Matteson, and Heuiseok Lim. Exploring the data efficiency of cross-lingual post-training in pretrained language models. *Applied Sciences*, 11(5), 2021. ISSN 2076-3417. URL https://www.mdpi.com/2076-3417/11/5/1974.

[Rust et al., 2022] Phillip Rust, Jonas F. Lotz, Emanuele Bugliarello, Elizabeth Salesky, Miryam de Lhoneux, and Desmond Elliott. Language modelling with pixels. *arXiv preprint arXiv: Arxiv-2207.06991*, 2022.

[Dhamecha et al., 2021] Tejas Indulal Dhamecha, Rudra Murthy V, Samarth Bharadwaj, Karthik Sankaranarayanan, and Pushpak Bhattacharyya. Role of language relatedness in multilingual fine-tuning of language models: A case study in indo-aryan languages. *arXiv preprint arXiv:2109.10534*, 2021.

[Eiselen and Puttkammer, 2014] Roald Eiselen and Martin J. Puttkammer. Developing text resources for ten South Aafrican languages. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3698–3703, Reykjavik, Iceland, May 2014. European Language Resources Association (ELRA).

[Team et al., 2022] NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. No language left behind: Scaling human-centered machine translation. 2022.

[Goyal et al., 2021] Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. The flores-101 evaluation benchmark for low-resource and multilingual machine translation. 2021.

[Guzmán et al., 2019] Francisco Guzmán, Peng-Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc'Aurelio Ranzato. The flores evaluation datasets for low-resource machine translation: Nepali–english and sinhala–english. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6098–6111, 2019.

[Wenzek et al., 2020] Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Édouard Grave. Ccnet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 4003–4012, 2020.

[Aharoni et al., 2019] Roee Aharoni, Melvin Johnson, and Orhan Firat. Massively multilingual neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019.

[Tiedemann, 2012] Jörg Tiedemann. Parallel data, tools and interfaces in opus. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC'2012)*, 2012.

[Zhang et al., 2020] Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. Improving massively multilingual neural machine translation and zero-shot translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020.

[Goldhahn et al., 2012] Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. Building large monolingual dictionaries at the leipzig corpora collection: From 100 to 200 languages. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, 2012.

[Marivate et al., 2023] Vukosi Marivate, Daniel Njini, Andani Madodonga, Richard Lastrucci, and Jenalea Dzingirai, Isheanesu Rajab. The vuk'uzenzele South Aafrican multilingual corpus, February 2023. URL `https://doi.org/10.5281/zenodo.7598539`.

[Tkachenko et al., 2020-2022] Maxim Tkachenko, Mikhail Malyuk, Andrey Holmanyuk, and Nikolai Liubimov. Label Studio: Data labeling software, 2020-2022. URL https://github.com/heartexlabs/label-studio. Open source software available from https://github.com/heartexlabs/label-studio.

[Artstein, 2017] Ron Artstein. Inter-annotator agreement. In *Handbook of Linguistic Annotation*. Springer, Dordrecht, 2017. doi: 10.1007/978-94-024-0881-2_11.

[Nakayama et al., 2018] Hiroki Nakayama, Takahiro Kubo, Junya Kamura, Yasufumi Taniguchi, and Xu Liang. doccano: Text annotation tool for human, 2018. URL https://github.com/doccano/doccano. Software available from https://github.com/doccano/doccano.

[Pedregosa et al., 2011] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[Bird et al., 2009] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.", 2009.

[Gage, 1994] Douglas W Gage. A new algorithm for data compression. *The C Users Journal*, 12(2): 23–32, 1994.

[Sennrich et al., 2015] Rico Sennrich, B. Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. *Annual Meeting Of The Association For Computational Linguistics*, 2015. doi: 10.18653/v1/P16-1162.

[Abadi et al., 2015] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL https://www.tensorflow.org/. Software available from tensorflow.org.

[Paszke et al., 2019] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[Bostrom and Durrett, 2020] Kaj Bostrom and G. Durrett. Byte pair encoding is suboptimal for language model pretraining. *FINDINGS*, 2020. doi: 10.18653/v1/2020.findings-emnlp.414.

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

[Bosch and Griesel, 2018] Sonja Bosch and Marissa Griesel. African Wordnet: facilitating language learning in African languages. In *Proceedings of the 9th Global Wordnet Conference*, pages 306–313, Nanyang Technological University (NTU), Singapore, January 2018. Global Wordnet Association. URL https://aclanthology.org/2018.gwc-1.36.

[Mikolov et al., 2013b] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013b.

[Meyer, 2016] David Meyer. How exactly does word2vec work?, 2016. URL https://www.semanticscholar.org/paper/How-exactly-does-word-2-vec-work-Meyer/49edbe35390224dc0c19aefe4eb28312e70b7e79.

[Bojanowski et al., 2016] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*, 2016.

[Aghajanyan et al., 2022] Anna Aghajanyan, Baolin Huang, Carl Ross, Vladimir Karpukhin, Haoran Xu, Nitin Goyal, ..., and Luke Zettlemoyer. Cm3: A causal masked multimodal model of the internet. *arXiv preprint arXiv:2201.07520*, 2022.

[Devlin et al., 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019. doi: 10.18653/v1/n19-1423. URL https://doi.org/10.18653/v1/n19-1423.

[Chen et al., 2008] Stanley F Chen, Douglas Beeferman, and Roni Rosenfeld. Evaluation Metrics For Language Models. 1 2008. doi: 10.1184/R1/6605324.v1. URL https://kilthub.cmu.edu/articles/journal_contribution/Evaluation_Metrics_For_Language_Models/6605324.

[Hochreiter and Schmidhuber, 1997] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735.

[Joulin et al., 2017] Armand Joulin, Édouard Grave, Piotr Bojanowski, and Tomáš 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, 2017.

[Dyer et al., 2013] Chris Dyer, Véronique Chahuneau, and Noah A Smith. A simple, fast, and effective reparameterization of ibm model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, 2013.

[de la Iglesia Castro, 2023] David de la Iglesia Castro. Dvc: Data version control - git for data & models, May 2023. URL https://doi.org/10.5281/zenodo.7886036.

[Yadan, 2019] Omry Yadan. Hydra - a framework for elegantly configuring complex applications. Github, 2019. URL https://github.com/facebookresearch/hydra.

[Wolf et al., 2019] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, T. Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. Huggingface's transformers: State-of-the-art natural language processing. *ARXIV.ORG*, 2019.

[Akiba et al., 2019] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. *arXiv preprint arXiv: Arxiv-1907.10902*, 2019.

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

[Lei et al., 2015] Tao Lei, Regina Barzilay, and Tommi Jaakkola. Molding cnns for text: non-linear, non-consecutive convolutions. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1565–1575, 2015.

[Chollet et al., 2015] François Chollet et al. Keras. https://keras.io, 2015.

[Rehurek and Sojka, 2011] Radim Rehurek and Petr Sojka. Gensim–python framework for vector space modelling. *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic*, 3(2), 2011.

[Kudo and Richardson, 2018] Taku Kudo and John Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-2012. URL https://aclanthology.org/D18-2012.

Appendix A

Model card

A.1 Zabantu-XLMR - Multilingual Language Models for South African Languages

A.1.1 Model Overview

This model card provides an overview of the multilingual language models developed for South African languages, with a specific focus on advancing Tshivenda natural language processing (NLP) coverage. Zabantu-XLMR refers to a fleet of models trained on different combinations of South African Bantu languages. These include:

- Zabantu-VEN : A monolingual language model trained on 73k raw sentences in Tshivenda
- Zabantu-NSO : A monolingual language model trained on 179k raw sentences in Sepedi
- Zabantu-NSO+VEN: A bilingual language model trained on 179k raw sentences in Sepedi and 73k sentences in Tshivenda
- Zabantu-SOT+VEN: A multilingual language model trained on 479k raw sentences from Sesotho, Sepedi, Setswana, and Tshivenda
- Zabantu-BANTU: A multilingual language model trained on 1.4M raw sentences from 9 South African Bantu languages

A.1.2 Model Details

- **Model Name:** Zabantu-XLMR
- **Model Version:** 1.0.0
- **Model Architecture:** XLM-RoBERTa architecture

- **Model Size:** 80 - 250 million parameters
- **Language Support:** Tshivenda, Nguni languages (Zulu, Xhosa, Swati), Sotho languages (Northern Sotho, Southern Sotho, Setswana), and Xitsonga.

A.1.3 Intended Use

The Zabantu models are intended to be used for various NLP tasks involving Tshivenda and related South African languages. In addition, the model can be fine-tuned on a variety of downstream tasks, such as:

- Text classification and sentiment analysis in Tshivenda and related languages.
- Named Entity Recognition (NER) for identifying entities in Tshivenda text.
- Machine Translation between Tshivenda and other South African languages.
- Cross-lingual document retrieval and question answering.

A.1.4 Performance and Limitations

- **Performance:** The Zabantu models demonstrates promising performance on various NLP tasks, including news topic classification with competitive results compared to similar pre-trained cross-lingual models such as AfriBERTa and AfroXLMR.

Monolingual test F1 scores on News Topic Classification

Weighted F1 [%]	Afriberta-large	Afroxlmr	zabantu-nsoven	zabantu-sotven	zabantu-bantu
nso	71.4	71.6	74.3	69	70.6
ven	74.3	74.1	77	76	75.6

Few-shot(50 shots) test F1 scores on News Topic Classification

Weighted F1 [%]	Afriberta	Afroxlmr	zabantu-nsoven	zabantu-sotven	zabantu-bantu
ven	60	62	66	69	55

- **Limitations:**
  - Although efforts have been made to include a wide range of South African languages, the model's coverage may still be limited for certain dialects. We note that the training set was largely dominated by Setwana and IsiXhosa.

  - We also acknowledge the potential to further improve the model by training it on more data, including additional domains and topics.
  - As with any language model, the generated output should be carefully reviewed and post-processed to ensure accuracy and cultural sensitivity.

A.1.5 Training Data

The models have been trained on a large corpus of text data collected from various sources, including SADiLaR, Leipnets, Flores, CC-100, Opus and various South African government websites. The training data covers a wide range of topics and domains, notably religion, politics, academics and health (mostly Covid-19).

A.1.6 Ethical Considerations

- Privacy: The models do not store or retain personal data or user-specific information during inference.
- Misuse: The models should not be used maliciously or to generate harmful or offensive content. Responsible use of the models is encouraged, adhering to legal and ethical guidelines
- Bias: The training data used for the models may reflect biases in the sources. Evaluating the model's output for fairness and addressing any potential biases during fine-tuning and deployment is recommended.

A.1.7 Conclusion

The Zabantu models provide a valuable resource for advancing Tshivenda NLP coverage and promoting cross-lingual learning techniques for South African languages. They have the potential to enhance various NLP applications, foster linguistic diversity, and contribute to the development of language technologies in the South African context.

Appendix B

Data sheet

B.1 Tshivenda/Sepedi News Topic Classification Dataset - Datasheet

B.1.1 Dataset Information

- Dataset Name: Tshivenda/Sepedi News Topic Classification Dataset
- Purpose: This dataset is designed for news topic classification in the Tshivenda and Sepedi languages.
- Description: The dataset consists of news headlines collected from public social media pages of local radio stations including *Thobela FM, Lesedi FM and Phalaphala FM*

B.1.2 Dataset Overview

- Total Samples: 11k Sepedi and 9k Tshivenda news headlines
- Features: The dataset includes two main columns:
  - Text: The news headlines text in Tshivenda or Sepedi.
  - Label: The best matching news topic genre for each news article.
  - Tags: All genres that may be associated with the news article.

Sample Tshivenda Headlines

text	label	tags
vhadzia misumbedzo vho thivha bada n14 vunduni la nwest	conflict, war and peace	conflict, war and peace #society

text	label	tags
lizhakandila la muzika wa jazz vho jonas gwangwa vho lovha vha na minwaha ya fumalo raru	human-interest	human-interest #arts, culture, entertainment and media
muhasho wa pfunzo vunduni la kwazulu natal wo lugela u vula zwikolo matshelo	education	education

Sample Sepedi Headlines

text	label	tags
economic freedom fighters mo limpopo o bolela gore tsatsi le lengwe le lengwe le swwanetse goba letsatsi la mandela mongwaledi wa mokgahlo mo profenseng jossey buthane o re eff e ya go hlwekisa lefelo la bagolofadi ka univesithing ya limpopo	politics	politics #society
univesithi ya limpopo e re e gopodisisa go amogela batho bao ba dirilego dikgopelo tsa bona ka sebele meraladi e metelele e fokotsegile lehono go se swane le maabaneba bangwe ba batho bao ba bolela gore ba thabisitswe ke ge ba kgonne go dira dikgopelo tsa go ithuta	education	education
sehlopha sa bahlakodiši ka gauteng se nyakana le bana ba babedi bao go dumelwago ba timeletše nakong ya dipula tše maatla maabane dipula tše maatla di sentše ka hammanskraal le moretele pretoria	disaster, accident and emergency incident	disaster, accident and emergency incident#weather

B.1.3 Data Collection Process

- Data Sources: News articles were collected from the social media pages of local radio stations catering to the Tshivenda and Sepedi-speaking communities.
- Preprocessing: The collected data underwent preprocessing steps, such as normalisation to remove diacritics which were not used consistently, removing extra spaces, and removing duplicate entries.

- For some entries, we also had to split the text into individual headlines using bullet points as delimiters.

B.1.4 Dataset Annotation

- Human annotators annotated a subset of 1k headlines for Tshivenda. Where possible, the adjudication process was used to resolve disagreements between annotators and ensure the quality of the annotations. The rest of the dataset was annotated following an active learning process, with the human-annotated dataset providing a baseline for the model to learn from.
- Similarly, for Sepedi 1.5k headlines were annotated by human annotators. The rest of the dataset was translated to English and annotated using zero-shot classification with the help of OpenAI's text-DaVinci-003 model.
- The categories used where based on the codes provided by the IPTC Media Topics taxonomy. The taxonomy is available at https://iptc.org/standards/media-topics/ and the codes are available at https://iptc.org/standards/media-topics/iptc-media-topics-codes/

B.1.5 Data Quality

- Incorrect Labels: The dataset was annotated by human annotators, and where possible, the adjudication process was used to resolve disagreements between annotators and ensure the quality of the annotations. However, there is a possibility of incorrect labels in the dataset especially on ambiguous topics such as human interests, lifestyles, society and entertainment.
- Data Limitations: Because the dataset was collected in the 2021/2022 period in South Africa, it may be skewed towards topics pertaining to Covid-19, state capture, and crime. Additionally, the dataset may not represent the entire spectrum of news topics in the Tshivenda and Sepedi languages.
- Data imbalance: The dataset is dominated by topics like politics, health, crime, disaster and education. As a result the models built with this dataset may struggle to classify news articles into the minority classes such as science and technology, weather or sports.
- Data Augmentation: For Sepedi articles, we used back-translation to augment the dataset with generative AI. However, we were unable to do the same for Tshivenda articles due to the unavailability of advanced translation models for Tshivenda to English.

B.1.6 Data Usage

- Tasks and Experiments: The dataset can be used for news topic classification tasks, where machine learning models are trained and evaluated to classify news articles into predefined news categories.

- Evaluation Metrics: Given the imbalance in the dataset, we recommend using the weighted F1 score as the primary evaluation metric for classification tasks.

B.1.7 Data License and Citation

- License: Creative Commons Attribution-ShareAlike (CC BY-SA)
- Citation: Please cite the following paper when using the dataset:
  - Nemakhavhani, N. (2023). Exploring cross-lingual learning techniques for advancing Tshivenda NLP coverage. Unpublished manuscript, University of Pretoria.

B.1.8 Data Privacy and Ethics

- Privacy Considerations: The dataset does not contain any personally identifiable information or sensitive data.
- Ethical Implications: We acknowledge the possibility of bias in our dataset as we collected data from social media pages over a brief period where Covid-19, state capture, and crime were the dominant topics. This could lead to an over-representation of these topics, making it challenging to generalise the model to other news topics. However, efforts were made to ensure a diverse range of news articles within these dominant topics to mitigate the impact of bias. Additionally, the dataset augmentation techniques employed, such as zero-shot classification and generative AI, aimed to introduce more variety and reduce the bias inherent in the original data collection. Therefore, it is crucial to consider these factors when interpreting and using the dataset for research or practical applications.

B.1.9 Data Distribution

- Availability: The dataset is available for research use and can be downloaded from the HuggingFace Datasets Hub or GitHub.

B.1.10 Contact Information

- For any questions or comments, please contact the dataset authors at [u13075463@tuks.co.za].

Appendix C

Translations

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
modimo (God)	kudu (very much)	ntle (good)	mathomong (in the beginning)	seisemane (English)
motho (a person)	bjo (bro)	letsatsi (day)	mathomong_seisemane (at the beginning_of English)	phetolelo (translation)
dira (enemies)	tsona (them)	bakeng (for)	ngwaga (a year)	oxford (Oxford)
jesu (Jesus)	bohlokwa (important)	feela (only)	kapa (or)	dictionaries (dictionaries)
ng	feta (arrive)	hao (your)	morena (king)	oxford_dictionaries (oxford_dictionaries)
jehofa (Jehova)	dingwe (others)	hau (you)	fihla (arrive)	phetolela_seisemane (translate_english)
mang (who)	nago (I have)	latela (follow)	thoma (start)	phetolela (translate)
baka (cause)	godimo (above)	sebaka (space or time)	tee (tea)	molao (the law)
nako (time)	bontsha (show)	fumana (get)	tloga (leave)	swanetse (should)
modiro (work)	swana (the same)	tattoo (tattoo)	matsatsi (days)	wo

Table C.1: Translations for Sepedi Raw corpus popular topic terms (Auto-Generated with Google Translate)

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
vhathu(people)	muthu(person)	vhathu	khothe(court)	mulayo(law)
muvhuso(government)	tshifhinga(time)	duvha(day)	mulandu(crime)	khethekanyo(segragation)
lushaka(nation)	khumbelo(request)	nwana(child)	mveledziso(development)	tshirema(black)
ndeme(important)	tshumelo(service delivery)	dzhena(enter)	thaidzo(problem)	uyu(this)
mushumo(job)	afrika	bvaho(from)	tsireledzo(protection)	komiti(committee)
shuma(work)	tshipembe	fhedza(finish)	thodisiso(investigation)	mulayotewa(constitution)
shumisa(use)	afrika_tshipembe(south africa)	mbo(was)	maduvha(days)	muhulwane(head or leader)
ndivho(knowledge)	masheleni(funds)	pfa(hear)	mbuelo(reward)	bvelela(emerged)
shumiswa(used)	thendelo(permission)	minwaha(years)	mbudziso(question)	sedzulusa(investigate)
pfanelo(rights)	tshelede(money)	tendelwa(allowed)	fha(give)	tshinwe(another)

Table C.2: Translations for Tshivenda Raw corpus popular topic terms

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
limpopo	kgoro (the door)	maphodisa (the police)	afrika (Africa)	sekolo (school)
badudi (residents)	tsheko (trial)	limpopo	afrika_borwa (south_africa)	thuto (education)
magato (steps)	mengwaga (years)	tikologong (environment)	borwa (south)	south (south)
kgoro (the door)	limpopo (creations)	ntle (good)	anc (anc)	sekolong (at school)
boipelaetso (protest)	feta (more)	mengwaga (years)	mmuso (government)	barutwana (learners)
tikologong (environment)	kgorong (at the door)	monna (man)	tona (minister)	african (african)
magato_boipelaetso (protest action)	wo (wow)	bagononelwa (suspects)	ditshelete (money)	kgoro_thuto (school_course)
ntle (good)	magistrata (magistrate)	bana (children)	lekala (branch)	kgoro (the door)
mmasepala (municipality)	kgoro_tsheko (koro_sheko)	ngoepe (I'm sorry)	ditsela (roads)	south_african (south_african)
bya (by)	kgorong_tsheko (korong_sheko)	fao (there)	merero (projects)	phagamego (height)
tikologo (environment)	molato (crime)	mosadi (a woman)	nageng (in the country)	morutwana (a student)
barutwana (learners)	lapa (hungry)	moatshe (beautiful)	maloko (members)	limpopo

Table C.3: Translations for Sepedi Human Annotated News Corpus popular topic terms(Auto-Generated with Google Translate)

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
mesomo(lessons)	limpopo(creations)	mmuso(government)	leratadima(the sun)	mpsha(new)
kudu(very much)	maphodisa(the police)	basomi(mockers)	bjo(bro)	bana(children)
wo(wow)	badudi(residents)	maphelo(lives)	khutso(silence)	bodumedi(religion)
dira(enemies)	ntle(good)	magato(steps)	bantsi(many)	dimilione(millions)
gape(again)	lefelo(space)	dikgwebo(businesses)	tlago(come)	feta(more)
setshaba(nation)	tsheko(a lawsuit)	fokotsa(reduce)	boemo_leratadima(background_level)	lefaseng(in the world)
swanetse(should)	molato(guilty)	theknolotsi(technology)	letetswe(expected)	ra(ra)
thusa(help)	mengwaga(years)	tikologo(environment)	dula(sit down)	matla(strength)
bohlokwa(important)	tikologong(environment)	melao(rules)	maatla(power)	motho(a person)
soma(laugh)	leo(that)	tlhokego_mesomo(demand_resources)	pego(report)	neng(when)
hwetsa(shout)	fao(there)	sireletsa(protect)	mmalwa(a few)	boela(again)
nako(time)	pula(the rain)	tsebagaditse(identified)	lefase(the world)	bophara(width)
mosomo(a joke)	polokwane(polokwane)	palo(number)	kudu(very much)	dilo(things)
tloga(leave)	kgauswi(soom)	sego(bye)	dithemperetsha(temperatures)	mentsi(many)
leo(that)	feta(more)	mahlale(science)	dutse(sit down)	ditumelo(beliefs)
covid(covid)	bekeng(a week)	thusa(help)	beke(a week)	thata(difficult)
fela(just)	polao(murder)	kimollo(relief)	mafelelong(in the end)	kotsi(danger)
mongwe(someone)	mmoleledi(the narrator)	mekgatlo(organizations)	nago(I have)	ditokelo(rights)
eupsa(yupsa)	monna(man)	dikhamphani(companies)	kgolo(growth)	phela(live)
bangwe(some)	bego(beg)	boletse(said)	bolwetsi(illness)	tumelo(faith)

Table C.4: Translations for Sepedi Machine Annotated News Corpus popular topic terms(Auto-Generated with Google Translate)

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
vhathu(people)	khomishini(commission)	limpopo	vhathu(people)	lihoro(party)
fu	vhathu	vunduni	vhadzulapo(citizens)	anc
madana(thousands)	zuma	minwaha(years)	tshifhinga(period or time)	lihoro_anc
zwigidi(thousands)	muhulwane(elder or leader)	vunduni_limpopo	duvha(day or sun)	khetho(elections)
mbili	zwiito(actions)	lovha(died)	tshipembe	vunduni
thanu(five)	muvhusoni(government)	mapholisa(police)	pfala(heard)	eff
covid	vhutanzi(evidence)	humbulelwa(suspected)	mapholisa(police)	masipala(municipality)
tshipembe	dzhenelela(interference)	fumi	tshumelo(service delivery)	mirado(members)
fumi	mavharivhari(rumours)	khothe(court)	zwavhudi(well)	masipalani
africa	muvhuso(government)	khombo(accident)	afrika	lihoro_eff(EFF party)
ina(has)	dzhenelela_vhathu	rathi(six)	tshimbila(walk)	mivhuso(governments)

Table C.5: Translations for Tshivenda Human Annotated News Corpus popular topic terms

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
vhathu(people)	khothe(court)	ramaphosa	vunduni	vhashumi(workers)
lovha(pass away)	mulandu(crime)	cyril	limpopo	masheleni(funds)
fu	mapholisa	tshipembe	mapholisa(police)	dzangano(party)
madana(hundred)	vhulaha(kill)	shango(world)	vunduni_limpopo(limpopo province)	rannda(rand)
covid	vhahumbulelwa(suspects)	phuresidennde	muhasho(department)	khamphani(company)
tshivhalo(count)	milandu(crimes)	coronavirus	gauteng	muvhuso
zwigidi(thousands)	vunduni(province)	vhulwadze	vhadzulapo	tshifhinga(time)
tshipembe(south)	munna(man)	nyiledzo(lockdown)	natal	nwaha(year)
mbili(two)	senga(testify)	muvhuso(government)	kwazulu	mbili
khombo(accident)	west	phuresidennde_cyril	vhathu	muhulwane(head or leader)
vhulwadze(disease)	north	afrika	vundu(province)	tshelede(money)
africa	muhumbulelwa(suspect)	vhathu	mec	fhungudza(reduce)
thanu(five)	minwaha(years)	vhadzulapo(citizens)	vhuponi(place)	fumi
coronavirus	humbulelwa(suspected)	afrika_tshipembe	doroboni(city)	muofisi(official)
fumi(ten)	senga_khothe(testify in court)	africa	johannesburg	vhuada(corruption)
rathi(six)	farwa(arrested)	covid	vhuendi(traffic)	million

Table C.6: Translations for Tshivenda Machine Annotated News Corpus popular topic terms