

Project Proposal: Post-Training Learning Dynamics of Pretrained Language Models for South African Languages

Siyabonga Madondo
University of Cape Town
Cape Town, South Africa
MDNSIY014@myuct.ac.za

Rakeen Ramdeen
University of Cape Town
Cape Town, South Africa
RMDRAK003@myuct.ac.za

ABSTRACT

Continued pretraining (CPT) has become the dominant approach for adapting pretrained language models (PLMs) to low-resource languages. While learning dynamics have been studied extensively during pretraining from scratch, the process of CPT remains poorly understood. In this proposal, we will investigate the learning dynamics of encoder-only and encoder-decoder models during CPT for isiXhosa, a morphologically complex, low-resource South African language. We will continue pretraining English-only, massively multilingual, and Nguni-specialised base models on isiXhosa text, extract intermediate checkpoints and evaluate them on downstream natural language understanding (NLU) and natural language generation (NLG) tasks. Through this, we aim to provide a clearer understanding of how linguistic knowledge develops during CPT, and how architecture and base model choice influence the rate and order of that development. This study will be one of the first comprehensive analyses of CPT learning dynamics for isiXhosa, offering insight into how prior linguistic knowledge and architectural choice affect adaptation on a low-resource language. Through our study, we aim to provide practical guidance on base model selection and training efficiency for low-resource languages.

KEYWORDS

Natural Language Processing, Learning Dynamics, Large Language Models, Continued Pretraining, AI Interpretability, Low-Resource Languages

1 INTRODUCTION

Pretrained Language Models (PLMs), trained on large amounts of text, have become the foundation of modern natural language processing (NLP), showing strong performance in many generation and understanding tasks [11, 18, 30]. Modern PLMs are based on deep neural networks, specifically multi-layer transformer architectures [33]. These models can be broadly categorised into encoder-only models designed for language understanding [11], encoder-decoder models designed for structured language generation [30], and decoder-only models used primarily for open-ended text generation [28]. They contain many parameters that interact non-linearly across layers, making individual predictions difficult to interpret [6]. It also remains largely unclear how linguistic knowledge is distributed across the network and when it develops during training [24]. Adapting these models to new languages is challenging, especially for morphologically complex¹, low-resource South

African languages where digital text is scarce, and resources are unevenly distributed [2, 23].

PLMs can be adapted to new languages and tasks through continued pretraining (CPT) and fine-tuning [15, 16]. CPT further trains an existing model on new language data, allowing it to specialise for a target language without training from scratch [15], while fine-tuning adapts the model to specific downstream tasks using labelled data [16]. Together, these steps are referred to as post-training. These approaches are especially important for low-resource languages such as isiXhosa, isiZulu, and Sesotho, where limited digital text often makes training models from scratch impractical [2, 23]. CPT has proven to be the most established approach for adapting multilingual models to low-resource languages, consistently resulting in meaningful performance improvements [5, 23]. However, while CPT is effective, it remains largely unclear *why* it works, *when* it is most effective, and *where* it fails.

Current research evaluates CPT only at the end of training, treating the adaptation process as a black box. The study of learning dynamics addresses this by analysing model behaviour at intermediate checkpoints throughout training [24, 31], allowing researchers to observe which types of knowledge are acquired first, which develop later, and how quickly they are learned [9, 17]. However, almost all existing research on learning dynamics has examined models trained from scratch. Studies of pretraining dynamics have shown a consistent hierarchical learning timeline, where models acquire basic syntactic knowledge rapidly, followed by factual and commonsense knowledge, with complex reasoning developing later [9, 17, 24]. Such analyses can also identify when and why learning breaks down, informing more efficient training strategies [17]. However, CPT is fundamentally different from pretraining from scratch. When pretraining from scratch, a model acquires all linguistic knowledge from random initialisation [9]. In CPT, the model already has linguistic knowledge encoded from its initial pretraining, and adaptation involves adjusting existing representations rather than optimising them from scratch. It is therefore unknown whether the hierarchical learning timeline shown during pretraining also occurs during CPT. While recent work has begun to study learning dynamics during CPT for low-resource languages [14], this work examines a context very different from how CPT is typically applied to African languages. The learning dynamics of CPT for South African languages, therefore, remain largely unexplored.

The project we propose investigates the learning dynamics of encoder-only and encoder-decoder models during CPT for the isiXhosa language, comparing English-only, massively multilingual

¹Morphological complexity refers to the degree to which words encode grammatical information through prefixes and suffixes. Languages such as isiXhosa and isiZulu are highly agglutinative, meaning a single written word can express what would require multiple written words in English.

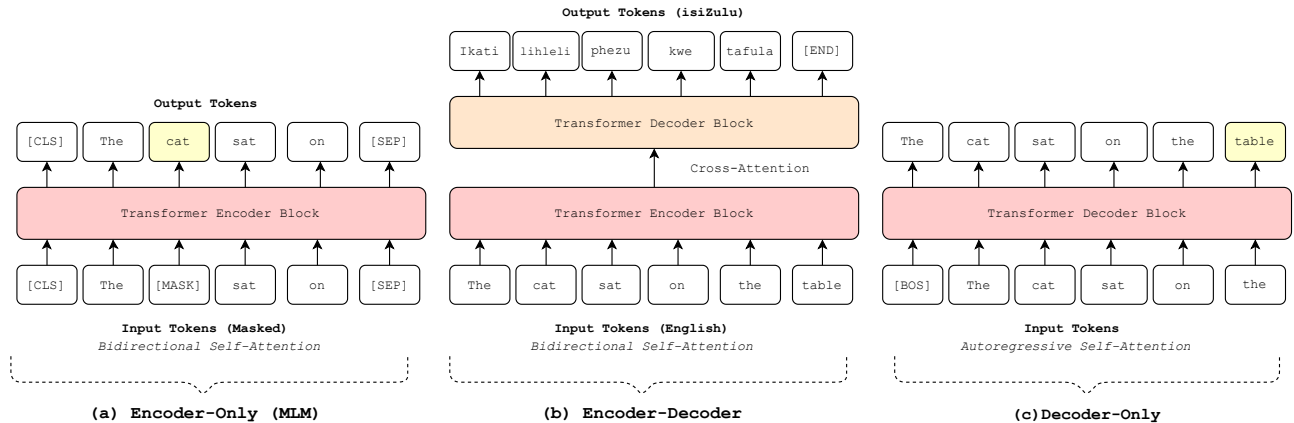


Figure 1: Illustration of encoder-only, encoder-decoder, and decoder-only transformer architectures with example tasks, adapted and simplified from Vaswani et al. [2017].

and Nguni-specialised base models. Specifically, we continue pre-training representative encoder-only and encoder-decoder models, extracting intermediate checkpoints and evaluating them on downstream understanding and generation tasks to observe when linguistic knowledge develops during CPT, and how the choice of base model affects these dynamics. We expect our findings to provide new insights into CPT as an important but poorly understood adaptation strategy for low-resource languages.

2 BACKGROUND

This section introduces the key concepts required for our proposal. We cover pretrained transformer architectures, CPT for low-resource South African languages, and learning dynamics as a method for analysing model behaviour during training.

2.1 Pretrained Transformer Models

This section outlines the key PLM architectures relevant to this project, with their architectures and example tasks illustrated in Figure 1.

2.1.1 Masked Language Models. Masked language models (MLMs) are pretrained transformers used for natural language understanding (NLU) tasks [11, 13]. They are trained using a masked language modelling objective, where a proportion of input tokens are randomly masked, and the model learns to predict them from the surrounding context [11]. This bidirectional training objective allows MLMs to learn rich contextual representations, making them well-suited for tasks such as named entity recognition (NER) and part-of-speech (POS) tagging [11, 23]. These representations support analyses of how linguistic knowledge develops over time. BERT [11] established this pretraining paradigm and demonstrated strong performance across several NLU benchmarks. RoBERTa [18] improves on BERT through an optimised pretraining strategy and more extensive training. XLM-R [10] extends this to the multilingual setting, pretraining on data from 100 languages and enabling cross-lingual transfer, where knowledge from high-resource languages improves performance on low-resource ones without additional

training. XLM-R serves as a base model for several PLMs adapted for South African languages, discussed in Section 2.2.

2.1.2 Encoder-Decoder Models. Encoder-decoder models are designed for structured natural language generation (NLG) tasks, in which text is generated from an input sequence [30]. An encoder processes the full input, and an autoregressive decoder generates the output token by token based on the encoder’s representations [33]. Their structure makes them well-suited for tasks such as machine translation (MT), summarisation, and data-to-text (D2T) generation [23, 30]. T5 [30] established a unified text-to-text framework where every NLP task is framed as converting an input text string to an output text string, demonstrating strong performance across a range of NLG benchmarks. mT5 [36] extends this to the multilingual setting, pretraining on data from 101 languages, including isiXhosa and isiZulu. ByT5 [35] extends mT5 further by operating on raw bytes rather than subword tokens, making it well-suited for morphologically complex languages such as isiXhosa [22, 27]. The T5 family serves as a base model for several encoder-decoder PLMs adapted for South African languages, discussed in Section 2.2.

2.1.3 Decoder-Only Models. Decoder-only models are used primarily for open-ended text generation tasks, such as question answering and code generation [7, 29]. They consist of a single autoregressive decoder that generates text token by token, conditioning each prediction on all preceding tokens [28]. Large-scale training has made decoder-only models increasingly dominant in modern NLP, as seen in the GPT series [7, 29], and more recent models adapted for African languages [8, 37]. Although not the focus of this project, decoder-only models could also be used to study generative CPT learning dynamics in future work.

2.2 Continued Pretraining in NLP

Transfer learning is a standard approach for adapting PLMs to new languages and tasks, where knowledge acquired during pretraining is applied to new contexts [15, 16, 30]. For language adaptation, this typically takes the form of CPT, in which an existing model is further trained on target-language data, allowing it to specialise

Table 1: South African language PLMs adapted using CPT and the languages they cover. Languages include Afrikaans (af), isiNdebele (nr), Sesotho (st), Siswati (ss), Setswana (tn), isiXhosa (xh), isiZulu (zu).

Model	af	nr	ss	st	tn	xh	zu	Authors
Encoder-Only Models								
Afro-XLMR	✓			✓		✓	✓	Alabi et al. [2022]
Nguni-XLMR		✓	✓			✓	✓	Meyer et al. [2024]
Encoder-Decoder Models								
Afri-ByT5	✓			✓		✓	✓	Adelani et al. [2022]
Afri-mT5	✓			✓		✓	✓	Adelani et al. [2022]
Nguni-ByT5		✓	✓			✓	✓	Meyer et al. [2024]
Decoder-Only Models								
AfriqueLLM	✓			✓	✓	✓	✓	Yu et al. [2026]
Lugha-Llama	✓					✓	✓	Buzaaba et al. [2025]

without requiring large amounts of new data [15]. This approach is particularly important for low-resource languages, where limited data often makes training models from scratch impractical [23].

CPT can be applied either monolingually, for a single target language, or multilingually, for multiple languages at once [5]. Monolingual CPT often improves performance for individual languages, whereas multilingual CPT is more resource-efficient, as a single model is used for several languages rather than one model per language, and improves cross-lingual transfer across similar languages [5, 23]. However, including too many unrelated languages can dilute model capacity, a phenomenon known as the curse of multilinguality [10]. Meyer et al. [2024] show that restricting CPT to linguistically similar languages improves cross-lingual transfer and avoids negative interference. PLMs adapted to South African languages using CPT are summarised in Table 1, highlighting the uneven coverage across South African languages.

2.3 Learning Dynamics

Learning dynamics is the study of how models acquire knowledge throughout training. Rather than evaluating a model only after training, it allows researchers to analyse model behaviour at multiple checkpoints to observe which capabilities emerge early, which develop later, and how quickly different types of knowledge are learned [24, 31]. This approach has been used to study how models develop syntactic, semantic, and reasoning abilities during pretraining from scratch [9, 17, 24]. Methods include probing, zero-shot evaluation, internal representation analysis, and fine-tuning on downstream tasks. Of these, fine-tuning-based evaluation is the primary method used in this project, as it provides a direct measure of task competence at each stage of training. Research on learning dynamics during CPT remains limited, with only very recent work beginning to address this [14]. Elhady et al. [2025] adapt a decoder-only model to Basque, Arabic, and Indonesian via CPT and identify a critical period early in training where catastrophic forgetting occurs, with capability dropping before any change is visible in training loss. Whether similar dynamics occur for encoder-only and encoder-decoder models adapted to South African languages remains an open question that this project aims to answer.

3 PROBLEM STATEMENT & RESEARCH QUESTIONS

CPT has proven effective for adapting PLMs to low-resource languages [2, 5, 23], yet the adaptation process remains poorly understood. Current research typically evaluates models only at the end of training, leaving open the question of how and when linguistic knowledge develops during CPT. As a result, recent work has analysed model behaviour across intermediate training checkpoints to provide a more detailed view of how performance evolves during CPT [14]. It is also unclear whether learning patterns observed during pretraining from scratch, which has been studied much more extensively, also occur when adapting an already pretrained model to low-resource languages such as isiXhosa, or how the choice of base model affects these dynamics. This project addresses these gaps through the following research questions:

1. *To what extent do the learning dynamics observed during pretraining from scratch apply when encoder-only and encoder-decoder models are adapted to isiXhosa via CPT?*

Motivation: Prior work has established a hierarchical learning timeline during pretraining from scratch, where syntactic knowledge is acquired rapidly, followed by semantic and reasoning abilities [9, 17, 24]. CPT differs in that the model already encodes linguistic knowledge from pretraining, making it unclear whether the same timeline applies when adapting PLMs to low-resource languages.

2. *How do the learning dynamics during CPT differ between English-only, massively multilingual, and Nguni-specialised base models when adapted to isiXhosa?*

Motivation: English-only, massively multilingual, and Nguni-specialised base models encode different levels of prior linguistic knowledge before CPT begins, which may directly influence how quickly and in what order isiXhosa-specific knowledge is acquired [4, 14, 23].

3. *How do the learning dynamics during CPT differ between encoder-only and encoder-decoder architectures when adapted to isiXhosa?*

Motivation: Encoder-only and encoder-decoder models are trained on fundamentally different objectives, which may lead to different learning trajectories during CPT. Fine-tuning intermediate checkpoints on respective NLU and NLG tasks allows us to compare how quickly and in what order task-relevant knowledge develops across the two architecture types.

These research questions aim to provide a clearer understanding of how linguistic knowledge develops during CPT for low-resource languages, and how architecture and base model choice influence the rate and order of that development.

4 METHODOLOGY

This section outlines the proposed approach for investigating the learning dynamics of encoder-only and encoder-decoder models during CPT. It covers data and model selection, continued pre-training setup, checkpoint extraction, and evaluation. The overall process is summarised in Figure 2.

4.1 Continued Pretraining Data Selection

We will use the WURA corpus [25] as the source of isiXhosa pre-training data. WURA is a curated multilingual corpus covering 16 African languages, created by filtering mC4 [36] to remove low-quality text and crawling additional verified news sources. This results in higher-quality data, improving training efficiency by providing cleaner and more informative text. Our choice is motivated by WURA’s use in prior pretraining work for isiXhosa [19, 22], making it a realistic corpus for our context. The isiXhosa training subset contains approximately 13 million words [22], which, while smaller than alternatives such as mC4, is sufficient for CPT and more manageable given our computational resources. The dataset is publicly available on Hugging Face² and has already been pre-processed for use in training language models. We will tokenise the data using each model’s pretrained tokeniser prior to CPT.

4.2 Model Selection

Our model selection is guided by our research questions, which require representative models across the English-only, massively multilingual, and Nguni-specialised language coverage categories. These categories differ in the amount of prior linguistic knowledge encoded before CPT begins, which may influence how quickly and in what order isiXhosa-specific knowledge is learned [4, 14, 23]. For each language coverage category, we select one encoder-only and one encoder-decoder model. This allows the two architecture classes to be studied independently by the two group members. All models discussed are open-source and publicly available on Hugging Face.

4.2.1 English-Only Models. We select RoBERTa³ and T5⁴ as the English-only baselines for the encoder-only and encoder-decoder architectures, respectively. RoBERTa is chosen over BERT because it improves on BERT’s pretraining procedure and provides the architectural foundation for XLM-R. This allows for a controlled comparison within the encoder-only category. We choose T5 as

it introduced the text-to-text framework and provides a strong English-only baseline for the encoder-decoder architecture.

4.2.2 Massively Multilingual Models. We select XLM-R⁵ and ByT5⁶ as the massively multilingual models for the encoder-only and encoder-decoder architectures. XLM-R was pretrained on 100 languages, including isiXhosa, and outperforms alternatives such as mBERT across cross-lingual benchmarks [10]. It builds directly on RoBERTa, making it suitable for controlled comparisons within the encoder-only family. ByT5 is pretrained on 101 languages, including isiXhosa. Its byte-level design aligns with Nguni-ByT5, which is in the same architecture family. Our choice is further supported by evidence that byte-level modelling can outperform subword tokenisation in low-resource, morphologically complex languages such as isiXhosa [22, 27].

4.2.3 Nguni-Specialised Models. We select Nguni-XLMR⁷ and Nguni-ByT5⁸ as the Nguni-specialised models for the encoder-only and encoder-decoder architectures. These models further specialise XLM-R and ByT5 on the four South African Nguni languages (isiXhosa, isiZulu, isiNdebele, and Siswati), resulting in improved performance over their massively multilingual counterparts [23]. Our choice is motivated by evidence that restricting CPT to linguistically similar languages improves cross-lingual transfer and reduces negative interference [23].

4.3 Continued Pretraining & Checkpoint Extraction

All six models will be further pretrained on the isiXhosa training subset of the WURA corpus using the Hugging Face Transformers library [34] and PyTorch [26], following the hyperparameters of Meyer et al. [2024]. We adopt these hyperparameters as Meyer et al. [2024] demonstrate their effectiveness in adapting XLM-R and ByT5 to Nguni languages. The encoder-only models will be trained for 600k steps with a learning rate of 5×10^{-5} , no warmup steps, and a batch size of 80, while the encoder-decoder models will be trained for 10k steps with a learning rate of 1×10^{-4} , 5k warmup steps, and a batch size of 1024. The suitability of these hyperparameters will be monitored by tracking validation loss on the development set throughout training, and adjusted if necessary.

We will then extract model checkpoints at intervals adapted from Elhady et al. [2025] across both architectures. During the first 10% of training steps, we will save a checkpoint every 1% of steps to capture early changes in model behaviour. We will use smaller checkpoint intervals here because model behaviour changes most rapidly during early training, and larger intervals risk missing important early training trends [24]. For the remaining 90% of training, where changes in model behaviour are less rapid, we will save a checkpoint every 10% of steps. This results in 19 checkpoints per model, and 114 checkpoints in total across all six models.

²<https://huggingface.co/datasets/castorini/wura>

³<https://huggingface.co/FacebookAI/roberta-base>

⁴<https://huggingface.co/google-t5/t5-base>

⁵<https://huggingface.co/FacebookAI/xlm-roberta-base>

⁶<https://huggingface.co/google/byt5-base>

⁷<https://huggingface.co/francois-meyer/nguni-xlmr-large>

⁸<https://huggingface.co/francois-meyer/nguni-byt5-large>

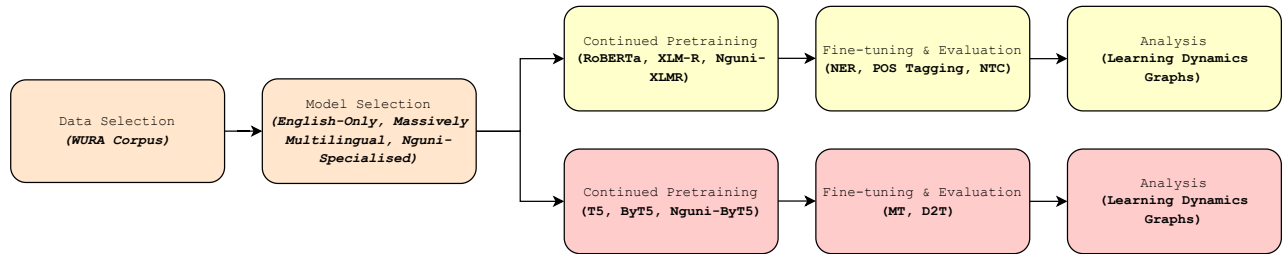


Figure 2: Overview of the proposed methodology. Orange boxes denote stages shared by both architecture types, yellow boxes denote encoder-only steps, and red boxes denote encoder-decoder steps.

4.4 Fine-tuning & Evaluation

Each intermediate checkpoint will be fine-tuned and evaluated on downstream tasks to track performance changes during CPT. Following the tasks, datasets, metrics, fine-tuning hyperparameters, and training epochs reported by Matzopoulos et al. [2025] and Meyer et al. [2024], we evaluate the encoder-only and encoder-decoder checkpoints as outlined in Table 2. We will then plot performance across checkpoints for each task to visualise and analyse how model performance changes on each task during CPT.

Table 2: Downstream evaluation tasks, datasets and metrics for each model type. Tasks include Named Entity Recognition (NER), Part-of-Speech (POS) tagging, News Topic Classification (NTC), Machine Translation (MT), and Data-to-Text generation (D2T).

Task	Dataset	Metric
Encoder-Only Models		
NER	MasakhaNER [1]	F1 Score
POS Tagging	MasakhaPOS [12]	F1 Score
NTC	MasakhaNEWS [3]	F1 Score
Encoder-Decoder Models		
MT	FLORES-200 [32]	BLEU
D2T	T2X [21]	chrF

5 ETHICAL, PROFESSIONAL & LEGAL ISSUES

This project does not anticipate significant ethical, professional, or legal issues. All models and datasets used are publicly available under open-source licences that permit academic research use, and all sources will be appropriately cited. No human participants are involved, and no personally identifiable information is used. The datasets used, including WURA and the downstream evaluation benchmarks such as MasakhaNEWS [3], may reflect biases present in their sources, which could influence model behaviour and evaluation outcomes. These factors will be considered when interpreting results. No new models will be publicly released as part of this project, reducing the risk of misuse or misrepresentation of these languages in downstream applications.

6 ANTICIPATED OUTCOMES

6.1 Potential Research Contributions

This project aims to make several contributions to the study of CPT for low-resource South African languages. Specifically, we aim to make the following contributions:

6.1.1 First Study of CPT Learning Dynamics for isiXhosa. This project will be one of the first comprehensive studies of learning dynamics during CPT for isiXhosa, addressing a broader gap in the interpretability of CPT as an adaptation strategy for low-resource languages. While learning dynamics have been studied extensively during pretraining from scratch, CPT remains poorly understood despite being the dominant approach for adapting PLMs to low-resource languages.

6.1.2 Prior Linguistic Knowledge and Adaptation. We will directly compare models within the encoder-only and encoder-decoder architecture families that differ in their prior linguistic knowledge of isiXhosa. This allows us to isolate the effect of prior knowledge on CPT learning dynamics. By applying CPT across the English-only, massively multilingual, and Nguni-specialised PLM language coverage categories, this study aims to provide direct insight into how prior linguistic knowledge affects adaptation speed and learning order during CPT for a low-resource language. Through this, we aim to offer guidance on base model selection for future work on adapting PLMs to low-resource languages.

6.1.3 Architectural Effects on Adaptation. This project will also offer the first controlled comparison of CPT learning dynamics across the encoder-only and encoder-decoder architectures when adapted to the same low-resource language. Since these architectures are trained using different objectives, they may follow different learning trajectories during CPT. In this study, we aim to compare these differences to provide insight into how architectural choices influence learning behaviour during adaptation.

6.1.4 Practical Guidance for Efficient CPT. Finally, this study aims to provide practical guidance for efficient CPT in low-resource contexts. By tracking task performance across intermediate checkpoints, we will identify when models reach near-peak performance during training. Through this, we will offer concrete recommendations on when training can be stopped early without significant loss in downstream task performance, supporting more efficient model training for low-resource languages. More broadly, this work aims to support the development of more effective NLP tools for

South African language speakers and to provide a replicable framework for studying CPT learning dynamics in other low-resource contexts.

7 PROJECT PLAN

This section outlines our overall project plan, including potential risks, expected timeline, required resources, key milestones, and work allocation.

7.1 Risks & Risk Management

We have identified key risks that could affect the success of the project, covering issues such as compute availability, timeline delays and potential group member burnout. These risks, along with their mitigation strategies, are summarised in Table 6.

7.2 Timeline

Our planned project timeline from April to September 2026 is presented as a Gantt chart in Figure 3. The timeline will be monitored and updated throughout the project to reflect progress and any changes recommended by our supervisor.

7.3 Milestones

The key project milestones and deliverables are summarised in Table 5.

7.4 Resources Required

The primary resource required for this project is access to high-performance computing GPU infrastructure for model training. This will be provided by the Centre for High Performance Computing (CHPC) Lengau cluster⁹, whose relevant specifications are summarised in Table 3. In addition, the project requires access to the WURA corpus for CPT and the downstream benchmark datasets for fine-tuning and evaluation. All datasets are publicly available on Hugging Face and are discussed in Section 4.1 and Section 4.4.

Table 3: CHPC Lengau GPU Node specifications.

Specification	GPU Nodes
CPU	Intel Xeon
Cores	36
Speed	2.2 GHz
RAM	32GiB
GPU	Nvidia V100
GPU RAM	16GiB / 32GiB

7.5 Work Allocation

Each team member is responsible for one architecture class, ensuring that the encoder-only and encoder-decoder investigations are conducted independently. Data preparation is the only shared task, completed collaboratively before each member proceeds with their respective models. The allocation is illustrated in Table 4.

Table 4: Work allocation by architecture class.

Team Member	Models
Siyabonga	RoBERTa, XLM-R, Nguni-XLMR
Rakeen	T5, ByT5, Nguni-ByT5

8 AI USAGE STATEMENT

Generative AI was used to provide feedback on writing, including clarity, structure, and organisation. Grammarly was used for proof-reading and grammar suggestions, which were reviewed before being applied. ChatGPT was used for general feedback on writing quality and structure, as well as assistance with LaTeX formatting, particularly for creating tables. No AI-generated text was copied directly into the document. All content was written and revised by the authors, with AI used only as a supporting tool.

REFERENCES

- [1] David Ifeoluwa Adelani, Jade Abbott, Graham Neubig, Daniel D’souza, Julia Kreutzer, Constantine Lignos, Chester Palen-Michel, Happy Buzaaba, Shruti Rijhwani, Sebastian Ruder, Stephen Mayhew, Israel Abebe Azime, Shamsuddeen H. Muhammad, Chris Chinenye Emezue, Joyce Nakatumba-Nabende, Perez Ogayo, Aremu Anuoluwapo, Catherine Gitau, Derguene Mbaye, Jesujoba Alabi, Seid Muhie Yimam, Tajuddeen Rabiū Gwadabe, Ignatius Ezeani, Rubungo Andre Niyongabo, Jonathan Mukiibi, Verrah Otiende, Iro Orife, Davis David, Samba Ngom, Tosin Adewumi, Paul Rayson, Mofetoluwa Adeyemi, Gerald Muriuki, Emmanuel Anebi, Chiamaka Chukwunke, Nkiruka Odu, Eric Peter Wairagala, Samuel Oyerinde, Clemencia Siro, Tobius Saul Bateesa, Temilola Oloyede, Yvonne Wambui, Victor Akinode, Deborah Nabagereka, Maurice Katusiime, Ayodele Awokoya, Mouhamadane Mboup, Dibora Gebreyohannes, Henok Tilaye, Kelechi Nwaike, Degaga Wolde, Abdoulaye Faye, Blessing Sibanda, Orevaoghene Ahia, Bonaventure F.P. Dossou, Kelechi Ogueji, Thierno Ibrahim Diop, Abdoulaye Diallo, Adewale Akinfaderin, Tendai Marengereke, and Salomey Osei. 2021. MasakhaNER: Named Entity Recognition for African Languages. *Transactions of the Association for Computational Linguistics* 9 (10 2021), 1116–1131. doi:10.1162/TACL_A_00416
- [2] David Ifeoluwa Adelani, Jesujoba Oluwadara Alabi, Angela Fan, Julia Kreutzer, Xiaoyu Shen, Machel Reid, Dana Ruiter, Dietrich Klakow, Peter Nabende, Ernie Chang, Tajuddeen Gwadabe, Freshia Sackey, Bonaventure F. P. Dossou, Chris Emezue, Colin Leong, Michael Beukman, Shamsuddeen H. Muhammad, Guyo D. Jarso, Oreen Yousuf, Andre N. Niyongabo Rubungo, Gilles Hacheme, Eric Peter Wairagala, Muhammad Umair Nasir, Benjamin A. Ajibade, Tunde Oluwaseyi Ajayi, Yvonne Wambui Gitau, Jade Abbott, Mohamed Ahmed, Millicent Ochieng, Anuoluwapo Aremu, Perez Ogayo, Jonathan Mukiibi, Fatoumata Ouoba Kabore, Godson Koffi Kalipe, Derguene Mbaye, Allahsera Auguste Tapo, Victoire M. Memdjokam Koagne, Edwin Munkoh-Buabeng, Valencia Wagner, Idris Abdulmumin, Ayodele Awokoya, Happy Buzaaba, Blessing Sibanda, Andiswa Bukula, and Sam Manthulu. 2022. A Few Thousand Translations Go a Long Way! Leveraging Pre-trained Models for African News Translation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Marine Carpuat, Marie-Catherine de Marnette, and Ivan Vladimir Meza Ruiz (Eds.). Association for Computational Linguistics, Seattle, United States, 3053–3070. doi:10.18653/v1/2022.naacl-main.223
- [3] David Ifeoluwa Adelani, Marek Masiak, Israel Abebe Azime, Jesujoba Oluwadara Alabi, Atnafu Lambebo Tonja, Christine Mwase, Odunayo Ogundepo, Bonaventure F.P. Dossou, Akintunde Oladipo, Doreen Nixdorf, Chris Chinenye Emezue, Sana Sabah Al-Azzawi, Blessing K. Sibanda, Davis David, Lolwethu Ndolela, Jonathan Mukiibi, Tunde O. Ajayi, Tatiana Moteu Ngoli, Brian Odhiambo, Abraham Toluwase Owodunni, Nnaemeka C. Obiefuna, Muhidin Mohamed, Shamsuddeen Hassan Muhammad, Teshome Mulugeta Ababu, Saheed S. Abdullahi, Mesay Gemedā Yigezu, Tajuddeen Gwadabe, Idris Abdulmumin, Mahlet Taye Bame, Oluwabusayo O. Awoyomi, Iyanuoluwa Shode, Tolulope Anu Adelani, Habiba Abdulganiy Kailani, Abdul Hakeem Omotayo, Adetola Adeeko, Afolabi Abebe, Anuoluwapo Aremu, Olanrewaju Samuel, Clemencia Siro, Wangari Kimotho, Onyekachi Raphael Ogbu, Chinedu E. Mbonu, Chiamaka I. Chukwunke, Samuel Fanijo, Jessica Ojo, Oyinkansola F. Awoson, Tadesse Kebede Guge, Sakayo Toadoun Sari, Pamela Nyatsine, Freedmore Sidume, Oreen Yousuf, Mardiyah Oduwole, Kanda P. Tshinu, Ussen Kimanuka, Thina Diko, Siyanda Nxakama, Sinodos G. Nugussie, Abdulmejid Tunī Johar, Shafie Abdi Mohamed, Fuad Mire Hassan, Moges Ahmed Mohamed, Evrard Ngabire, Jules Twagirayezu, Ivan Ssenkungu, and Pontus Stenetorp. 2023. MasakhaNEWS: News Topic

⁹<https://wiki.chpc.ac.za/chpc:lengau>

- Classification for African languages. *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics: Long Papers, IJCNLP-AACL 2023* 1 (2023), 144–159. doi:10.18653/V1/2023.IJCNLP-MAIN.10
- [4] Wisdom Aduah and Francois Meyer. 2025. Designing and Contextualising Probes for African Languages. (5 2025). <http://arxiv.org/abs/2505.10081>
 - [5] Jesujoba O. Alabi, David Ifeoluwa Adelani, Marius Mosbach, and Dietrich Klakow. 2022. Adapting Pre-trained Language Models to African Languages via Multilingual Adaptive Fine-Tuning. In *Proceedings of the 29th International Conference on Computational Linguistics*, Nicoletta Calzolari, Chu-Ren Huang, Hansaem Kim, James Pustejovsky, Leo Wanner, Key-Sun Choi, Pum-Mo Ryu, Hsin-Hsi Chen, Lucia Donatelli, Heng Ji, Sadao Kurohashi, Patrizia Paggio, Nianwen Xue, Seokhwan Kim, Younggyun Hahm, Zhong He, Tony Kyungil Lee, Enrico Santus, Francis Bond, and Seung-Hoon Na (Eds.). International Committee on Computational Linguistics, Gyeongju, Republic of Korea, 4336–4349. <https://aclanthology.org/2022.coling.1382/>
 - [6] Yonatan Belinkov and James Glass. 2019. Analysis Methods in Neural Language Processing: A Survey. *Transactions of the Association for Computational Linguistics* 7 (1 2019), 49–72. <http://arxiv.org/abs/1812.08951>
 - [7] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems* 2020-December (5 2020). <https://arxiv.org/pdf/2005.14165>
 - [8] Happy Buzaaba, Alexander Wettig, David Ifeoluwa Adelani, and Christiane Fellbaum. 2025. Lugh-Llama: Adapting Large Language Models for African Languages. (4 2025). <http://arxiv.org/abs/2504.06536>
 - [9] Cheng-Han Chiang, Sung-Feng Huang, and Hung yi Lee. 2020. Pretrained Language Model Embryology: The Birth of ALBERT. *EMNLP 2020 - 2020 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference* (10 2020), 6813–6828. <http://arxiv.org/abs/2010.02480>
 - [10] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised Cross-lingual Representation Learning at Scale. *Proceedings of the Annual Meeting of the Association for Computational Linguistics* (4 2020), 8440–8451. <http://arxiv.org/abs/1911.02116>
 - [11] Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference* 1 (10 2018), 4171–4186. <https://arxiv.org/pdf/1810.04805>
 - [12] Cheikh M. Bamba Dione, David Ifeoluwa Adelani, Peter Nabende, Jesujoba O. Alabi, Thapelo Sindane, Happy Buzaaba, Shamsuddeen Hassan Muhammad, Chris Chinenye Emezue, Perez Ogayo, Anuoluwapo Aremu, Catherine Gitau, Derguene Mbaye, Jonathan Mukiibi, Blessing Sibanda, Bonaventure F.P. Dossou, Andiswa Bukula, Rooweither Mabuza, Allahsera Augustine Tapo, Edwin Munkoh-Buabeng, Victoire Memdjokam Koagne, Fatoumata Ouoba Kabore, Amelia Taylor, Godson Kalipe, Tebogo Macucwa, Vukosi Marivate, Tajuddeen Gwadabe, Elvis Tchiaye Mboning, Ikechukwu Onyenwe, Gratien Atindogbe, Tolulope Anu Adelani, Idris Akinaade, Olanrewaju Samuel, Marien Nahimana, Théogène Musabeyezu, Emile Niyomutabazi, Ester Chimbenga, Kudzai Gotosa, Patrick Mizha, Apelete Agbolo, Seydou Traore, Chinedu Uchechukwu, Aliyu Yusuf, Muhammad Abdullahi, and Dietrich Klakow. 2023. MasakhaPOS: Part-of-Speech Tagging for Typologically Diverse African languages. *Proceedings of the Annual Meeting of the Association for Computational Linguistics* 1 (2023), 10883–10900. doi:10.18653/V1/2023.ACL-LONG.609
 - [13] Mengnan Du, Fengxiang He, Na Zou, Dacheng Tao, and Xia Hu. 2023. Short-cut Learning of Large Language Models in Natural Language Understanding. *Commun. ACM* 67 (12 2023), 110–120. Issue 1. doi:10.1145/3596490
 - [14] Ahmed Elhady, Eneko Agirre, and Mikel Artetxe. 2025. Emergent Abilities of Large Language Models under Continued Pretraining for Language Adaptation. *Proceedings of the Annual Meeting of the Association for Computational Linguistics* 1 (9 2025), 32174–32186. <http://arxiv.org/abs/2506.00288>
 - [15] Suchin Gururangan, Ana Marasovic, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. *Proceedings of the Annual Meeting of the Association for Computational Linguistics* (4 2020), 8342–8360. doi:10.18653/v1/2020.acl-main.740
 - [16] Jeremy Howard and Sebastian Ruder. 2018. Universal Language Model Fine-tuning for Text Classification. *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)* 1 (5 2018), 328–339. <http://arxiv.org/abs/1801.06146>
 - [17] Leo Z. Liu, Yizhong Wang, Jungo Kasai, Hannaneh Hajishirzi, and Noah A. Smith. 2021. Probing Across Time: What Does RoBERTa Know and When? *Findings of the Association for Computational Linguistics, Findings of ACL: EMNLP 2021* (9 2021), 820–842. <http://arxiv.org/abs/2104.07885>
 - [18] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. (7 2019). <http://arxiv.org/abs/1907.11692>
 - [19] Anri Lombard, Simbarashe Mawere, Temi Aina, Ethan Wolff, Sbonelo Gumede, Elan Novick, Francois Meyer, and Jan Buys. 2026. MzansiText and MzansiLM: An Open Corpus and Decoder-Only Language Model for South African Languages. (3 2026). <https://arxiv.org/pdf/2603.20732>
 - [20] Alexis Matzopoulos, Charl Hendriks, Hishaam Mahomed, and Francois Meyer. 2025. BabyLMs for isiXhosa: Data-Efficient Language Modelling in a Low-Resource Context. (1 2025). <https://arxiv.org/pdf/2501.03855>
 - [21] Francois Meyer and Jan Buys. 2024. Triples-to-isiXhosa (T2X): Addressing the Challenges of Low-Resource Agglutinative Data-to-Text Generation. 16841–16854 pages. <https://aclanthology.org/2024.lrec-main.1464/>
 - [22] Francois Meyer and Jan Buys. 2025. The Learning Dynamics of Subword Segmentation for Morphologically Diverse Languages. (11 2025). <http://arxiv.org/abs/2511.09197>
 - [23] Francois Meyer, Haiyue Song, Abhisek Chakrabarty, Jan Buys, Raj Dabre, and Hideki Tanaka. 2024. NGLUEni: Benchmarking and Adapting Pretrained Language Models for Nguni Languages. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue (Eds.). ELRA and ICCL, Torino, Italia, 12247–12258. <https://aclanthology.org/2024.lrec-main.1071/>
 - [24] Max Müller-Eberstein, Rob van der Goot, Barbara Plank, and Ivan Titov. 2023. Subspace Chronicles: How Linguistic Information Emerges, Shifts and Interacts during Language Model Training. *Findings of the Association for Computational Linguistics: EMNLP 2023* (10 2023), 13190–13208. <http://arxiv.org/abs/2310.16484>
 - [25] Akintunde Oladipo, Mofetoluwa Adeyemi, Orevaoghene Ahia, Odunayo Ogundepo, Abraham Toluwalase Owodunni, David Ifeoluwa Adelani, and Jimmy Lin. 2023. Better Quality Pre-training Data and T5 Models for African Languages. *EMNLP 2023 - 2023 Conference on Empirical Methods in Natural Language Processing, Proceedings* (2023), 158–168. doi:10.18653/v1/2023.emnlp-main.11
 - [26] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Advances in Neural Information Processing Systems* 32 (12 2019). <https://arxiv.org/pdf/1912.01703>
 - [27] Aleksandar Petrov, Emanuele La Malfa, Philip H. S. Torr, and Adel Bibi. 2023. Language Model Tokenizers Introduce Unfairness Between Languages. *Advances in Neural Information Processing Systems* 36 (10 2023). <http://arxiv.org/abs/2305.15425>
 - [28] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving Language Understanding by Generative Pre-Training. (2018). https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf
 - [29] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners. *OpenAI* (2019). https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf
 - [30] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research* 21 (9 2023), 1–67. <http://arxiv.org/abs/1910.10683>
 - [31] Naomi Saphra and Adam Lopez. 2019. Understanding Learning Dynamics Of Language Models with SVCCA. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference* 1 (4 2019), 3257–3267. doi:10.18653/v1/N19-1329
 - [32] Chihiro Taguchi, Seng Mai, Keita Kurabe, Yusuke Sakai, Georgina Agyei, Soudabeh Eslami, and David Chiang. 2025. Languages Still Left Behind: Toward a Better Multilingual Machine Translation Benchmark. (8 2025), 20142–20154. doi:10.18653/v1/2025.emnlp-main.1018
 - [33] Ashish Vaswani, Google Brain, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. (6 2017), 1. <https://arxiv.org/pdf/1706.03762>
 - [34] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi 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 M. Rush. 2019. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *Journal of Machine Learning Research* (10 2019). <https://arxiv.org/pdf/1910.03771>

- [35] Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2022. ByT5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics* 10 (3 2022), 291–306. <http://arxiv.org/abs/2105.13626>
- [36] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. *NAACL-HLT 2021 - 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference* (3 2021), 483–498. <http://arxiv.org/abs/2010.11934>
- [37] Hao Yu, Tianyi Xu, Michael A. Hedderich, Wassim Hamidouche, Syed Waqas Zamir, and David Ifeoluwa Adelani. 2026. AfriqueLLM: How Data Mixing and Model Architecture Impact Continued Pre-training for African Languages. (1 2026). <http://arxiv.org/abs/2601.06395>

9 APPENDICES

APPENDIX A: PROJECT MILESTONES & DELIVERABLES

Table 5: Project milestones grouped by phase, with key official deliverable deadlines. Yellow rows indicate the draft milestone and its corresponding official submission deadline. Orange rows indicate the final paper milestone and its corresponding official submission deadline.

Project Phase	Milestone Description	Target Date
Phase 1	Submit final project proposal	5 May 2026
	Gain CHPC access	9 May 2026
Phase 2	Environment setup complete	16 May 2026
	Data processing complete	27 May 2026
Phase 3	MLM checkpoint extraction complete	6 July 2026
	Encoder-Decoder checkpoints extraction complete	6 July 2026
Phase 4	MLM finetuning complete	26 July 2026
	Evaluation metrics for MLM captured	30 July 2026
	Encoder-Decoder models finetuning complete	26 July 2026
	Evaluation metrics for Encoder-Decoder models captured	30 July 2026
Phase 5	Learning dynamics analysis complete	10 August 2026
Phase 6	Project paper scaffold complete	6 August 2026
	Project paper draft complete	23 August 2026
	Final project paper complete	3 September 2026
Phase 7	Project code finalised	13 September 2026
	Poster completed	21 September 2026
	Website completed	29 September 2026
Key Official Deadlines		
Project Phase	Milestone Description	Target Date
<i>Official</i>	Full draft of final paper	24 August 2026
<i>Official</i>	Final research paper submission	4 September 2026
<i>Official</i>	Final project demonstration	16 – 23 September 2026

APPENDIX B: RISK & MITIGATION MATRIX

Table 6: Risk matrix outlining key technical, skill, scheduling, and team risks along with their mitigation strategies.

Risk	Category	Cause	Trigger	Impact	Probability	Mitigation Strategy
Task Underestimation	Schedule	Unforeseen complexities cause project phases to take longer than expected.	A project phase exceeds its allocated time on the Gantt chart.	High - Delays subsequent phases and final submission.	Likely	Regularly track progress and build in buffer time. Tweak checkpointing methodology from 10% to 20%. Drop one finetuning evaluation task.
Compute Resource Bottleneck	Technical	Long CPT jobs and hundreds of fine-tuning evaluations cause severe delays in the shared CHPC queue.	CHPC jobs remain pending for 48+ hours.	High - Delays evaluation phase.	Likely	Automate batch scripts and request access to CHPC early. Drop one finetuning task (e.g MT/NER) to halve finetuning jobs.
Storage Capacity Overload	Technical	Saving all CPT checkpoints with every fine-tuned downstream model will exhaust the CHPC hard drive.	CHPC storage reaches 90% capacity.	High - Training crashes mid-run.	Likely	Keep essential CPT model weights. Discard fine-tuned model weights after evaluation. Move older checkpoints to external storage.
CHPC Learning Curve	Skills	Lack of prior experience with CHPC job scheduling and environment modules causes setup delays.	Taking more than 3 days to run a basic test GPU job.	Medium - Delays the Environment Setup phase.	Likely	Run basic test jobs and configure the PyTorch environment early. Consult official CHPC documentation. Escalate to CHPC support.
Competing Academic Workload	Schedule	Exams in June and remaining coursework clash with planning phase and Block 3 start.	Team member is unavailable for scheduled project work.	Medium - Delays in initial setup and data preparation.	Likely	Allocate planning tasks and sync academic calendars. Front load tasks. Prioritise critical tasks to meet deadlines.
Team Member Burnout/Drop-out	Team	Heavy overall workload or personal issues lead to severe burnout or dropping Honours.	A team member withdraws or is unresponsive for more than 7 days.	Very High - Loss of the project's workforce and research contribution.	Unlikely	Open communication between teammates. Buffer times and days to cope. Redefine project to just the single architecture for remaining member.

APPENDIX C: GANTT CHART

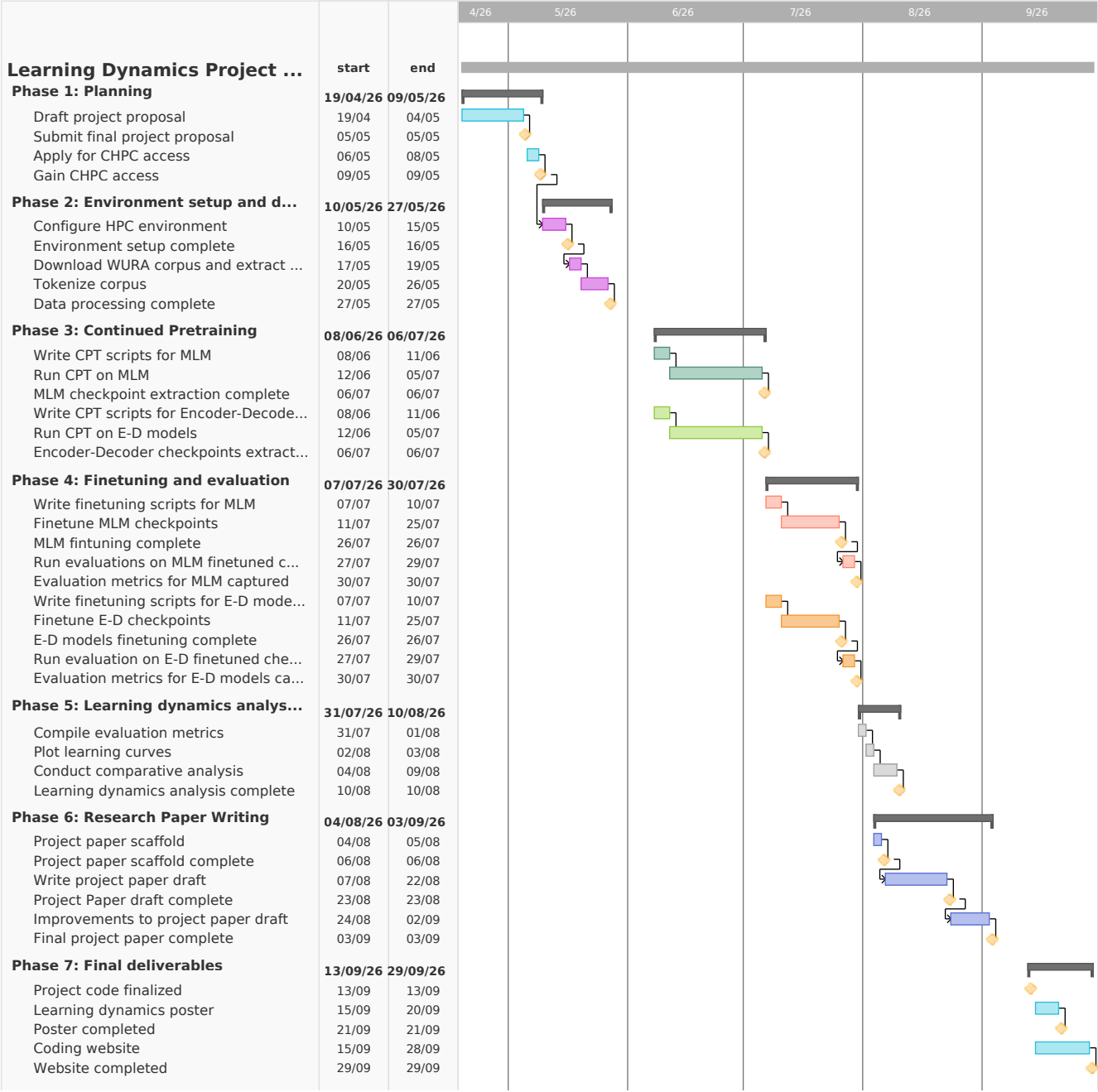


Figure 3: Gantt chart illustrating the project schedule from April to September 2026.