Anchor-Slave Contrastive Fine-Tuning: Task Adaptation for Low-Resource Languages via English Embedding Projection

Emmanuel O. Olateju
username: oluwasegun
oluwasegun@aims.ac.za
Department of AI / Mathematical Sciences
African Institute of Mathematical Sciences, South Africa
University of Cape Town, South Africa
https://github.com/emmanuel-olateju/anchor-slave-LLM

Abstract

The success of modern Natural Language Processing (NLP) is heavily concentrated on high-resource languages like English, leaving a significant gap for low-resource languages, particularly in specialized downstream tasks. This work addresses the challenge of extending task-specific knowledge to low-resource African languages: Amharic, Hausa, and Swahili; this work also prioritizes computational accessibility.

We propose an Anchor-Slave Contrastive Fine-Tuning framework. This methodology utilizes a high-resource, task-optimized English model as a static knowledge anchor, compelling a smaller, resource-efficient slave model to project its low-resource language embeddings into the anchor's proven task-specific embedding space. The fine-tuning incorporates a dual objective: classification and alignment loss (a contrastive loss).

Our experiments validate that African languages can be effectively adapted to tasks by being anchored on English-optimized model embeddings. We systematically evaluate various Anchor/Slave architectural pairings, demonstrating that task-specific knowledge transfer is highly successful, even between non-matching architectures (e.g., LaBSE anchor guiding distilled-BERT slave). This finding establishes the feasibility of our ultimate goal: enabling low-resource languages to quickly and efficiently adapt to new tasks using computationally accessible models, provided an optimized English model exists. We also discuss the critical trade-off between model compression and the architectural capacity required for complex low-resource languages.

Github: github.com/emmanuel-olateju/anchor-slave-LLM

1 Introduction

The remarkable progress in Natural Language Processing (NLP) over the past decade has been largely driven by sophisticated deep learning models, particularly the Transformer architecture, which underpins modern Large Language Models (LLMs) and sentence encoders (Vaswani et al., 2017). However, this advancement presents a critical paradox: while NLP capabilities have soared, their benefits remain heavily concentrated on high-resource languages, most notably English. This disparity creates a significant access and equity gap for the majority of the world's languages, which are classified as low-resource.

Low-resource languages, such as the African languages investigated here (Amharic, Hausa, and Swahil) suffer from a severe lack of high-quality and task-specific training data (Sohail et al., 2020). This data scarcity makes it impractical to train state-of-the-art models from scratch. Furthermore, even when training is feasible, the resulting models often perform poorly or generalize insufficiently, especially for nuanced downstream tasks like polarization classification.

Exacerbating this issue is the computational burden of large models. Modern LLMs require vast computational resources for training and inference, which is often an insurmountable barrier for researchers and developers in resource-constrained environments. Our project is motivated by a crucial requirement: to prioritize the development of solutions that are not only performant but also computationally accessible (i.e., small, distilled models) to ensure the democratization of NLP technology.

1.1 Multilingual Alignment and the Knowledge Transfer Imperative

To bridge this gap, research has increasingly focused on cross-lingual transfer learning, leveraging knowledge learned from high-resource languages to boost performance in low-resource ones. Multilingual pre-trained models such as BERTs (Devlin et al., 2019), Sentence-BERT (SBERT) (Reimers and Gurevych, 2019), and the Language Agnostic

BERT Sentence Embeddings (LaBSE) (Feng et al., 2020), represent a significant stride in this direction. The latter model is explicitly trained with contrastive objectives to map similar sentences from different languages into a shared, unified embedding space . While effective, the general application of these models for specific, downstream tasks (like polarization classification) often requires fine-tuning, and the models themselves are still large.

1.2 The Proposed Solution: Anchor-Slave Contrastive Fine-Tuning

This work proposes and validates a novel, resource-efficient strategy: Bi-Lingual Contrastive Fine-Tuning using an Anchor-Slave Model Pair framework.

- **Anchor Model:** We utilize an existing optimized English model for the target task (e.g., polarization classification) to serve as the Anchor Model. This model's embedding space acts as the target knowledge structure.

- **Slave Model:** A smaller, computationally accessible model (the Slave Model) is fine-tuned on the African language data. Critically, its fine-tuning objective is dual-purpose: performing the task classification and aligning its output embeddings through a contrastive loss with the static embeddings generated by the English Anchor Model.

Our investigation systematically evaluates this framework using various encoder architectures for both the Anchor and Slave roles, including small, distilled models like distilled-BERT and deberta-small.

The ultimate goal of this project is to develop a robust, generalized methodology that enables any low-resource language to easily adapt to any new NLP task, provided an optimized English model for that task exists. The core finding that African languages can be effectively trained on specific tasks by being anchored on English-optimized model embedding demonstrates the feasibility of this vision.

The remainder of this report is organized as follows: Section 2 presents the Methods for data, models, and the Anchor-Slave contrastive fine-tuning, acting as an executive summary of the experimental approach. Section 3 details the experiments conducted, discussing the results of each experiment in the chronological order of their execution, beginning with the initial decoupled model fine-tuning. Section 4 discusses and analyzes the comparative performance of various Anchor-Slave pairs, providing analysis of the final task performance. Section 5 concludes the work and outlines future directions for low-resource language adaptation.

2 Methods

2.1 Data

The experiments focus on the task of polarization classification across four languages: English and three African languages: Amharic, Hausa, and Swahili; using the dataset provided in the SemEval Polarization challenge on codabench.

For knowledge distillation of embeddings, the FLORES-600 dataset was utilized, incorporating several African languages, including: Yoruba, Hausa, Igbo, Amharic, Swahili, Zulu, Somali, and Kinyarwanda, in addition to English.

We note a challenge with the Amharic language in the SemEval dataset. Sentences of the Ahmaric language are presented in the native Amharic script, which, along with its unique morphology and grammar, may contribute to lower model performance.

2.2 Initial Decoupled Models (Smaller)

The study first explored separate fine-tuning of models on the SemEval-Polarization task without an alignment objective, then the study adopted contrastive alignment of embeddings. The study later adopts both larger and distilled models, prioritizing computational accessibility in terms of model size, and the use larger models for comparison.

The distilled models in Tables **??** - 1 were initially fine-tuned separately for each language without an explicit alignment objective. At this stage classification employed a binary cross entropy loss for the binary-polarization task, and cross-entropy for the other tasks (polarization type and manifestation classification). The result of the decoupled fine-tuning are shown in Figures S1 - S3. The (binary-)cross-entropy loss is given as:

$$\mathcal{L}_{BCE}(y,\hat{y}) = -\tfrac{1}{N}\textstyle\sum_{i=1}^{N}\left[y_i\log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)\right] \tag{1}$$

$$\mathcal{L}_{CE} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\log(\hat{y}_{i,c}) \tag{2}$$

2.3 Bi-Lingual Constrastive Fine-Tuning

The core of the methodology is the bi-lingual contrastive fine-tuning, adopting the LaBSE philos-

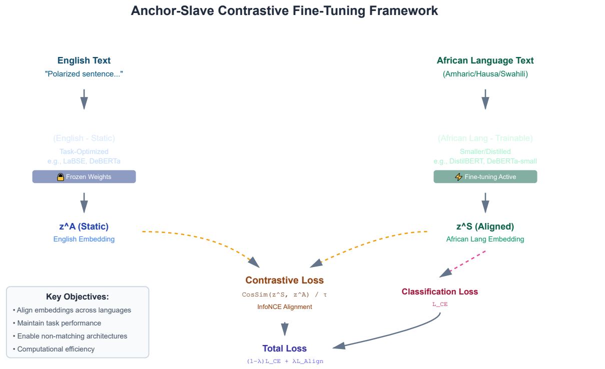

Figure 1: Bi-lingual Embedding Alignment

ophy of multi-lingual alignment. The training process involves an Anchor-Slave Model Pair to project embeddings. An anchor model (**ANC(EN)**) for the high-resource English language generates static embeddings, while a slave model (textbfSL) is fine-tuned for a low-resource African language, with the objective of aligning its embeddings with those of the Anchor Model for positive examples; this is done through a contrastive loss that picks positive examples as instances of English sentences with the same polarity.

2.3.1 Loss Function Simplified

The loss function is composed of a similarity function which is the temperature scaled cosine similarity between the L2-normalized aligned Slave embedding (z_i^S) and the static Anchor embedding (z_j^A):

$$s_{i,j} = \frac{CosSim(\mathbf{z}_i^S, \mathbf{z}_j^A)}{\tau} = \frac{(\mathbf{z}_i^S/\|\mathbf{z}_i^S\|) \cdot (\mathbf{z}_j^A/\|\mathbf{z}_j^A\|)}{\tau} \quad (3)$$

where z^S: Aligned Slave Model Embedding (query), z^A: Static Anchor Model Embedding (key), τ: Temperature parameter.

Based on the similarity function we use the InfoNCE-style contrastive loss (Goyal, 2020), averaged over M total flattened samples (Batch Size X Subset Size)

$$\mathcal{L}_{Align} = -\frac{1}{M}\sum_{i=1}^{M} \log\left(\frac{\sum_{j \in P(i)} e^{s_{i,j}}}{\sum_{k=1}^{M} e^{s_{i,k}}}\right) \quad (4)$$

where M: total number of samples in the flattened batch, $P(i)$: The set of indices j in the batch where the anchor label matches the slave label ($y_i^S = y_j^A$), representing positive pairs.

Then the weighted cross-entropy loss is applied to the slave language classification task, averaged over the batch sie N:

$$\mathcal{L}_{CE} = -\frac{1}{N}\sum_{i=1}^{N} W_{y_i} \sum_{c=1}^{C} y_{i,c} \log(\hat{y}_{i,c}) \quad (5)$$

The total loss function is the weighted combination of the classificatioin and alignment objectives:

$$\mathcal{L}_{total} = (1 - \lambda_{align}) \cdot \mathcal{L}_{CE} + \lambda_{align} \cdot \mathcal{L}_{Align} \quad (6)$$

where $_{align}$ is the alignment weighting factor which typicallt ranges between 0 and 1.

The fine-tuning process is summarized into polarization classification on the African language and alignment with the English embeddings from the Anchor Model. This process was conducted for each African language, with English always serving as the anchor; this is described by Figure 1.

2.3.2 Models

We first adopt smaller distilled models for decoupled (mono-lingual) fine-tuning as shown in Figure 1. Then, two larger models were then considered to address the consistently low performance of the small-distilled models on Ahmaric (Table 2). The **cardiffnlp/twitter-roberta-base-sentiment-latest**, optimized for sentiment analysis, slightly improved polarization identification on the English language. While the **sentence-transformers/LaBSE**, LaBSE (Language Agnostic BERT Sentence Embeddings) was adopted as the predominant model for multi-lingual alignment. It is specifically trained to align similar sentences from various languages in a shared embedding space using a contrastive objective. Initial fine-tuning of LaBSE on each language resulted in a sudden performance increase for Amharic to 70%..

2.4 Encoder Strucuture Validation

Experiments systematically compared the effect of Anchor and Slave model architectures in two stages:

- **Similar Encoder Structure:** Testing pairs where the Anchor and Slave models use the same architecture (e.g., small-deberta with small-deberta). The small-deberta, MiniLM, twitter-roberta-base, and LaBSE were initially considered as English-optimized anchor models.
- **Non-Matching Encoder Pairs:** Evaluating all possible combinations where the Anchor-Encoder architecture is not the same as the Slave-Encoder. The cardiff-NLP model was excluded due to poor classification performance after cross-lingual alignment.

Model	Amharic	English	Hausa	Swahili
microsoft/deberta-v3-small	0.43	0.79	0.86	0.76
prajjwal1/bert-tiny	0.43	0.43	0.47	0.69
microsoft/multilingual-MiniLM-L12-H384	0.43	0.74	0.47	0.77
distilbert-base-multilingual-cased	0.43	0.73	0.82	0.77
google/electra-small-discriminator	0.43	0.72	0.47	0.75

Table 1: Decoupled performance of distilled models on polarization task.

2.5 Knowledge Distillation for Comparison

To mitigate the concern of LaBSE's large size , knowledge distillation was explored as a method of model compression (Hinton et al., 2015). The goal was to distill the "wisdom" into smaller models like the deberta-small and distilled-BERT. An already distilled model, LaBSE-small (with poor Amharic performance of 52.9%) and the distilled-BERT, were each used as student models. The distillation focused on transferring embeddings (rather than logits) from the original base LaBSE (the teacher) to the LaBSE-small. The performance of the distilled slave models was then assessed in the context of bi-lingual fine-tuning with the distilled-BERT and large LaBSE as anchor models.

3 Experiments & Results

3.1 Decoupled Performances & Optimal English Models

We initiated experiments by fine-tuning five small, distilled models on polarization classification for all four languages (Table 1, Figures S1 - S3). On English, all models except bert-tiny achieved strong performance, validating their baseline capacity. For Hausa and Swahili, the results were consistently lower than English, illustrating the expected resource gap. However, the models showed limited consistency across African languages, prompting further investigation.

The most significant finding from the baseline was the consistently low performance on Amharic (Figure S1), with most small models failing to exceed 55% F1 score. This severe degradation, likely attributed to the language's unique non-Latin script, divergent morphology, and scarce data, established the necessity for a powerful cross-lingual transfer mechanism to effectively adapt to the task.

3.1.1 Smaller Models

We proceed to fine-tune distilled models as this study prioritizes task-specific and low resource alignment of models for use with African languages. We evaluate the following models: 'microsoft/deberta-v3-small' (He et al., 2021), 'prajjwal1/bert-tiny', 'microsoft/multilingual-MiniLM-L12-H384', 'distilbert-base-multilingual-cased', 'google/electra-small-discriminator'.

Model	Amharic	English	Hausa	Swahili
cardiffnlp/twitter-roberta-base-sentiment-latest	0.43	0.81	0.82	0.77
sentence-transformers/LaBSE	0.70	0.78	0.82	0.77

Table 2: Decoupled performance of large models on polarization task.

3.1.2 Larger Models

Due to the persistent poor performance on Amharic, we ask three key questions: (1) Will the projection of African language embeddings to the English language embedding-space within the task context level classification performance across languages, (2) Will a secondary objective of sequence-to-sequence translation (African language to English) alongside classification improve performance, and (3) Will prior translation to English plus single classification objective level performance across all languages? While various methods can be designed and adopted to answer these questions, the associated computational and time costs with the second and third questions rather pushes us in the direction of contrastive multi-lingual sentence embeddings. We evaluated two larger architectures, the twitter-RoBERTa-base for sentiment analysis and the contrastively trained LaBSE.

While the task-specific twitter-roberta provided only marginal gains on English, fine-tuning the LaBSE model proved transformative. LaBSE, pretrained for cross-lingual embedding alignment, immediately improved Amharic performance from the baseline 55% to a competitive 70% F1 score (Table 2). This result was crucial, as it confirmed our hypothesis: **effective task adaptation for low-resource languages requires explicit embedding alignment.** This validation led directly to the design of the Anchor-Slave contrastive framework.

3.2 Comparison of Anchor Models

While we select the English language as the anchor language due to its high resourced-ness and plethora of optimized models, we also compare the effect of changing the anchor models–which generate static embeddings of the anchor language–and slave-models (models to be fine-tuned). We consider fist the scenario where anchor- and slave-models use the same encoder architecture, then we evaluate the performance of all possible pairs of anchor and slave-models where the anchor-encoder

Model	Amharic		Hausa		Swahili		English
	SL	SL(EN)	SL	SL(EN)	SL	SL(EN)	ANC(EN)
microsoft/deberta-v3-small	0.50	0.58	0.69	0.40	0.75	0.74	0.79
distilbert-base-multilingual-cased	0.54	0.75	0.85	0.48	0.77	0.48	0.73
cardiffnlp/twitter-roberta-base-sentiment-latest	0.43	0.74	0.47	0.38	0.33	0.26	0.81
sentence-transformers/LaBSE	0.65	0.62	0.84	0.62	0.77	0.71	0.78

Table 3: Performance after cross-lingual alignment of embedding using same anchor- and slave-model encoder model architecture. SL = Slave-Model on African language; SL(EN) = Slave-Model on English language; ANC(EN) = Anchoe-Model on English-Language.

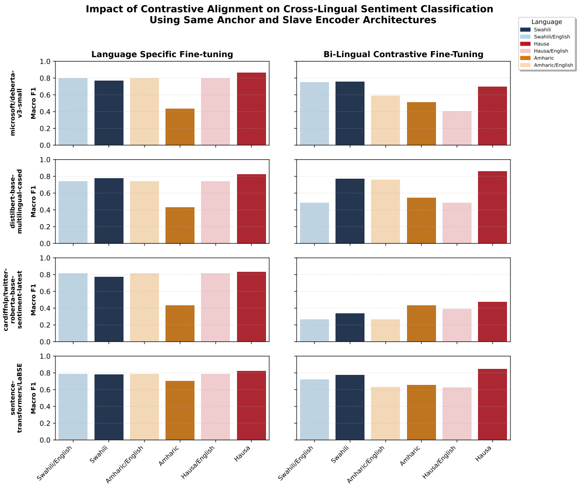

Figure 2: comparing bi-lingual embedding alignment to language decoupled fine-tuning.

architecture is not the same as the slave. This is done for each of the African language considered in this study, and the anchor language is always English.

3.2.1 Similar Encoder Structure

We implemented the Anchor-Slave contrastive fine-tuning framework to explicitly align African language embeddings (z^S) with the static task-specific English anchor embeddings (z^A). Initial tests with matching encoder architectures (e.g., LaBSE-Anchor → LaBSE-Slave) established the performance ceiling, yielding the highest F1 scores across all languages (Table 3). This established the effectiveness of the dual-objective loss in maintaining task performance while enforcing inter-lingual alignment.

3.2.2 Non-matching Encoder Pairs

Crucially, we observed that high performance was retained even with non-matching encoder pairs. The combination of the powerful LaBSE Anchor with a computationally efficient distilled-BERT Slave model and vice-versa successfully transferred the task knowledge with minimal performance drop when the LaBSE is the slave (Figures 3-6). This result confirms the core hypothesis that the Anchor model's embedding space is sufficient for guiding resource-constrained models, paving the way for efficient, task-adaptive low-resource NLP.

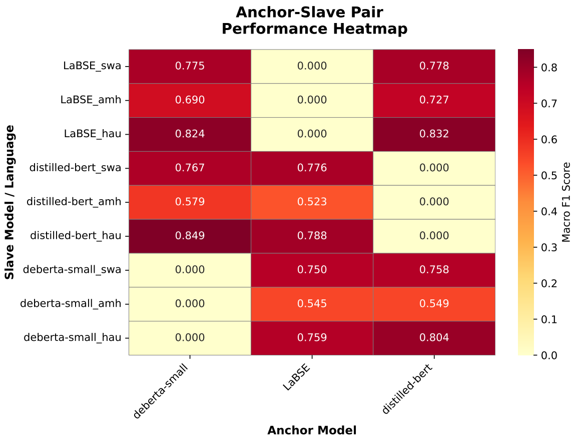

Figure 3: Macro-F1 performance on African languages with different anchor-slave encoder pairs.

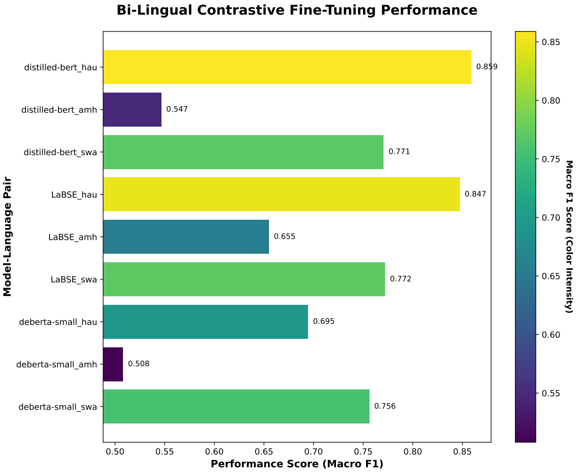

Figure 4: Macro-F1 performance on African languages with same anchor-slave encoder architectures.

Figures 3-6 demonstrates a significant reduction in inter-lingual variance for same-class embeddings after fine-tuning, confirming the successful projection of the slave language representations into the English anchor space. However, the robustness and diversity of LaBSE is not sufficient alone to perserve performance on the English language. The large size of the LaBSE as a slave-model is also of concern. So we look to methods of compressing the large LABSE, and multiple anchors optimized to overcome the limitations of neighboring anchors as ensembles.

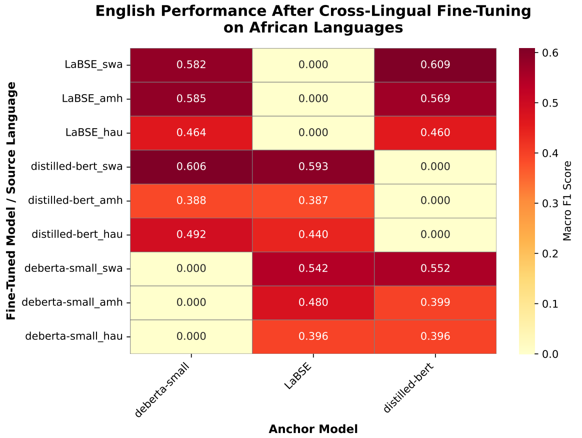

Figure 5: Macro-F1 performance on tehj English language with different anchor-slave encoder pairs.

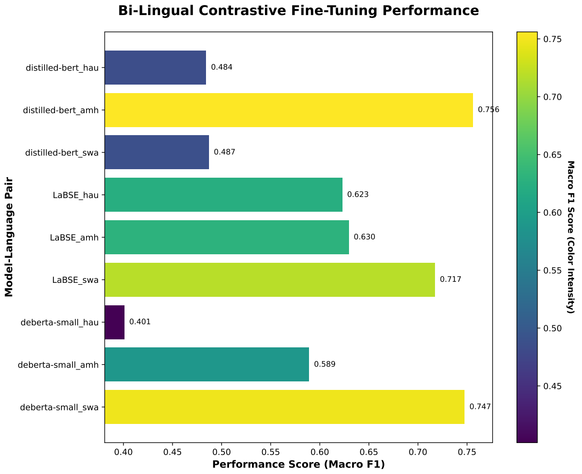

Figure 6: Macro-F1 performance on the English language with same anchor-slave encoder architectures.

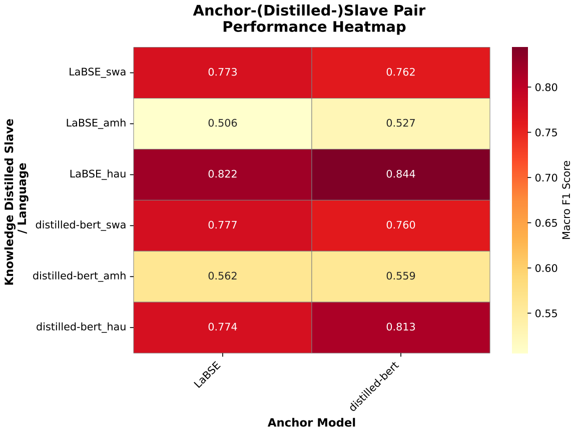

Figure 7: Macro-F1 using slave-models distilled from embeddings of base LaBSE model.

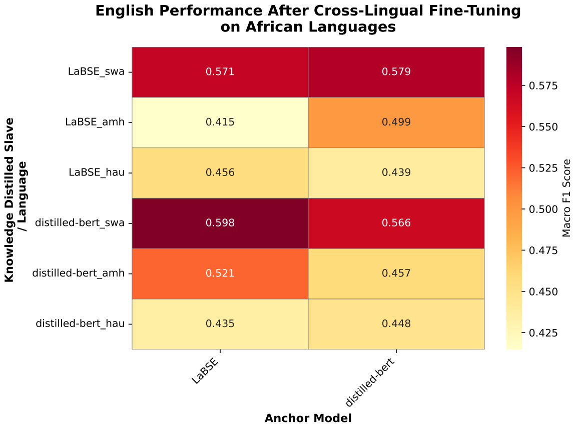

Figure 8: Macro-F1 on English after bi-lingual fine-tuning; slave-models are distilled from embeddings of base LaBSE model.

3.3 LaBSE Compression

To address the computational cost of LaBSE, we investigated model compression by distilling the base LaBSE embeddings (the teacher) to smaller architectures (the student, LaBSE-small/distilled-BERT). This process, which used a broad set of African language data for training, resulted in an overall performance degradation across all African languages (Figures 7-8). The most severe drop was observed for Amharic (down to 52.9% F1), indicating that the aggressive architectural compression compromised the model's capacity to handle complex, low-resource language dynamics. This suggests that while anchoring is effective, future work must explore less invasive compression methods, such as quantization or LORA, to preserve the necessary architectural depth of the best-performing models.

4 Discussion and Analysis

4.1 The Imperative of Cross-Lingual Alignment

The initial decoupled experiments established the necessity for our framework: small, distilled models fine-tuned independently failed on low-resource African languages, particularly Amharic. This indicated that simple multilingual pre-training is insufficient for specialized tasks. Our pivotal finding was that fine-tuning the LaBSE model immediately boosted Amharic performance to 70% F1. Given LaBSE's foundation in embedding alignment, this result strongly validated our central hypothesis: **effective task performance in low-resource settings requires successful cross-lingual embedding alignment.** This evidence necessitated the development of the Anchor-Slave framework to explicitly enforce this task-specific projection.

4.2 Efficacy of the Anchor-Slave Task Projection Framework

The Bi-Lingual Contrastive Fine-Tuning framework successfully transitioned from general multilingual alignment to task-specific embedding projection. The experimental results consistently proved that the dual-objective loss function allows task-optimized English embeddings (z^A) to guide the representation learning of African language models (z^S). While matching architectures (e.g., LaBSE-Anchor → LaBSE-Slave) provided the performance ceiling, the most critical finding was the strong performance retained by non-matching pairs (e.g., LaBSE Slave guided by a distilled-BERT Anchor and vice-versa). This success demonstrates that task knowledge transfer can transcend architectural boundaries, establishing the core feasibility of using a powerful English model to adapt a computationally small, low-resource model.

4.3 The Architectural Trade-off: Accessibility vs. Complexity

Although the Anchor-Slave projection was successful, attempts to achieve ultimate computational accessibility via Knowledge Distillation highlighted a critical trade-off. Distilling the task-specific embeddings from the base LaBSE to the smaller LaBSE-small resulted in performance degradation across African languages, most severely on Amharic, where the F1 score dropped to 52.9%. This suggests that the architectural complexity of languages like Amharic with its unique morphology and non-Latin script may demand a larger model capacity than provided by the smallest distilled architectures. The architecture itself imposes a constraint on compression. While the efficacy of the anchoring method was proven, future work must explore less invasive compression techniques, such as quantization or LORA, that preserve the robust architectural capacity of the base LaBSE model.

4.4 Implications and Future Work

The overall success of the Anchor-Slave framework confirms the central conclusion: low-resource languages can be effectively adapted to specialized tasks by leveraging the embedding space of a task-optimized high-resource model. This methodology is a significant step toward democratizing high-performance NLP by reducing the dependency on massive computational power and large native language datasets.

5 Conclusion

This project successfully investigated a robust, resource-efficient methodology for extending task-specific NLP capabilities to low-resource African languages (Amharic, Hausa, and Swahili). We proposed and validated a Bi-Lingual Contrastive Fine-Tuning framework that utilizes an optimized English model as a static knowledge anchor to guide the task-specific embedding projection of a smaller, low-resource slave model. Our experiments confirmed this approach effectively facilitates task-specific knowledge transfer, achieving successful polarization classification performance where decoupled small-model baselines failed. The systematic evaluation of matching and non-matching encoder pairs demonstrated that task adaptation is feasible even with architectural mismatches, strongly supporting the feasibility of our ultimate goal: enabling low-resource languages to quickly adapt to any new NLP task via an English anchor.

However, our work also revealed a critical limitation concerning computational accessibility. Attempts at aggressive model compression via knowledge distillation resulted in significant performance degradation, particularly for the complex Amharic language. This indicates that there is an inherent architectural capacity required for modeling complex, low-resource linguistic phenomena. This underscores the need to find the optimal balance between small model size and the architectural capacity necessary to retain performance.

Future research should focus on two key areas. First, we must explore alternative compression techniques, such as quantization (Zafrir et al., 2019) or Low-Rank Adaptation (LORA) (Hu et al., 2022), to reduce memory footprint without compromising the structural integrity of the high-capacity Anchor and Slave models. Second, the Anchor-Slave framework should be applied to a wider variety of downstream tasks (e.g., Named Entity Recognition, Question Answering) to validate the generalizability of this task-specific embedding projection methodology.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Shi Feng, Duyu Wang, Hai Xu, Wenyu Wang, Shujian Zhang, Bo Zhao, Congying Li, Muwan Liu, and Zhiyuan Zhang. 2020. Language-agnostic BERT sentence embedding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5596–5605.

Vivek Goyal. 2020. InfoNCE does not equal contrastive learning. In *International Conference on Machine Learning (ICML)*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced BERT with disentangled attention. *arXiv preprint arXiv:2111.05069*.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. In *NIPS Deep Learning Workshop*.

Edward J Hu, Yelong Shen, Shuzheng Zheng, Chin-Yi Zheng, Haotian Wu, and Younes Awadalla. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-networks. 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 3982–3992.

Al-Hassan Sohail, Kelechi Ogueji, Shamsuddeen Muhammad, David Ogayo, and 1 others. 2020. Masakhane: Machine translation for African languages. *arXiv preprint arXiv:2003.11183*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Ofir Zafrir, Guy Boudoukh, Peter Izsak, and Moshe Farchi. 2019. Q8BERT: Quantized 8-bit BERT on the edge. In *NeurIPS Workshop on Efficient Deep Learning*.

6 Supplementary

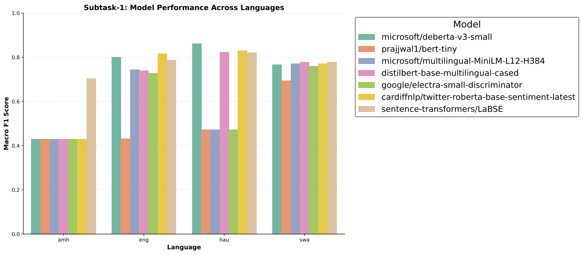

Figure S1: Enter Caption

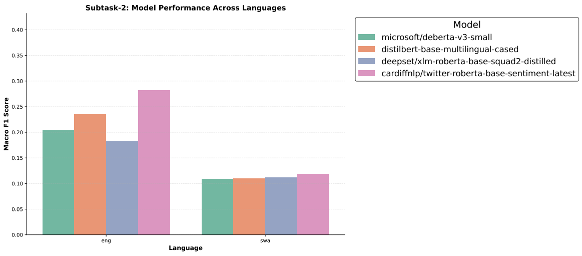

Figure S2: Enter Caption

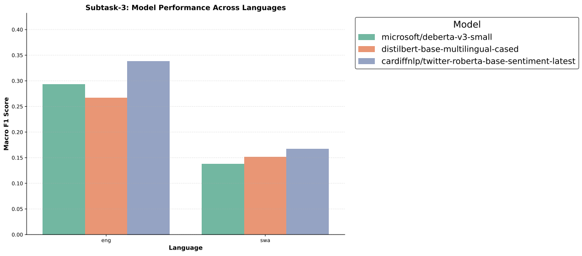

Figure S3: Enter Caption

This is an appendix.