Efficient Adaptation of English Language Models for Morphologically Rich and Underrepresented Languages: The Case of Arabic

Ahmed Eldamaty[1] **Mohamed Maher Abdelrahman**[2] **Mohamed Mostafa Ibrahim**[1]
Mariam Ashraf[1] **Radwa Elshawi**[2]
[1]Giza Systems, Cairo, Egypt [2]University of Tartu, Tartu, Estonia
{ahmed.aldamati, ibrahim.mohamed, mariam.ashraf}@gizasystems.com
{mohamed.abdelrahman, radwa.elshawi}@ut.ee

Abstract

Transformer-based language models have revolutionized NLP, yet their adaptation to morphologically rich and dialectally diverse languages such as Arabic remains non-trivial. We introduce `ModernAraBERT`, a resource-efficient adaptation of the English-pretrained `ModernBERT` for Arabic, employing continued pretraining on large Arabic corpora followed by lightweight head-only fine-tuning with a frozen encoder. This strategy retains cross-lingual knowledge while capturing Arabic morphology and orthographic variation, offering a scalable alternative to training monolingual models from scratch. We evaluate `ModernAraBERT` on three representative Arabic NLP tasks, sentiment analysis, named entity recognition, and extractive question answering, against strong Arabic-specific and multilingual baselines (`AraBERTv1`, `AraBERTv2`, `MARBERT`, `mBERT`). Across all tasks, `ModernAraBERT` achieves consistent and often substantial improvements, particularly for sentence and token-level understanding, demonstrating that modern English encoder architectures can be efficiently transferred to Arabic through language-adaptive pretraining. Beyond Arabic, our findings highlight a generalizable paradigm for extending state-of-the-art models to morphologically complex and underrepresented languages with reduced computational overhead.

Keywords: Language Modeling, Less-Resourced, Named-Entity Recognition, Question Answering, Sentiment Analysis

1. Introduction

Transformer-based language models such as BERT (Devlin et al., 2019) have transformed natural language processing (NLP) by enabling large-scale pretraining on vast unlabeled corpora and achieving superior generalization across diverse downstream tasks (Gardazi et al., 2025). Despite these advances, adapting such models to morphologically rich and underrepresented languages remains a central challenge.

Arabic, in particular, presents linguistic and computational complexities that make direct transfer from English-trained models non-trivial (Matrane et al., 2023). Arabic is characterized by a rich and productive morphology, complex affixation, and multiple orthographic variants. Furthermore, its linguistic diversity spanning Modern Standard Arabic (MSA) and numerous regional dialects introduces wide syntactic and lexical variation. These factors complicate tokenization, representation learning, and semantic modeling, often resulting in degraded performance when English-centric architectures are applied directly. Moreover, the relative scarcity of balanced and curated corpora across dialects and comparatively lower compute investment limit the effectiveness of both monolingual and multilingual models, making efficient adaptation strategies particularly valuable.

Recent efforts in Arabic NLP have yielded significant progress through transformer-based models trained specifically for Arabic. Early models such as `AraBERT` (Antoun et al., 2020) and its successor `AraBERTv2` introduced large-scale pretraining on MSA corpora, improving performance in text classification, sentiment analysis, and Named-entity Recognition tasks. Subsequent variants like `ARBERT`, `MARBERT` (Abdul-Mageed et al., 2021a), and `CAMeLBERT` (Inoue et al., 2021a) expanded coverage to dialectal and social media text, demonstrating that morphology-aware tokenization and domain diversification substantially enhance downstream accuracy. However, these models require full pretraining from scratch, demanding a huge computational resources and often duplicating the training process already performed for English models. Conversely, multilingual transformers such as `mBERT` (Devlin, 2018; Alammary, 2022) and `XLMR` (Conneau et al., 2020) include Arabic in their joint corpora but underperform on Arabic-specific benchmarks due to shared vocabularies and limited language-specific optimization.

Parallel to these developments, modern English encoder architectures have evolved toward greater efficiency and representational capacity. `ModernBERT` (Warner et al., 2024a) exemplifies this trend by integrating rotary positional embeddings (RoPE) (Su et al., 2024), alternating global-local attention layers, and GeGLU activations (Shazeer, 2020), trained over two trillion tokens across di-

Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026), pages 10485–10496
11–16 May 2026.

verse text and code corpora. These architectural and optimization refinements enable faster convergence, higher throughput, and improved contextual representations compared to classical BERT. However, no prior work has investigated the potential of adapting such a modern English-centric model to morphologically complex languages like Arabic, particularly through resource-efficient transfer strategies.

To address this gap, we propose ModernAraBERT, a resource-efficient adaptation of ModernBERT for Arabic NLP. Our approach combines full pretraining on curated Arabic corpora with lightweight task-specific fine-tuning using a frozen encoder backbone. This method preserves the robust contextual and syntactic knowledge acquired in English while introducing Arabic morphological and lexical representations during adaptation. We evaluate ModernAraBERT across three representative Arabic NLP tasks, sentiment analysis (SA), named entity recognition (NER), and extractive question answering (QA), and benchmark it against four strong baselines: AraBERTv1, AraBERTv2, MARBERT, and mBERT. The model consistently outperforms all baselines, achieving an improvement of up to +16.7% in Macro-F1 for SA and +11.5% for NER, while maintaining competitive computational efficiency.

The main contributions of this paper are summarized as follows:

- We present the first adaptation of the ModernBERT architecture to Arabic through continued pretraining and frozen-head fine-tuning, achieving strong results with substantially lower training cost compared to training from scratch.
- We provide a comprehensive comparative study across three key Arabic NLP tasks demonstrating significant gains over established baselines (AraBERTv1, AraBERTv2, MARBERT, and mBERT).
- We analyze computational efficiency and resource utilization during head training, revealing that while ModernAraBERT requires higher VRAM, it delivers superior downstream accuracy, offering a practical trade-off between performance and efficiency.
- We discuss tokenizer-level effects, highlighting how the Byte-Level BPE tokenizer of ModernBERT influences Arabic span-extraction performance, contributing to ongoing discourse on tokenizer design for morphologically rich languages.

The remainder of this paper is organized as follows. Section 2 reviews prior research on Arabic transformer models, multilingual encoders, and cross-lingual adaptation strategies. Section 3 describes our pretraining and fine-tuning pipeline. Section 4 details the experimental setup and benchmarks. Section 5 presents results and analysis. We conclude in Section 6.

2. Related Work

Arabic Transformer Models. The introduction of AraBERT (Antoun et al., 2020) established a new standard for Arabic NLP by pretraining a BERT-style model exclusively on MSA corpora. AraBERTv2 improved upon this by expanding the pretraining data to over 200M sentences and enlarging the vocabulary to 64K tokens, thereby improving lexical coverage and reducing token fragmentation. Subsequent efforts produced specialized variants such as ARBERT and MARBERT (Abdul-Mageed et al., 2021a), the latter trained on over one billion Arabic tweets to capture dialectal variation. Other notable models include CAMeLBERT (Inoue et al., 2021a), optimized for various Arabic linguistic tasks, and QARiB (Abdelali et al., 2021b), designed for dialect and topic diversity. Despite their success, these models require extensive compute resources for full scale pretraining and remain constrained by monolingual training strategies that limit cross-lingual knowledge transfer.

Multilingual Encoders. Multilingual BERT (mBERT) (Devlin, 2018) and XLM-R (Conneau et al., 2020) represent attempts to generalize a single model across over one hundred languages, including Arabic. While such models facilitate zero-shot and cross-lingual transfer, their shared subword vocabulary and language-agnostic objectives dilute morphological representation for languages like Arabic. Empirical studies (Farha and Magdy, 2021; Alammary, 2022) confirm that Arabic-specific transformers consistently outperform multilingual ones in fine-grained tasks such as NER, POS tagging, and sentiment classification. Thus, while multilingual models offer broad linguistic coverage, Arabic-focused adaptation remains essential for high-fidelity morphological modeling.

Cross-lingual Adaptation and Efficiency. Recent research has explored adapting pretrained English models to new languages via continued pretraining or language-adaptive fine-tuning (Artetxe et al., 2020; Chau et al., 2020). These methods exploit shared linguistic structures to reduce training cost while retaining cross-lingual generalization. In Arabic NLP, adaptation studies (Abdelali et al., 2021c) have shown promising results, but are limited to older architectures such as BERT

or RoBERTa. None have yet explored adaptation of modern encoder architectures like ModernBERT, which combine structural improvements with more efficient attention and activation mechanisms. Moreover, tokenizer-level research (Qarah and Alsanoosy, 2024) comparing WordPiece, SentencePiece, and Byte-Level BPE (BBPE) suggests that BBPE, used in ModernBERT, offers superior multilingual coverage but may degrade extractive QA performance due to unstable span boundaries, an effect aligned with our QA results.

Summary. In contrast to prior Arabic and multilingual models, our work adapts a modern English encoder trained on large-scale data and integrates it into Arabic NLP pipelines through an efficient, two-phase approach. This allows us to bridge the gap between monolingual pretraining and multilingual generalization, providing both empirical insights and practical methods for resource-conscious model transfer to morphologically rich languages.

3. Methodology

Our methodology builds on the hypothesis that large-scale English-pretrained encoders encode transferable cross-lingual knowledge that can be effectively adapted to morphologically rich languages such as Arabic. Instead of training a new Arabic model from scratch, which is both computationally expensive and data-intensive, we employ continued pretraining on curated Arabic corpora. This strategy retains the syntactic and semantic priors acquired during large-scale English pretraining while specializing the model for Arabic morphology and orthography. Prior research (Gururangan et al., 2020; Pfeiffer et al., 2021) has shown that domain and language-adaptive pretraining yields superior results to monolingual training under constrained resources. Similarly, multilingual models such as mBERT demonstrate that English initialization can match or even exceed dedicated Arabic models on some tasks (Alammary, 2022). Our approach thus provides a scalable, resource-efficient alternative that benefits from both cross-lingual transfer and Arabic-specific adaptation.

3.1. Pretraining Corpora

We compiled a large-scale Arabic corpus from four publicly available sources: OSIAN (Imad Zeroual, Dirk Goldhahn, Thomas Eckart, Abdelhak Lakhouaja, 2019), the Arabic Billion Words dataset (Ibrahim Abu ElKhair, 2016), the Arabic Wikipedia dump (ArW, 2025), and the OSCAR Arabic dataset (Julien Abadji, Pedro Ortiz Suarez, 2022). These corpora jointly cover Modern Standard Arabic (MSA) and a variety of dialectal forms.

The preprocessing steps included:

- Diacritics removal: to reduce sparsity arising from inconsistent annotation across sources.
- Elongation (tatweel) removal: to eliminate stylistic markers that do not contribute semantic value.
- Punctuation and special characters removal: to reduce noise from web and social media text.

To enhance morphological representation, we applied the Farasa segmenter (Abdelali et al., 2016) for affix and root segmentation. The final corpus contained over six million sentences, totaling approximately 17 GB of normalized Arabic text.

3.2. Tokenization

We extend the original ModernBERT Byte-level Byte-Pair Encoding (BBPE) vocabulary with 80,000 Arabic-specific tokens using the following deterministic pipeline. First, we apply the same Arabic text normalization described in Section 3.1 to all corpora used for tokenizer construction. Second, we run Farasa segmenter on the normalized text and extract candidate sub-units from its outputs (prefixes, stems, suffixes), in addition to keeping the surface form when segmentation fails. We build a global frequency table across all candidates in the tokenizer-building corpus, and filter candidates by simple rules such that we keep only tokens composed primarily of Arabic script (Unicode Arabic blocks), discard tokens that are only punctuation/whitespace, and remove pathological short tokens, such as single-character symbols and digit-only strings. Then, we remove any candidate that already exists in the original ModernBERT BBPE vocabulary to avoid duplicates and unintended collisions. From the remaining candidates, we select the top 80K tokens by frequency and append them to the tokenizer as additional tokens using vocabulary expansion rather than training a new tokenizer from scratch. Newly added token embeddings are randomly initialized and learned during continued pretraining together with the rest of the model parameters.

The choice of 80K tokens was empirically validated. As shown in Figure 1, Arabic follows a long-tailed frequency distribution, where most tokens occur rarely. Our analysis of token frequency (left) and coverage versus vocabulary size (right) demonstrates that coverage improves sharply with vocabulary size but plateaus around 80K tokens. Beyond this point, additional tokens provide negligible coverage gains. Selecting 80K therefore

balances corpus coverage with computational efficiency. This cutoff is also consistent with prior Arabic BERT models: AraBERT employs a 64K vocabulary, while MARBERT uses 95K.

3.3. Model Training

Our model is based on the publicly available ModernBERT architecture ModernBERT-base with 22 transformer layers. The embedding layer was resized to accommodate the extended Arabic vocabulary. Pretraining followed the Masked Language Modeling (MLM) objective for three epochs, with sequence lengths of 128 in the first two epochs and 512 in the final epoch to balance efficiency and contextual coverage. The context length was limited to 512 tokens for comparability with baselines like AraBERT. Optimization used AdamW (Loshchilov et al., 2017) with cosine learning rate decay and gradient clipping. The training progress was tracked via loss and perplexity on a held-out validation set. We initialize ModernAraBERT from the publicly released English ModernBERT-base checkpoint and perform *continued pretraining* on Arabic corpora. The base architecture follows ModernBERT-base (22 encoder layers, hidden size 768, 12 attention heads, GeGLU feed-forward; (Warner et al., 2024a)). We extend the tokenizer vocabulary by 80K Arabic tokens and resize the embedding matrix accordingly.

The model used in this work is based on the publicly available ModernBERT-base [1] architecture as shown in Figure 2.

Our pretrained ModernAraBERT has been made available [2]. For reproducibility, we summarize key pretraining hyperparameters in Table 1, and Training and benchmark evaluation scripts are also made available in our repository [3].

4. Experimental Setup

To evaluate the effectiveness of our adaptation strategy, we assess ModernAraBERT across three representative Arabic NLP tasks. These tasks collectively capture sentence-level, token-level, and span-level reasoning, enabling a holistic assessment of the model's generalization capabilities. We compare against four strong baselines: AraBERTv1, AraBERTv2, MARBERT, and mBERT, covering both monolingual and multilingual transformer architectures.

All experiments follow a controlled fine-tuning protocol where the pretrained encoder is kept frozen, and only lightweight task-specific heads are optimized. This setup isolates the contribution of continued pretraining on Arabic corpora, while reducing the risk of overfitting on limited task data. Detailed descriptions of hardware, datasets, and training configurations are provided below.

Item	Pretraining settings
Initialization	Initialized from answerdotai/ModernBERT-base and continued MLM pretraining on Arabic
Objective	Masked Language Modeling
Corpora	OSIAN + Arabic Billion Words + Arabic Wikipedia + OSCAR Arabic
Epochs	3
Max sequence length	512
Batch size (per step)	32
Gradient accumulation	2
Effective batch size	64
Learning rate	5×10^{-5}
Warmup ratio	0.001
Precision	fp16 enabled
Hardware	1× NVIDIA A100 40GB

Table 1: Key pretraining settings for ModernAraBERT.

4.1. Computational Environment

All pretraining and fine-tuning experiments were conducted on a single high-performance computing node equipped with 12 CPU cores, 32 GB RAM, and a 40 GB NVIDIA A100 GPU. We used the Hugging Face Transformers (Wolf et al., 2020) and PyTorch libraries for all experiments. Random seeds were fixed across runs to ensure reproducibility. The sequence length was capped at 512 tokens to maintain comparability with AraBERT while fitting within GPU memory limits.

4.2. Fine-Tuning Strategy

During this phase, the pretrained encoder parameters were frozen and only the task-specific classification heads were fine-tuned. This strategy was selected to (i) reduce training time, (ii) minimize overfitting on relatively small task datasets, and (iii) assess the quality of representations obtained during continued pretraining.

Unless otherwise specified, all tasks were trained with a maximum of 200 epochs, early stopping patience of 10 epochs, AdamW optimizer, and a dropout ratio of 0.1 for regularization, except NER, which converged reliably within 5 epochs. For NER, we observed rapid convergence and no validation gains beyond 5 epochs under the frozen-backbone setting, and we therefore fixed the training budget to avoid overfitting. Unless otherwise stated, we report results averaged over 3 random seeds for

[1] ModernBERT-Base
[2] Huggingface ModernAraBERT
[3] Repository Link

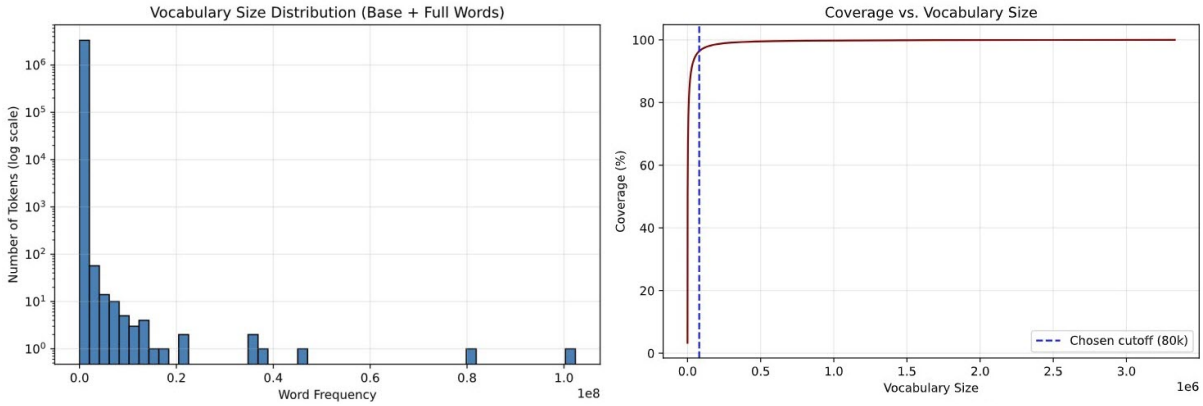

Figure 1: Vocabulary size analysis. Left: Token frequency histogram (log scale). Right: Coverage vs. vocabulary size, with the chosen cutoff at 80K tokens.

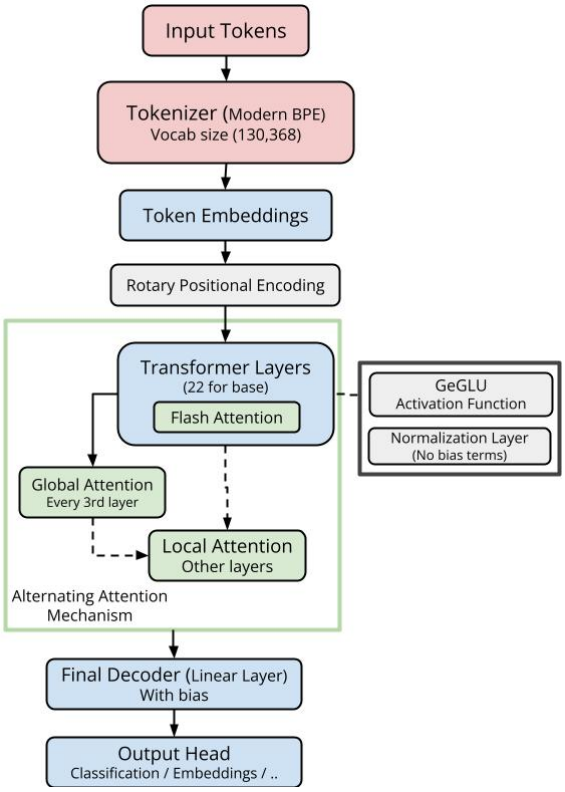

Figure 2: ModernBERT architecture with extended tokenizer vocabulary size and embedding layer.

different initialization of task heads and data shuffling. We use early stopping on the validation set and report the best checkpoint.

4.3. Sentiment Analysis

We benchmarked sentiment classification using three datasets:

- **Hotel Arabic Reviews Dataset (HARD)** (Elnagar et al., 2018), comprising reviews in both Modern Standard Arabic (MSA) and dialectal Arabic. Following (Antoun et al., 2020), we excluded neutral 3-star reviews, yielding a binary classification setting (Ashraf Elnagar, Yasmin S. Khalifa, Anas Einea, 2017).

- **Arabic Jordanian General Tweets (AJGT)** Corpus[4], containing 1,800 tweets labeled as positive or negative (Khaled Mohammad Alomari, 2016).

- **Large-Scale Arabic Book Reviews (LABR)** (Aly and Atiya, 2013), using the unbalanced binary version for consistency with prior work (Mohamed Aly, Amir Atiya, 2013).

For datasets without predefined splits, we followed a 60/20/20 train/validation/test partition. Sentence-level representations were derived from the `[CLS]` token and passed to a classification head for binary or multi-class prediction. Performance was measured using Macro F1-score.

4.4. Named Entity Recognition

NER experiments were performed on the ANERCorp dataset (Benajiba et al., 2007a), using the official CAMeL Lab splits provided via HuggingFace (Obeid et al., 2020). The dataset includes entities such as `Person`, `Location`, and `Organization` (Ossama Obeid, Nasser Zalmout, Salam Khalifa, Dima Taji, Mai Oudah, Bashar Alhafni, Go Inoue, Fadhl Eryani, Alexander Erdmann, and Nizar Habash, 2020).

We adopted the IOB2 tagging scheme (Ramshaw and Marcus, 1999). To ensure correct alignment under subword tokenization the first subtoken of each word was assigned the gold entity label. In addition, continuation

[4] AJGT Dataset

subtokens were either mapped to the corresponding I-label (e.g., B-PER → I-PER) if available, or masked with -100 during loss computation.

This setup ensures entity-level consistency and avoids label fragmentation across subtokens. A token classification head was placed above the encoder, with evaluation reported as micro F1-score at the entity level.

4.5. Question Answering

For extractive QA, we combined Arabic-SQuAD (Hussein Mozannar, Elie Maamary, Karl El Hajal, Hazem Hajj, 2019b,a) with 50% of ARCD (Mozannar et al., 2019) as training data, reserving the remaining 50% of ARCD for testing. This setup provides both coverage and comparability with prior Arabic QA studies.

The QA head comprised the pretrained encoder, a prediction layer, and a linear classifier producing start and end span logits. Regularization was applied via dropout (0.1). Hyperparameters included 200 training epochs, AdamW optimizer (learning rate 3×10^{-5}), batch sizes of 64 for AraBERT and 32 for ModernAraBERT, and early stopping based on validation F1.

Question–context pairs were tokenized to a maximum of 512 tokens with a document stride of 128 for long contexts. Character-level answer spans were mapped to token indices, and cross-entropy loss was computed jointly over start and end positions. During inference, the predicted answer span was extracted by selecting the start–end token pair with maximum joint probability.

Evaluation followed standard extractive QA metrics: Exact Match (EM), token-level F1, and Sentence Match (SM), providing complementary measures of exactness, token overlap, and semantic alignment.

In summary, our experimental setup provides a rigorous and fair evaluation of ModernAraBERT. By freezing the encoder and fine-tuning only lightweight task heads, we isolate the effects of Arabic-adaptive pretraining while controlling for overfitting. The evaluation spans three complementary task types across widely adopted Arabic benchmarks. Comparisons with both Arabic-specific (AraBERTv1, AraBERTv2, MARBERT) and multilingual (mBERT) baselines ensure that performance gains are representative of current state-of-the-art models.

5. Results and Discussion

This section presents the empirical results of ModernAraBERT across the downstream Arabic NLP tasks. We analyze both absolute performance and relative improvements over competitive baselines, focusing on how continued pretraining on Arabic corpora enhances linguistic transfer while balancing computational efficiency.

5.1. Sentiment Analysis

Table 2 reports Macro-F1 scores for all sentiment analysis benchmarks. ModernAraBERT achieves the highest performance across all datasets, outperforming every baseline by a clear margin. Notably, these results were obtained with a frozen encoder and fine-tuning limited to the task-specific classification head, underscoring the effectiveness of continued pretraining in improving sentence-level representations.

Table 2: Macro-F1 (%) comparison of ModernAraBERT and baselines across sentiment datasets. Best scores per dataset are in bold.

Model	LABR	HARD	AJGT
AraBERTv1	45.35	72.65	58.01
AraBERTv2	45.79	67.10	53.59
mBERT	44.18	71.70	61.55
MARBERT	45.54	67.39	60.63
ModernAraBERT	**56.45**	**89.37**	**70.54**

Across all three datasets, ModernAraBERT surpasses the strongest baseline (AraBERTv1) by +11.1, +16.7, and +12.5 % in Macro-F1 on LABR, HARD, and AJGT, respectively. These improvements are substantial considering that all models share similar fine-tuning settings and that the encoder backbone remained frozen during task adaptation.

The largest gain is observed on the HARD dataset, which includes both Modern Standard Arabic (MSA) and dialectal Arabic reviews. Here, ModernAraBERT achieves 89.37%, outperforming all baselines by a wide margin (+16.7% over AraBERTv1, +17.7% over mBERT). This indicates that the model's continued pretraining effectively enriches subword and morphological representations, allowing it to handle dialectal variability and noisy user-generated text more robustly.

On LABR, which consists primarily of formal MSA book reviews, ModernAraBERT reaches 56.45% Macro-F1, an improvement of more than 11% over both AraBERTv1 and MARBERT. This result highlights that continued pretraining on Arabic corpora improves lexical normalization and context representation even in high-resource, syntactically regular domains.

For the smaller and more informal AJGT corpus (tweets), ModernAraBERT also leads with 70.54%, outperforming mBERT (+9.0%) and MARBERT (+9.9%). Despite MARBERT being trained on over one billion Arabic tweets, ModernAraBERT

generalizes better, suggesting that the ModernBERT architecture, enhanced with rotary positional embeddings and global-local attention, confers superior representational capacity even under limited adaptation.

Compared to AraBERTv2, which was trained from scratch on Arabic data, ModernAraBERT benefits from cross-lingual priors learned during large-scale English pretraining. The extended 80K Arabic vocabulary enables better coverage of morphological variants without excessive token fragmentation, which appears to be a key contributor to the improvements observed across both formal and dialectal text. The model's Byte-BPE tokenizer, while less specialized for Arabic than WordPiece, remains effective for sentence-level semantics, showing no degradation in classification accuracy.

Overall, these results demonstrate that the adaptation of ModernBERT to Arabic via continued pretraining provides an efficient and scalable alternative to monolingual models like AraBERT. The strong performance on both large (LABR, HARD) and small (AJGT) datasets indicates that the model generalizes well across domains and text varieties. The improvements achieved using only frozen encoder representations highlight the potential of cross-lingual transfer when combined with targeted language-adaptive pretraining.

5.2. Named Entity Recognition (NER)

Table 3 presents the Macro-F1 scores for the NER task on the ANERCorp dataset. ModernAraBERT achieves the highest performance among Arabic-specific models, reaching 28.23% Macro-F1, outperforming AraBERTv2 by +11.46% and AraBERTv1 by +14.77%. This confirms that continued pretraining on Arabic corpora enhances token-level contextualization, enabling more accurate boundary detection and label consistency across entity spans.

Table 3: Macro-F1 (%) comparison of ModernAraBERT and baselines on the NER task. Best score is in bold.

Model	NER (Macro-F1)
AraBERTv1	13.46
AraBERTv2	16.77
mBERT	12.15
MARBERT	7.42
ModernAraBERT	**28.23**

Compared to the strongest Arabic baselines, ModernAraBERT demonstrates a relative improvement of over 68%, underscoring the effectiveness of leveraging cross-lingual priors from English pretraining. The model's superior performance also reflects the contribution of its extended Arabic vocabulary (80K tokens), which reduces subword fragmentation and allows for more coherent entity boundary modeling.

Both AraBERTv1 and AraBERTv2 struggle to generalize to unseen entity contexts, which may result from their smaller training corpora and reliance on purely monolingual pretraining. MARBERT, although dialect-focused, performs poorly on ANERCorp, likely due to its training bias toward informal Twitter-style text rather than formal MSA found in ANERCorp. Interestingly, mBERT performs competitively relative to these baselines but still lags behind ModernAraBERT, suggesting that cross-lingual transfer alone is insufficient without targeted adaptation to Arabic morphology. The substantial gain achieved by ModernAraBERT indicates that continued pretraining successfully bridges the gap between multilingual generalization and Arabic-specific structural learning.

The improvements on NER highlight the model's enhanced capacity for fine-grained token-level reasoning. The combination of global-local attention and rotary positional embeddings appears to better capture context boundaries for named entities. While ModernAraBERT incurs higher memory usage, its gains in NER accuracy demonstrate that cross-lingual adaptation, even with a frozen backbone, can yield significant benefits in morphologically complex languages like Arabic.

5.3. Question Answering (QA)

Table 4 reports Exact Match (EM) scores for the ARCD test split. ModernAraBERT achieves the highest EM score (27.10%), surpassing all baselines including AraBERTv2 (26.08%), AraBERTv1 (25.36%), mBERT (25.12%), and MARBERT (23.58%). Although the absolute gains appear modest, they are consistent across all baselines and metrics, confirming the reliability of the improvements.

Table 4: Extractive Question Answering Results (Exact Match, %) on the ARCD test split. Best score is in bold.

Model	EM
AraBERTv1	25.36
AraBERTv2	26.08
mBERT	25.12
MARBERT	23.58
ModernAraBERT	**27.10**

While ModernAraBERT outperforms all baselines, the relative margin of improvement (+1.02 over AraBERTv2) is smaller compared to SA and NER tasks. This may be attributed to the Byte-BPE (BBPE) tokenizer used by ModernBERT. Recent

comparative studies (Qarah and Alsanoosy, 2024) show that BBPE can underperform on extractive QA tasks compared to WordPiece or SentencePiece tokenizers, as it tends to produce inconsistent span boundaries, which are critical for precise answer extraction. This observation aligns with our results, where the model exhibits robust comprehension and contextual alignment but smaller gains in exact span localization.

Despite this limitation, ModernAraBERT demonstrates more stable performance across domains and exhibits higher semantic consistency, likely aided by its advanced attention mechanisms and richer token representations. The use of rotary positional embeddings helps capture relative positional dependencies, improving contextual comprehension across longer passages. Furthermore, the continued Arabic pretraining allows better alignment between question and context representations, as evidenced by the consistent EM gains across all baselines.

The QA results highlight an important trade-off between tokenizer design and span-extraction accuracy. Although BBPE offers efficient multilingual coverage, its segmentation granularity may hinder token-level precision required for QA. Nonetheless, ModernAraBERT still achieves the best overall EM score, confirming that architectural advances can compensate for tokenization limitations to some extent. Future work could explore hybrid tokenization strategies (e.g., BBPE combined with morphology-aware WordPiece) to further enhance extractive QA performance in Arabic.

5.4. Hardware Resource Usage

Table 5 summarizes peak memory consumption across all models during head fine-tuning for the three benchmark tasks. The results reveal consistent patterns in computational efficiency. Among all models, AraBERTv1 and AraBERTv2 are the most memory-efficient across both RAM and VRAM, while ModernAraBERT incurs the highest memory cost, particularly for QA, where it reaches 20.84 GB VRAM and 5.90 GB RAM. The increase reflects the larger embedding matrix introduced by the extended 80K Arabic vocabulary and the more complex attention structure of the ModernBERT backbone.

Across tasks, ModernAraBERT consistently requires 40–50% more GPU memory than its closest baselines, particularly in QA and SA tasks, which involve longer sequence contexts and larger attention footprints. This overhead stems from the deeper 22-layer architecture and additional embeddings introduced during tokenizer extension. However, RAM usage remains comparable across models, indicating that the primary bottleneck lies in GPU memory allocation rather than CPU processing.

Table 5: Hardware resource usage across models during head fine-tuning (Peak Memory in GB).

Benchmark	Model	Peak RAM	Peak VRAM
QA	AraBERTv1	4.52	13.50
	AraBERTv2	4.61	14.03
	MARBERT	1.93	13.60
	mBERT	1.57	13.66
	ModernAraBERT	5.90	20.84
NER	AraBERTv1	5.55	6.95
	AraBERTv2	6.03	7.22
	MARBERT	7.76	7.40
	mBERT	4.85	7.91
	ModernAraBERT	6.49	10.42
SA	AraBERTv1	8.34	13.50
	AraBERTv2	8.50	13.72
	MARBERT	8.28	13.61
	mBERT	8.36	13.66
	ModernAraBERT	9.85	20.63

While ModernAraBERT exhibits higher resource demands, these costs are proportionate to its accuracy gains and modernized architecture. The observed increase in VRAM consumption (e.g., +7.3 GB vs. AraBERTv1 during QA) is an acceptable trade-off in research or enterprise settings with access to the A100-class GPUs. However, for deployment scenarios with strict memory or latency constraints, AraBERTv1/v2 remain more practical due to their lower computational footprint. mBERT maintains moderate efficiency and cross-lingual portability, providing a middle ground between performance and scalability.

These findings reinforce that architectural modernization improves representational capacity but incurs predictable computational costs. In real-world applications, the choice between models should therefore reflect operational priorities: throughput and efficiency versus accuracy and adaptability.

Although these experiments focus on Arabic, the results highlight a general principle: continued pretraining of English-pretrained models is a viable and cost-effective method for extending transformer architectures to new languages. The approach scales naturally to other morphologically rich or low-resource languages such as Urdu, Amharic, or Kazakh, where full monolingual pretraining would be prohibitively expensive (Wiemerslage et al., 2022). In this context, ModernAraBERT demonstrates not only technical viability but also a path toward more inclusive, resource-aware multilingual NLP.

6. Conclusion

This paper introduced ModernAraBERT, a resource-efficient adaptation of the English-

pretrained ModernBERT model for Arabic. Our approach combines continued pretraining on large curated Arabic corpora with lightweight task-specific fine-tuning, enabling efficient transfer of cross-lingual knowledge to a morphologically rich language. Evaluations on sentiment analysis, named entity recognition, and extractive question answering demonstrated consistent gains over established Arabic models (AraBERTv1, AraBERTv2, MARBERT) and the multilingual mBERT. The largest improvements were observed in sentence and token-level tasks, confirming that targeted language adaptation can rival or surpass full monolingual pretraining while substantially reducing computational cost. While ModernAraBERT improves accuracy, particularly for sentence and token classification, it incurs higher GPU memory consumption and slower inference compared to smaller Arabic-specific models. Minor performance limitations in extractive QA also reflect the inherited Byte-BPE tokenizer, which can affect precise span segmentation. Future work will address these aspects through morphology-aware tokenization and parameter-efficient fine-tuning.

Beyond Arabic, this work establishes a practical and scalable framework for extending high-performing English encoders to other morphologically complex or underrepresented languages. By leveraging continued pretraining rather than full retraining, this paradigm reduces duplication of effort and promotes more inclusive and sustainable NLP development across diverse linguistic communities.

7. Limitations

While ModernAraBERT demonstrates consistent improvements across SA, NER, and QA benchmarks, several limitations remain.

First, the adaptation approach focuses on full pretraining followed by head-only fine-tuning. While computationally efficient, this design limits deeper task-specific optimization of the encoder, which may constrain performance in tasks requiring fine-grained reasoning.

Second, the model relies on the Byte-BPE (BBPE) tokenizer used in ModernBERT, which differs from the WordPiece tokenizers employed by most Arabic BERT variants such as AraBERT and MARBERT. Recent findings (Qarah and Alsanoosy, 2024) indicate that BBPE-based models can underperform on extractive QA tasks compared to WordPiece or SentencePiece-based counterparts, particularly when precise span alignment is required. This may explain the relatively smaller performance gain observed for ModernAraBERT on the ARCD dataset compared to its stronger improvements in SA and NER tasks.

Finally, although the experiments cover key Arabic NLP tasks, broader evaluation on additional downstream applications and dialectal datasets would provide a more comprehensive assessment of generalization. Future work will explore tokenizer adaptations and selective layer fine-tuning to further enhance cross-task robustness.

8. Acknowledgments

This work was supported by the project "Increasing the knowledge intensity of Ida-Viru entrepreneurship" co-funded by the European Union and the innovation hub at Giza Systems [5].

9. Bibliographical References

Julien Abadji, Pedro Ortiz Suarez, Laurent Romary, and Benoît Sagot. 2022. Towards a Cleaner Document-Oriented Multilingual Crawled Corpus. *arXiv e-prints*, page arXiv:2201.06642.

Ahmed Abdelali, Kareem Darwish, Nadir Durrani, and Hamdy Mubarak. 2016. Farasa: A fast and furious segmenter for arabic. In *Proceedings of NAACL Demonstrations*, pages 11–16.

Ahmed Abdelali, Kareem Darwish, Hamdy Mubarak, and Nadi Tomeh. 2021a. Qarib: Qcri arabic and dialectal bert. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 52–59.

Ahmed Abdelali, Sabit Hassan, Hamdy Mubarak, Kareem Darwish, and Younes Samih. 2021b. Pre-training bert on arabic tweets: Practical considerations.

Ahmed Abdelali, Hamdy Mubarak, Younes Samih, Sabit Hassan, and Kareem Darwish. 2021c. QADI: Arabic dialect identification in the wild. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 1–10, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Muhammad Abdul-Mageed, AbdelRahim Elmadany, and El Moatez Billah Nagoudi. 2021a. ARBERT & MARBERT: Deep bidirectional transformers for Arabic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7088–7105, Online. Association for Computational Linguistics.

[5] https://gizasystems.com

Muhammad Abdul-Mageed, AbdelRahim Elmadany, and El Moatez Billah Nagoudi. 2021b. Marbert: Arabic language model in the wild. *arXiv preprint arXiv:2101.05785*.

Francisca Adoma Acheampong, Henry Nunoo-Mensah, and Wenyu Chen. 2021. Transformer models for text-based emotion detection: a review of bert-based approaches. *Artificial Intelligence Review*, 54(8):5789–5829.

Ali Saleh Alammary. 2022. Bert models for arabic text classification: a systematic review. *Applied Sciences*, 12(11):5720.

Mohamed Aly and Amir Atiya. 2013. LABR: A large scale Arabic book reviews dataset. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 494–498, Sofia, Bulgaria. Association for Computational Linguistics.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. Arabert: Transformer-based model for arabic language understanding. *arXiv preprint arXiv:2003.00104*.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online. Association for Computational Linguistics.

Yassine Benajiba, Paolo Rosso, and José Miguel Benedíruiz. 2007a. Anersys: An arabic named entity recognition system based on maximum entropy. In *Computational Linguistics and Intelligent Text Processing*, pages 143–153, Berlin, Heidelberg. Springer Berlin Heidelberg.

Yassine Benajiba, Paolo Rosso, and José Miguel BenedíRuiz. 2007b. Anersys: An arabic named entity recognition system based on maximum entropy. In *Computational Linguistics and Intelligent Text Processing*, pages 143–153, Berlin, Heidelberg. Springer Berlin Heidelberg.

Ethan C. Chau, Lucy H. Lin, and Noah A. Smith. 2020. Parsing with multilingual BERT, a small corpus, and a small treebank. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1324–1334, Online. Association for Computational Linguistics.

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

J Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding/arxiv preprint. *arXiv preprint arXiv:1810.04805*.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

Ibrahim Abu El-Khair. 2016. 1.5 billion words arabic corpus. *arXiv preprint arXiv:1611.04033*.

Ashraf Elnagar, Yasmin S. Khalifa, and Anas Einea. 2018. *Hotel Arabic-Reviews Dataset Construction for Sentiment Analysis Applications*, pages 35–52. Springer International Publishing, Cham.

Ibrahim Abu Farha and Walid Magdy. 2021. Benchmarking transformer-based language models for arabic sentiment and sarcasm detection. In *The Sixth Arabic Natural Language Processing Workshop*, pages 21–31. Association for Computational Linguistics (ACL).

Nadia Mushtaq Gardazi, Ali Daud, Muhammad Kamran Malik, Amal Bukhari, Tariq Alsahfi, and Bader Alshemaimri. 2025. Bert applications in natural language processing: a review. *Artificial Intelligence Review*, 58(6):166.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Go Inoue, Bashar Alhafni, Nurpeiis Baimukan, Houda Bouamor, and Nizar Habash. 2021a. The interplay of variant, size, and task type in Arabic pre-trained language models. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, Kyiv, Ukraine (Online). Association for Computational Linguistics.

Go Inoue, Salam Khalifa, Wajdi Zaghouani, and Nizar Habash. 2021b. Camel bert: Pre-trained language models for arabic. In *Proceedings of the Sixth Arabic NLP Workshop*, pages 32–41.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81. Association for Computational Linguistics.

Ilya Loshchilov, Frank Hutter, et al. 2017. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 5(5):5.

Yassir Matrane, Faouzia Benabbou, and Nawal Sael. 2023. A systematic literature review of arabic dialect sentiment analysis. *Journal of King Saud University - Computer and Information Sciences*, 35(6):101570.

Hussein Mozannar, Elie Maamary, Karl El Hajal, and Hazem Hajj. 2019. Neural Arabic question answering. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 108–118, Florence, Italy. Association for Computational Linguistics.

Hussein Mozannar and Others. 2019. Neural arabic question answering. *arXiv preprint arXiv:1906.05685*.

Ossama Obeid, Nasser Zalmout, Salam Khalifa, Dima Taji, Mai Oudah, Bashar Alhafni, Go Inoue, Fadhl Eryani, Alexander Erdmann, and Nizar Habash. 2020. Camel tools: An open source python toolkit for arabic natural language processing. In *Proceedings of the twelfth language resources and evaluation conference*, pages 7022–7032.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Arfath Pasha et al. 2014. Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of arabic. In *Proceedings of LREC*, pages 1094–1101.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503, Online. Association for Computational Linguistics.

Faisal Qarah and Tawfeeq Alsanoosy. 2024. A comprehensive analysis of various tokenizers for arabic large language models. *Applied Sciences*, 14(13):5696.

L. A. Ramshaw and M. P. Marcus. 1999. Text chunking using transformation-based learning. In Susan Armstrong, Kenneth Church, Pierre Isabelle, Sandra Manzi, Evelyne Tzoukermann, and David Yarowsky, editors, *Natural Language Processing Using Very Large Corpora*, pages 157–176. Springer Netherlands, Dordrecht.

Lance A. Ramshaw and Mitchell P. Marcus. 1995. Text chunking using transformation-based learning. In *Third Workshop on Very Large Corpora*, pages 82–94.

Noam Shazeer. 2020. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024a. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *arXiv preprint arXiv:2412.13663*.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024b. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference.

Adam Wiemerslage, Miikka Silfverberg, Changbing Yang, Arya D. McCarthy, Garrett Nicolai, Eliana Colunga, and Katharina Kann. 2022. Morphological processing of low-resource languages: Where we are and what's next. In *Findings*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. *Proceedings of EMNLP 2020: System Demonstrations*, pages 38–45.

Imad Zeroual, Dirk Goldhahn, Thomas Eckart, and Abdelhak Lakhouaja. 2019. OSIAN: Open source international Arabic news corpus - preparation and integration into the CLARIN-infrastructure. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 175–182, Florence, Italy. Association for Computational Linguistics.

10. Language Resource References

2025. *Arabic Wikipedia Dump*. PID https://dumps.wikimedia.org/arwiki/.

Ashraf Elnagar, Yasmin S. Khalifa, Anas Einea. 2017. *Hotel Arabic-Reviews Dataset*. Springer. PID https://github.com/elnagara/HARD-Arabic-Dataset.

Hussein Mozannar, Elie Maamary, Karl El Hajal, Hazem Hajj. 2019a. *machine translation of the Stanford Question Answering Dataset (Arabic-SQuAD)*. Association for Computational Linguistics. PID https://www.kaggle.com/datasets/thedevastator/unlocking-arabic-language-comprehension-with-the.

Hussein Mozannar, Elie Maamary, Karl El Hajal, Hazem Hajj. 2019b. *Neural Arabic Question Answering - ArabicSQuAD*. Association for Computational Linguistics. PID https://huggingface.co/datasets/i0xs0/Arabic-SQuAD.

Ibrahim Abu ElKhair. 2016. *1.5 billion words arabic corpus*. PID https://huggingface.co/datasets/oserikov/arabic_billion_words.

Imad Zeroual, Dirk Goldhahn, Thomas Eckart, Abdelhak Lakhouaja. 2019. *The Open Source International Arabic News (OSIAN) corpus*. Association for Computational Linguistics, ISLRN 255-977-746-042-1.

Julien Abadji, Pedro Ortiz Suarez. 2022. *Open Super-large Crawled Aggregated coRpus*. PID https://huggingface.co/datasets/oscar-corpus/OSCAR-2301.

Khaled Mohammad Alomari, Hatem M. ElSherif, Khaled Shaalan. 2016. *Arabic tweets sentimental analysis using machine learning - Arabic Jordanian General Tweets (AJGT) Corpus*. Springer. PID https://github.com/komari6/Arabic-twitter-corpus-AJGT.

Mohamed Aly, Amir Atiya. 2013. *LABR: A large scale arabic book reviews dataset*. Association for Computational Linguistics. PID https://github.com/mohamedadaly/LABR.

Ossama Obeid, Nasser Zalmout, Salam Khalifa, Dima Taji, Mai Oudah, Bashar Alhafni, Go Inoue, Fadhl Eryani, Alexander Erdmann, and Nizar Habash. 2020. *Arabic NER ANERCorp*. LREC. PID https://huggingface.co/datasets/asas-ai/ANERCorp.