Efficient Adaptation of English Language Models for Low-Resource and Morphologically Rich Languages: The Case of Arabic

Anonymous submission

Abstract

Transformer-based language models have revolutionized natural language processing, yet their adaptation to morphologically rich and underrepresented languages remains challenging. In this work, we introduce `ModernAraBERT`, a resource-efficient adaptation of the English-pretrained `ModernBERT` to Arabic. Our approach leverages continued pretraining on large curated Arabic corpora, followed by lightweight task-specific fine-tuning with frozen encoder backbones. This strategy preserves cross-lingual knowledge while effectively capturing Arabic morphology, offering a practical alternative to training monolingual models from scratch. We evaluate `ModernAraBERT` across three representative Arabic NLP tasks—sentiment analysis (SA), named entity recognition (NER), and extractive question answering (QA) using widely adopted benchmarks. Results show consistent improvements over the established `AraBERT v1` baseline and competitive performance relative to `mBERT`. Notably, `ModernAraBERT` yields gains of up to ≈17% in SA, and significant improvements in NER and span-based QA metrics. Analysis further highlights trade-offs between accuracy and efficiency: while `ModernAraBERT` requires higher GPU memory than `AraBERT`, it provides superior downstream accuracy, especially for sentence and span level tasks. Beyond Arabic, our findings demonstrate that language-adaptive pretraining offers a scalable framework for extending state-of-the-art English models to other morphologically rich or low-resource languages, thereby reducing duplication of effort and broadening NLP inclusivity.

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

1. Introduction

Transformer encoder models such as BERT (Devlin et al., 2019) have revolutionized the approach to language processing tasks, especially within the English language. These models are characterized by their robustness, versatility, and success in numerous NLP applications and language representation tasks (Gardazi et al., 2025). The Arabic language presents distinctive challenges: it is morphologically rich, replete with inflections, and encompasses various dialects. Consequently, adapting existing models trained in English to Arabic entails unique difficulties (Matrane et al., 2023).

Recent developments in Arabic NLP have been notably driven by the integration of deep learning and transformer-based architectures. The release of `AraBERT` as a pretrained model for Modern Standard Arabic (MSA) marked the inception of many subsequent Arabic BERT-based models (Antoun et al., 2020). Some focus specifically on MSA, such as `CAMeLBERT` (Inoue et al., 2021a) and `ARBERT` (Abdul-Mageed et al., 2021a), which utilized carefully curated data. Others target dialectical variations, including `MARBERT` (Abdul-Mageed et al., 2021a) and `QARiB` (Abdelali et al., 2021b). Such models, though effective, often incur high computational costs and require training from scratch or extensive adaptation without efficient transfer or reduction in training duplication. Another widely used model is multilingual BERT (`mBERT`) (Devlin, 2018; Alammary, 2022). Unlike the Arabic-specific variants, `mBERT` was pretrained on Wikipedia data covering 104 languages, including Modern Standard Arabic. Its architecture consists of 12 transformer layers with 768 hidden units each, 12 self-attention heads, and approximately 110M trainable parameters.

Recently, significant advancements have been made through the introduction of `ModernBERT` (Warner et al., 2024a), which modernizes encoder-only transformer architectures. `ModernBERT` addresses the critical limitations of previous models by training on an extensive dataset of 2 trillion tokens. Its architecture integrates several efficiency and performance improvements, such as rotary positional embeddings (RoPE) (Su et al., 2024), alternating global-local attention layers, and the utilization of GeGLU activation functions (Shazeer, 2020). These architectural enhancements, combined with a modern tokenizer optimized for diverse textual and code-related data, enable `ModernBERT` to achieve state-of-the-art performance in a broad spectrum of classification and retrieval tasks, thus providing an optimal foundation for adapting advanced language models to languages with unique linguistic challenges, such as Arabic.

The main contributions of this paper can be summarized as follows.

- We propose a resource-efficient strategy to extend high-performing English language models to Arabic by conducting efficient pretraining on curated Arabic corpora, thus providing a practical alternative by adapting an existing English-pretrained model to Arabic, instead of developing an entirely new Arabic-specific model from scratch.

- We present ModernAraBERT, an adapted version of the state-of-the-art English ModernBERT model, specifically fine-tuned and optimized for Arabic NLP tasks.

- We empirically evaluate ModernAraBERT across three essential Arabic NLP benchmarks: sentiment analysis, named entity recognition, and question answering, demonstrating its superior performance compared to AraBERT v1 and mBERT baselines. Our approach significantly reduces computational overhead, enhancing accessibility and applicability for research communities and industry practitioners working with Arabic language processing.

- We empirically evaluate ModernAraBERT across three essential Arabic NLP benchmarks: sentiment analysis, named entity recognition, and question answering against two strong baselines: AraBERT v1, the most widely recognized and cited Arabic BERT model with performance comparable to later variants (Antoun et al., 2020; Farha and Magdy, 2021), and mBERT, a widely adopted multilingual model covering Arabic among 104 languages (Alammary, 2022). Our results show that ModernAraBERT consistently outperforms both models while reducing computational overhead.

2. Methodology

Our methodology builds on that large-scale English-pretrained models encode transferable cross-lingual knowledge that can be effectively adapted to morphologically rich languages such as Arabic. Instead of training a new Arabic-specific model from scratch, which is both computationally costly and resource intensive, we employ continued pretraining on curated Arabic corpora. This strategy preserves the syntactic and semantic priors acquired during large-scale English pretraining, while adapting the model to capture Arabic-specific morphology and orthographic variations. Prior studies have demonstrated that domain and language-adaptive pretraining often yields superior performance compared to monolingual training under resource constraints (Gururangan et al., 2020; Pfeiffer et al., 2021). Moreover, multilingual models such as mBERT have shown that English initialization can rival or even surpass dedicated Arabic models (e.g., AraBERT) in certain tasks (Alammary, 2022). Our approach thus provides a scalable and resource-efficient alternative to monolingual pretraining, while ensuring comparability with established baselines.

2.1. Pretraining Corpora

We compiled a large-scale Arabic corpus from four publicly available sources: OSIAN (Zeroual et al., 2019), the Arabic Billion Words dataset (El-Khair, 2016), the Arabic Wikipedia dump[1], and the OSCAR Arabic dataset (Abadji et al., 2022). These corpora jointly cover Modern Standard Arabic (MSA) and a variety of dialectal forms.

Preprocessing 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.

2.2. Tokenization

We extended the original ModernBERT tokenizer, which was trained on English corpora, by adding 80,000 Arabic-specific tokens. The extended vocabulary included segmented roots, inflected forms, and common affixes, ensuring that Arabic morphology was more faithfully captured. Frequent morphological constructions were explicitly added as standalone tokens to improve segmentation consistency and reduce fragmentation.

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.

[1] https://dumps.wikimedia.org/arwiki/

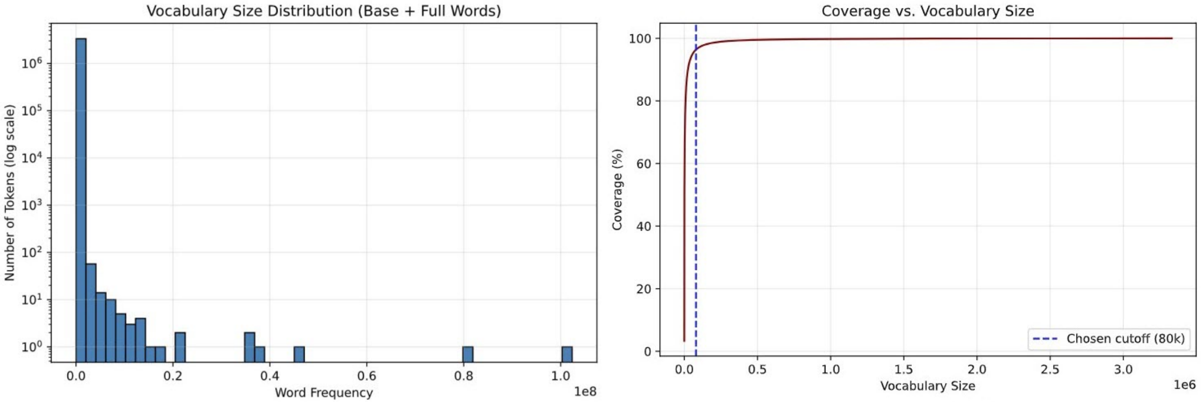

Figure 1: Vocabulary size analysis. Left: Token frequency histogram (log scale). Right: Coverage vs. vocabulary size, with the chosen cutoff at 80K tokens.

2.3. Model Training

Our model is based on the publicly available ModernBERT-base[2] architecture with 22 transformer layers (Figure 2). To accommodate the extended vocabulary, we resized the embedding layer accordingly.

Pretraining was conducted with the Masked Language Modeling (MLM) objective, updating all model parameters. Training proceeded for three epochs: the first two epochs used sequences of length 128 for efficiency, while the final epoch employed sequences of 512 to model longer contexts. The context length was restricted to 512 tokens both to ensure fair comparability with AraBERT (which also uses 512) and to fit within the 40 GB GPU memory available. Longer contexts (e.g., 8,192 tokens) were not computationally feasible under our hardware constraints. Optimization used AdamW with cosine learning rate decay and gradient clipping. Training progress was monitored via loss and perplexity on a held-out validation set.

The pretrained ModernAraBERT will be released on HuggingFace[3], together with training and evaluation scripts in our repository[4].

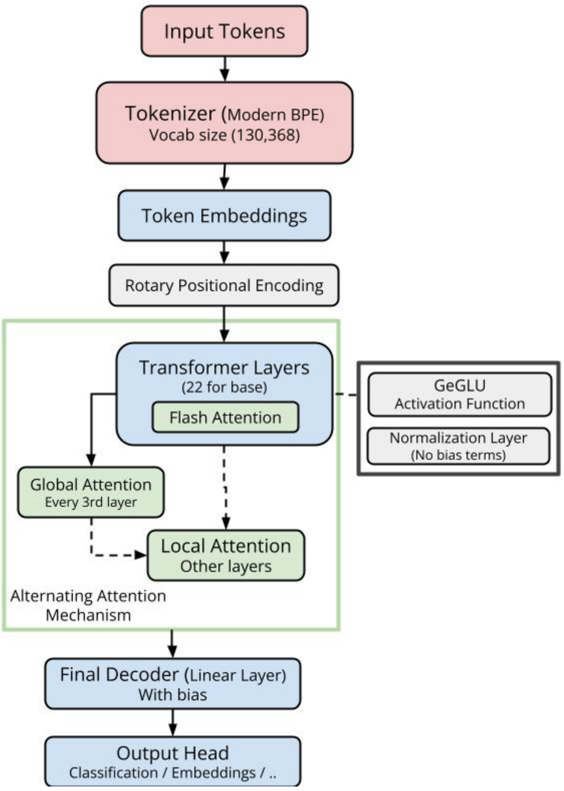

Figure 2: ModernBERT architecture with extended tokenizer vocabulary size and embedding layer.

3. Experimental Setup

To assess the effectiveness of our proposed adaptation strategy, we evaluate ModernAraBERT on three representative Arabic NLP tasks: sentiment analysis (SA), named entity recognition (NER), and extractive question answering (QA). These tasks were chosen as they collectively cover sentence-level, sequence-labeling, and span-extraction settings, providing a comprehensive evaluation of model capabilities. We compare against two strong baselines: AraBERT v1, the most established Arabic-specific BERT variant, and mBERT, a widely adopted multilingual model.

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.

[2] ModernBERT-Base
[3] URL to be revealed after double-blind review
[4] Repository link

3.1. Computational Environment

All pretraining and fine-tuning experiments were conducted on a high-performance computing node equipped with 12 CPU cores, 32 GB RAM, and a 40 GB NVIDIA A100 GPU. These specifications constrained the choice of sequence length (512 tokens) to ensure comparability with `AraBERT` while remaining feasible within GPU memory.

3.2. 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. An exception was NER, which converged reliably within 5 epochs.

3.3. 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.
- **Arabic Jordanian General Tweets (AJGT)** Corpus[5], containing 1,800 tweets labeled as positive or negative.
- **Large-Scale Arabic Book Reviews (LABR)** (Aly and Atiya, 2013), using the unbalanced binary version for consistency with prior work.

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.

3.4. 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`.

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, - continuation 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, following established NER practice.

3.5. Question Answering

For extractive QA, we combined Arabic-SQuAD (Mozannar and Others, 2019) 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 was designed to provide a rigorous and fair evaluation of `ModernAraBERT`. We assessed the model across three complementary task types—sentence-level classification (SA), sequence labeling (NER), and span extraction (QA)—using widely adopted benchmark datasets. Comparisons against both Arabic-specific (`AraBERT v1`) and multilingual (`mBERT`) baselines ensure that our evaluation is representative of the state of the art. By freezing the encoder and fine-tuning only lightweight task-specific heads, we isolate the contribution of continued pretraining on Arabic corpora while controlling for overfitting.

[5] AJGT Dataset

The following section presents the results of these experiments, highlighting both performance gains and computational trade-offs relative to the baselines.

4. Results and Discussion

This section reports the empirical results of ModernAraBERT on the three mentioned downstream Arabic NLP tasks. Our discussion emphasizes both absolute performance gains and relative improvements over baselines, highlighting the effectiveness of continued pretraining as well as any trade-offs observed.

4.1. Sentiment Analysis

Table 4.1 summarizes performance on the three sentiment datasets. Across all benchmarks, ModernAraBERT achieves substantial improvements over both AraBERT v1 and mBERT, despite only fine-tuning the prediction head. This demonstrates that continued pretraining on Arabic corpora effectively enhances the model's sentence-level representations, yielding strong transfer to downstream classification tasks.

Performance gains are particularly notable on the HARD dataset (+16.7 % over AraBERT, +17.7 % over mBERT), which contains both MSA and dialectal Arabic, indicating that the model captures cross-variant sentiment signals more effectively. On AJGT and LABR, improvements of 12.5 % and 11 % respectively confirm robustness across both small-scale (tweets) and large-scale (book reviews) corpora. These consistent gains suggest that ModernAraBERT generalizes well across domains and genre variations.

Dataset	AraBERT	mBERT	ModernAraBERT
AJGT	58.0	61.5	**70.5**
HARD	72.7	71.7	**89.4**
LABR	45.5	45.5	**56.5**

Table 1: Macro-F1 (%) on sentiment datasets.

4.2. Named Entity Recognition (NER)

Table 4.2 reports the NER results on ANERCorp. ModernAraBERT achieved a micro F1-score of 82.1 %, surpassing AraBERT v1 (78.9%) while remaining below mBERT (90.7%). These results highlight two important observations. First, continued pretraining on Arabic corpora improves token-level representation quality relative to AraBERT, confirming the benefits of our adaptation strategy for sequence labeling. Second, the superior performance of mBERT indicates that multilingual training at scale may provide broader cross-lingual generalization for this task, potentially due to larger training diversity and multilingual alignment.

Model	Micro F1
AraBERT (Antoun et al., 2020)	78.9
mBERT	**90.7**
ModernAraBERT	82.1

Table 2: Named Entity Recognition results (Micro F1-score, %) on ANERCorp.

4.3. Question Answering (QA)

Results for the QA task are presented in Table 4.3. ModernAraBERT consistently outperformed both baselines across all evaluation metrics. Compared to AraBERT, our model achieved a 41.3% relative gain in Exact Match (18.73 vs. 13.26), a 15.6% relative gain in F1-score (47.18 vs. 40.82), and a 7.3% relative gain in Sentence Match (76.66 vs. 71.47). These improvements indicate that ModernAraBERT not only produces more exact matches but also captures semantic information more effectively at both token and sentence levels.

Interestingly, while mBERT performed competitively (15.27 EM, 46.12 F1-Score), it lagged behind in Sentence Match (63.11), suggesting weaker ability to capture longer-span semantic coherence. By contrast, ModernAraBERT demonstrated stronger semantic alignment, which we attribute to the continued pretraining on Arabic corpora and task-specific fine-tuning.

Model	EM	F1	SM
AraBERT (Antoun et al., 2020)	13.26	40.82	71.47
mBERT	15.27	46.12	63.11
ModernAraBERT	**18.73**	**47.18**	**76.66**

Table 3: Extractive QA results (%) on ARCD Test Split.

EM: Exact Match, F1: token-level F1, SM: Sentence Match.

4.4. Overall Analysis

Across all evaluated tasks, adapting an English pretrained model to Arabic through our two phase strategy continued pretraining followed by task-specific fine-tuning proved highly effective. ModernAraBERT consistently outperformed the Arabic-specific AraBERT v1 baseline, with particularly strong gains in sentiment analysis and question answering. These improvements confirm that resource-efficient language adaptation can rival or surpass monolingual models trained from scratch.

At the same time, results underscore important trade-offs. While ModernAraBERT achieved higher accuracy, AraBERT maintained superior inference throughput achieving a higher throughput of 925.4 samples/sec vs. 495.8 for ModernAraBERT and lower GPU memory usage, especially evident in NER experiments. This suggests that model selection should be guided by application requirements: latency-critical or resource-constrained scenarios may favor AraBERT, whereas accuracy-oriented deployments benefit more from ModernAraBERT. The competitive performance of mBERT in NER further highlights the potential of large multilingual pretraining for token-level tasks, though it lagged behind in sentence-level semantic alignment (SM) for QA.

4.5. Hardware Resource Usage

Table 4.5 summarizes peak memory consumption across benchmarks. A consistent pattern emerges: AraBERT is the most memory-efficient model, while ModernAraBERT incurs higher VRAM usage, particularly for QA (3.22 GB vs. 2.07 GB for AraBERT). RAM usage remained broadly similar across models, with only minor fluctuations.

These findings confirm that ModernAraBERT offers accuracy improvements at the cost of increased GPU memory demand, especially in span-extraction tasks. From a practical perspective, this trade-off is acceptable for research and enterprise environments with sufficient GPU capacity, but may limit deployment on edge devices or latency-sensitive pipelines. By contrast, AraBERT remains attractive for lightweight applications, while mBERT provides a balanced middle ground for scenarios requiring cross-lingual portability.

Benchmark	Model	Peak RAM	Peak VRAM
NER	AraBERT	1.53	0.52
	mBERT	1.60	0.68
	ModernAraBERT	1.49	0.83
QA	AraBERT	1.42	2.07
	mBERT	1.46	2.84
	ModernAraBERT	1.39	3.22
SA	AraBERT	1.65	0.52
	mBERT	1.66	0.68
	ModernAraBERT	1.36	0.82

Table 4: Hardware resource usage across models and benchmarks (GB).

Although our experiments focus on Arabic, the proposed methodology is inherently language-agnostic. By leveraging English-pretrained backbones such as ModernBERT and applying language-adaptive pretraining on target corpora, similar strategies can be employed for other morphologically rich or low-resource languages. This reduces the need to train monolingual models from scratch, a process that is often prohibitively expensive in both data and compute. The results suggest that languages with limited dedicated resources may benefit disproportionately from this approach. For instance, languages such as Amharic, Urdu, or Kazakh—which face challenges of sparse annotated data and high morphological complexity could be supported through continued pretraining on monolingual corpora while reusing the cross-lingual knowledge encoded in large English models (Wiemerslage et al., 2022).

From a community perspective, this paradigm promotes inclusivity by lowering the barrier to building competitive NLP models for underrepresented languages. It complements multilingual models like mBERT by offering a targeted, resource-efficient alternative that can achieve stronger task-specific performance without requiring massive multilingual pretraining. Thus, the broader impact of our work lies in presenting a scalable framework for extending state-of-the-art NLP to languages that remain marginalized in the current landscape of large language models.

5. Conclusion

This paper introduced ModernAraBERT, a resource-efficient adaptation of the English-pretrained ModernBERT model to Arabic. Our approach leverages continued pretraining on curated Arabic corpora followed by lightweight task-specific fine-tuning. Experimental results across three representative tasks—sentiment analysis, named entity recognition, and question answering—demonstrated that ModernAraBERT consistently outperforms the widely used AraBERT v1 baseline and achieves competitive results against the multilingual mBERT. The results highlight important trade-offs. While ModernAraBERT improves accuracy, especially in sentence- and span-level tasks, it incurs higher GPU memory usage and slower inference compared to AraBERT. This suggests that model selection should be deployment-specific: lightweight models for latency-critical scenarios, and adapted models like ModernAraBERT for accuracy-oriented applications.

Beyond Arabic, our approach has a broader applicability of language-adaptive pretraining as a scalable alternative to monolingual model development. The methodology can be extended to other morphologically rich or underrepresented languages, offering a pathway to reduce duplication of effort while maintaining strong downstream performance.

In future work, we plan to (i) explore mixed adaptation strategies combining language and domain adaptive pretraining, (ii) investigate parameter-efficient tuning techniques to further reduce memory overhead. We believe these directions will strengthen the case for resource-efficient cross-lingual adaptation as a practical paradigm for building inclusive NLP systems.

6. 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.

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.

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*.

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.

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.

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.

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.

7. Language Resource References