Investigating Zero-shot Cross-lingual Language Understanding for Arabic

Zaid Alyafeai*
Department of Computer Science
King Fahd University of Petroleum
and Minerals
Dhahran, Saudi Arabia
g201080740@kfupm.edu.sa

Moataz Ahmed
Department of Computer Science
King Fahd University of Petroleum
and Minerals
Dhahran, Saudi Arabia
moataz@kfupm.edu.sa

Abstract

Numerous languages exhibit shared characteristics, especially in morphological features. For instance, Arabic and Russian both belong to the fusional language category. The question arises: Do such common traits influence language comprehension across diverse linguistic backgrounds? This study explores the possibility of transferring comprehension skills across languages to Arabic in a zero-shot scenario. Specifically, we demonstrate that training language models on other languages can enhance comprehension of Arabic, as evidenced by our evaluations in three key tasks: natural language inference, question answering, and named entity recognition. Our experiments reveal that certain morphologically rich languages (MRLs), such as Russian, display similarities to Arabic when assessed in a zero-shot context, particularly in tasks like question answering and natural language inference. However, this similarity is less pronounced in tasks like named entity recognition.

1 Introduction

Language models have been mainly utilized by training on a large corpus using a monolingual approach i.e. on a single language like BERT (Devlin et al., 2018), GPT (Radford et al., 2018), and Roberta (Liu et al., 2019). On the other hand, there were some attempts to train such language models on multiple languages like multilingual BERT (mBERT) (Devlin et al., 2018) by combining text from different languages. A tokenizer such as WordPiece, is trained on the joined text from different languages to be able to recognize the scripts from such languages. This makes the vocabulary size of such models huge. For example, mBERT has a shared vocabulary size of 110K across the 104 languages that were used for training compared to the 30K vocabulary size that was used to train the monolingual BERT. With such a huge vocabulary and the number of languages, it is not clear how knowledge or language understanding is shared across such languages. More importantly, it is important to investigate how such knowledge is shared among similar languages, especially in terms of morphological features. We mainly focus on languages that exhibit rich morphology like Arabic. In this study, our primary objective is to explore the integration of knowledge into Arabic by fine-tuning mBERT across various tasks, including question answering, natural language inference, and named entity recognition in multiple languages, followed by a zero-shot evaluation specifically on Arabic.

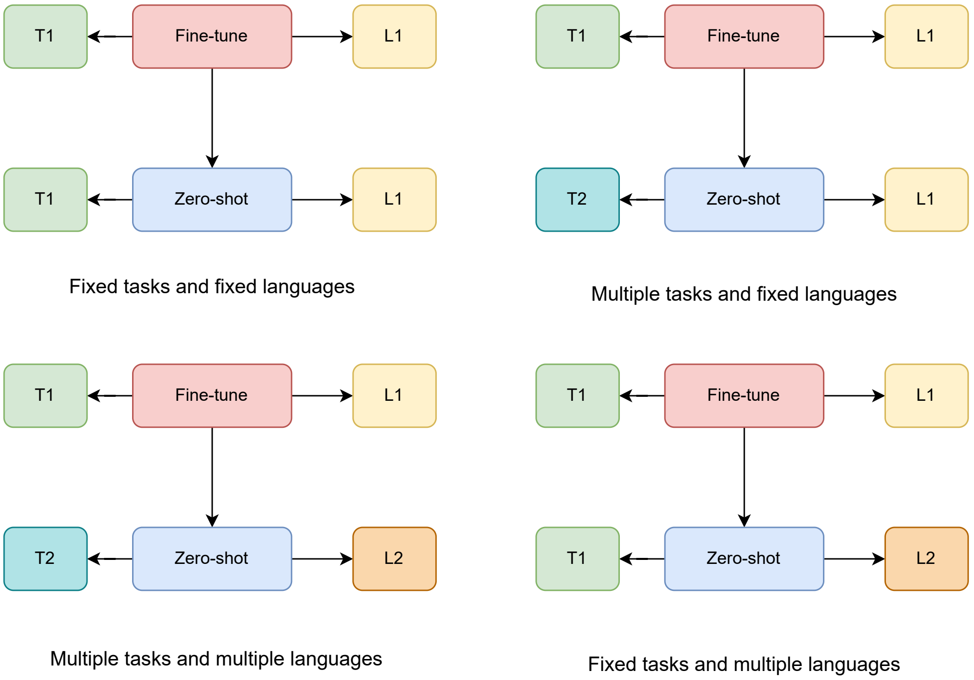
This paper is organized as follows. In Section 2, we discuss the related studies to our work. In Section 3, we focus on discussing the scope of our work. In Sections 4 and 5, we discuss morphology in general and how it's an intrinsic property of Arabic. In Section 6, we investigate mBERT and why it's an important model to evaluate such properties on. In Section 7, we detail the datasets and tasks used for evaluating our study. Finally, in Section 8, we detail our experiments and discuss our results.

2 Related Work

Multilinguality focuses on training language models with shared vocabulary for multiple languages. Over the past few years, many models have adopted this strategy like multi-lingual BERT (mBERT) for 104 languages (Devlin et al., 2018), XLM-R for 100 languages (Conneau et al., 2019), and mT5 for 101 languages (Xue et al., 2020). The advantage of using such models is the simplicity of creating a shared vocabulary using a uniform linear mapping between the different multilingual embeddings. Interestingly, mBERT demonstrates proficiency in zero-shot cross-lingual model transfer, as observed in prior research (Pires et al., 2019). This capability aids in comprehending a given language in a universal context. However such models are required

*corresponding author

Proceedings of the The First Arabic Natural Language Processing Conference (ArabicNLP 2023), pages 324–334
December 7, 2023 ©2023 Association for Computational Linguistics



Fixed tasks and fixed languages

Multiple tasks and fixed languages

Multiple tasks and multiple languages

Fixed tasks and multiple languages

Figure 1: Different approaches for zero-shot evaluation with variations in tasks and languages. In each figure, we show the tasks and languages used for fine-tuning and the tasks and languages used for zero-shot evaluation. In this study, we focus on the approach of fixed tasks and multiple languages.

to be trained on a multilingual objective in order to generalize more for distant languages with different typography (Lauscher et al., 2020). Not to mention how to transfer knowledge to low-resource languages. Lately, there has been growing interest in utilizing more sophisticated architectures to enhance knowledge optimization in low-resource scenarios. One of the most interesting approaches is using adapter modules to avoid catastrophic forgetting [1] when training multilingual models on different languages. (Pfeiffer et al., 2020) focused on creating a framework for multi-task adapter-based cross-lingual transfer. (Hu et al., 2020) created a benchmark of the evaluation of cross-lingual transfer for 40 languages XTREME.

In the literature, there were some limited efforts to apply zero-shot understanding for Arabic. (Khalifa et al., 2021) used Self-Training of pre-trained language models for zero- and few-shot multi-dialectal Arabic sequence labeling by first fine-tuning on modern standard Arabic (MSA). Some studies focused on applying these techniques for Arabic like GigaBERT which can achieve bilingual zero-shot understanding from English to Arabic (Lan et al., 2020). They apply these methods for information extraction tasks (IE) like part of speech tagging (POS), named entity recognition (NER), relation extraction (RE), and argument role labeling (ARL) tasks. (Abboud et al., 2022) studied cross-lingual understanding from English and French to Arabic. They show strong performance in a zero-shot setting despite the differences between the source and target languages in terms of morphology and grammar. There were many efforts also to benchmark ChatGPT models in a zero-shot fashion on multiple tasks for Arabic without fine-tuning (Kadaoui et al., 2023), (Alyafeai et al., 2023), (Khondaker et al., 2023), and (Abdelali et al., 2023). Models like ChatGPT which was trained on a large mixture of scripts for hundreds of languages were able to attain strong performance on multiple tasks in a zero-shot fashion.

3 Zero-shot Evaluation

The default approach of evaluating a language model on a given task is by training the model on that dataset and then evaluating on the unseen split of the dataset. We assume that both training and test splits belong to the same language/task. However, we can also argue that we can train the language model on a given task say T1 then evaluate on another task, say, T2. Similarly, we can

[1] Happens when the weights are fine-tuned on new datasets. The models usually forget the previous knowledge.

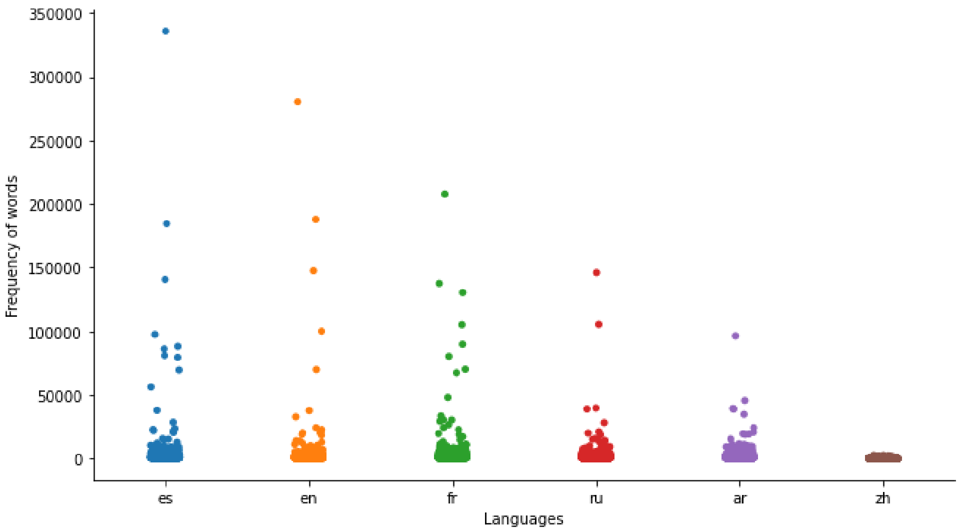


Figure 2: Comparing languages in terms of the frequency of the top 1000 words in the corpus.

train a language model on a given language L1 and evaluate on another language L2 (see Figure 1). However, in order to do that, the language model has to be able to predict or generate tokens in that language. Hence, multilingual models have been utilized to evaluate cross-lingual understanding. Such language models like mBERT (discussed in Section 6) are trained on a corpus that contains multiple languages. As a result, we can hypothesize that such language models have attained some kind of relationship across different languages either in terms of script or topology. As an example, Arabic and Persian have the same script and share many common words. More interestingly, using zero-shot evaluation we can test whether a given language is closer to other languages in terms of more complex features like morphology. For example, both Arabic and Russian are rich in terms of morphology and they are both inflectional languages. In this paper, we mainly focus on zero-shot evaluation on the same set of tasks but in different languages. To summarize, given a language L1 we train it on a given task T1 and zero-shot evaluate on L2 on the same task T1. In this paper, L2 is Arabic, and L1 could be any language.

4 Morphology (Arabic and Beyond)

Tackling morphology is a very important step toward improving language modeling for languages like Arabic. In the literature, (Antoun et al., 2020) showed slightly better results by pre-splitting words using the Farasa segmentation tool (Abdelali et al., 2016) on multiple tasks. The morphological segmentation results in better performance in text classification tasks while worse results in question answering and named entity recognition tasks. Similarly, (Oudah et al., 2019) showed that we can get some improvement when we employ different morphological analyzers on top of neural and statistical models for machine translation. (AlKhamissi et al., 2020) showed that by utilizing a combination of character- and word-level representations they achieved better results on the diacritization task. (Alkaoud and Syed, 2020) modified the tokenization algorithm for multilingual BERT to achieve better results than monolingual BERT on two different datasets.

Morphology also exists in other languages but with different levels of complexity depending on the language as shown in Table 1. (Hofmann et al., 2020) modified BERT for generating derivationally more complicated English words using masked language modeling objective. The conditioned language model on the word can predict the prefixes and suffixes of that masked word. (Sennrich et al., 2015) compared Byte-Pair Encoding (BPE) and unigram language model [2] (Kudo, 2018) for the translation between low-resource morphologically rich languages (Turkish and Swahili) into English (Richburg et al., 2020). They showed an improvement in using unigram language models. Similarly, Bostrom and Durrett showed an improvement in using unigram language models for tokenization over BPE for morphologically rich languages (Bostrom and Durrett, 2020). The generated tokens align bet-

[2] Uses a language model to predict the morphemes.

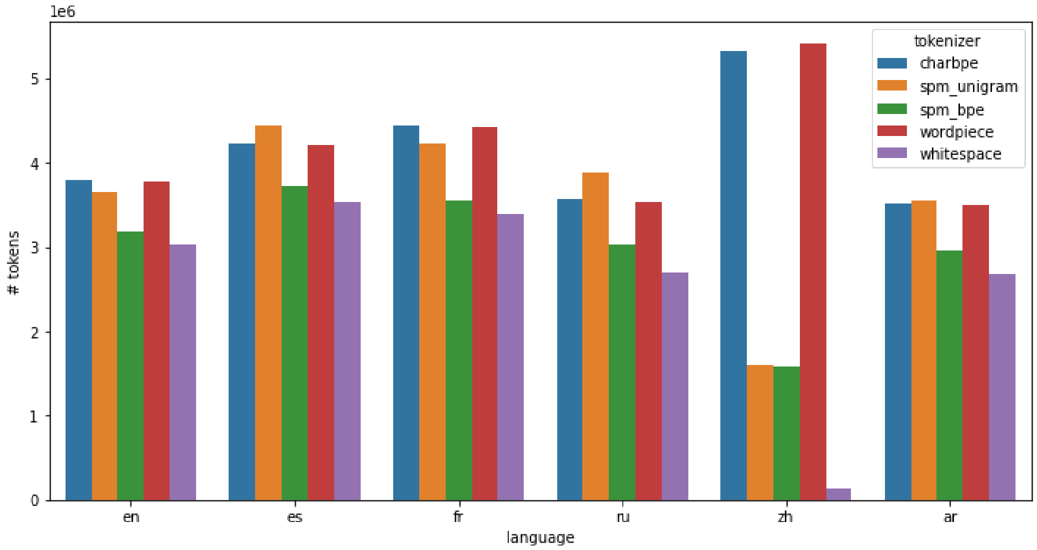


Figure 3: Comparing the number of tokens generated by different tokenizers in different languages. The language codes represent en:English, es:Spanish, fr:French, ru:Russian, zh:Chinese, and ar:Arabic.

ter with morphology for the unigram language models compared to BPE. (King et al., 2020) analyzed sequence-to-sequence models used for translation on Russian languages for morphological inflections. They showed that conditioning such models with word embeddings for lexical semantics can improve the results for translation. Klein and Tsarfaty tested the morphological attributes of the WordPiece algorithm for modern Hebrew and showed that the linear split of the tokenization algorithms might be sub-optimal (Klein and Tsarfaty, 2020). They report that by using more language-dependent tokenization approaches we can improve language understanding for morphologically rich languages. (Gerz et al., 2018) suggest an approach for tackling morphology in language modeling via a combination of characters- with word-level predictions for 50 languages.

Table 1: Complexity of morphology in different languages (Clark et al., 2021).

Dataset	**Language**
Impoverished Morphology	English
Agglutinative Morphology	Turkish
Non-concatenative Morphology	Arabic
Reduplication	Kiswahili
Compounding	German
Consonant Mutation	Welsh
Vowel harmony	Finnish

5 Arabic's Vocabulary Sparsity

Vocabulary sparsity is the problem of having a very large vocabulary set with many different inflections. This presents a challenge for language modeling because typically, when designing a language model, we aim to acquire vocabulary embeddings. In this section, we analyze Arabic morphology through vocabulary counting of parallel datasets.

To conduct this experiment, we utilized the parallel dataset sourced from the United Nations (Rafalovitch et al., 2009). This dataset comprises United Nations General Assembly Resolutions in six distinct languages: Arabic, Chinese, English, French, Russian, and Spanish. In Table 2, we conducted a comparison between the number of tokens and the vocabulary size across these six languages. The vocabulary size denotes the count of unique tokens within the dataset for each language. From the table, we can discern that languages with rich morphology, such as Arabic and Russian, rank first and second, respectively, in terms of the number of unique tokens, even though they have a relatively smaller number of tokens compared to Spanish and French. This phenomenon can be attributed to the extensive inflections present in morphologically rich languages (MRLs). One potential approach to mitigating this issue involves applying morphological segmentation techniques, such as using a tool like FARASA (Abdelali et al., 2016). However, as indicated in the last row of the table, this segmentation introduces a trade-off: while it substantially reduces the vocabulary size, it simultaneously increases the number of tokens. This trade-off poses challenges during the training of language models, making it more difficult for the model to comprehend longer and more sophisticated sequences.

Table 2: Number of tokens and vocabulary size in other languages compared to Arabic and segmented Arabic. In this context, a token is equivalent to a word.

Dataset	**# of Tokens**	**Vocab Size**
English (en)	2,963,479	70,330
Spanish (es)	3,465,588	79,005
French (fr)	3,328,567	63,907
Russian (ru)	2,628,322	96,292
Chinese (zh)	60,107	51,884
Arabic (ar)	2,601,126	**103,339**
Arabic Segmented	**7,844,083**	15,250

In Figure 2, we compare the most frequent 1000 words in each language in the United Nations corpus. As we can see, even though the number of tokens is very high for Arabic, the frequency is low. Note that the Chinese language achieves the lowest frequency. This is due to the fact that Chinese doesn't support white space tokenization which causes its vocabulary set to be very large, hence low frequency for repetition. Interestingly, the graph shows a linear change in the frequency for the languages starting with Spain with the highest up to Chinese with the lowest.

In Figure 3 we compare five different tokenizer approaches applied to the six different languages. As we can observe MRLs like Russian and Arabic generate a relatively small number of tokens. More importantly, the distribution of the frequency of the number of tokens across the different tokenizers seems very similar for such languages. The number of tokens generated across different languages seems to depend on the language. SentencePiece with unigram seems to generate the smallest number of tokens across different languages. However, there is no distinction between which tokenizer creates the maximum number of tokens. Note that as expected, White-space tokenization generates the lowest number of tokens for all the languages.

6 Multilingual BERT

BERT (Devlin et al., 2018) is a transformer-based model that was trained on a large corpus using unsupervised learning. The main architecture of the model is based on the transformer model from (Vaswani et al., 2017) which leverages attention to design an efficient encoder-decoder model that beats the existing machine translation models at that time. BERT is trained using a concatenation of two objectives:

- Masked language modeling: The main task is to randomly mask 15% of the tokens during training and the model has to predict these masked tokens at the end.
- Next sentence prediction: the BERT model separates sentences by a special operator `[sep]`. Then with certain probability can attach unrelated sentences together from the corpus. The objective is then focused on predicting if the second sentence is possible given the first sentence.

Using the concatenation of such objectives, the model can learn efficient text representation and can be fine-tuned on multiple tasks by attaching some uninitialized weights at the end of the model.

The multilingual version of BERT trains the model on a multilingual corpus that contains 104 languages. The initial training corpus was extracted from Wikipedia with the top 100 languages then Thai and Mongolian were later added. This results in some languages which are under-represented. To mediate that, the authors used sampling to reduce the probability of training on high-resource languages like English and increase the probability of sampling from low-resource languages like Icelandic. The base model used a shared vocabulary size of 110K which was extracted using the WordPiece tokenization algorithm. Similar to sampling, the word counts are multiplied with the sampling factor as in the training to reduce the effect of variation in the existence of different languages.

7 Tasks

In this paper, we mainly focus on three tasks which are natural language inference, question answering, and named entity recognition. We use the datasets that have parallel sentences i.e. the same sentences are used for training the language models but in different languages. The reason for that choice is

Table 3: The number of samples in each dataset across the different languages.

Dataset	Number of languages	Train	Valid	Test
XNLI	15	50,000	2,490	5,010
XQuAD	11	952	119	119
MASSIVE	52	11,514	2,033	2,974

1) we want the same amount of data in terms of height (number of samples) and roughly the same depth (number of tokens per sentence) and 2) we don't want to infuse any types of bias due to using different sentences for different languages i.e we want to force the model to use the knowledge in a similar setting to machine translation. We chose three datasets which are XNLI for natural language inference, XQuAD for question answering, and MASSIVE for named entity recognition. Here is a detailed explanation of each dataset.

1. **XNLI (Conneau et al., 2018)** is a natural language inference dataset that was extracted from MNLI (Williams et al., 2018) which is a multi-genre dataset that contains more than 400K pairs of sentences. XNLI contains 392,702 training, 2,490, and 5,010 samples machine-translated into 14 different languages. The main purpose of natural language inference is to predict if the `hypothesis` follows from a `premise` i.e. entailment or contradiction or neither. Given the hypothesis and premise, the task is to predict one of the three labels so, this can be considered a more generalized classification task. Due to the size of the dataset and the limited compute, we only extract 50K samples from the dataset for fine-tuning.

2. **XQuAD (Artetxe et al., 2019)** is a cross-lingual question answering dataset. It was extracted from the SQuAD v1.1 (Rajpurkar et al., 2016) benchmark by collecting 240 paragraphs and 1,190 question-answer pairs from the development set. Then it was translated into ten languages which are Arabic, Chinese, Hindi, German, Greek, Russian, Spanish, Thai, Turkish, and Vietnamese. Hence, the dataset contains parallel samples from 11 languages. We split the dataset into 952 training, 119 validation, and 119 testing splits. Each sample of the dataset contains, `question`, `context`, and `answer_span`.

3. **MASSIVE (FitzGerald et al., 2022)** contains 1 million sentences that span across 52 languages. Each language contains 19,521 samples that were split into 11,514 training, 2,033, and 2,974 testing. The dataset is annotated for natural language understanding tasks. We mainly use the dataset for named entity recognition tasks which contains 111 tags spanning different entities like food, person, coffee, time, etc.

In Table 3, we summarize the number of samples in each dataset for each split and the number of languages for each dataset. Note that, although there are many datasets that test cross-lingual understanding, we only consider datasets that have parallel samples in each language.

8 Results and Discussions

We fine-tune mBERT[3] using the Trainer class[4] which provides a simple way for training and fine-tuning transformer-based models. All the experiments were run using Google Colab[5] with the default virtual machine that contains a T4 NVIDIA card with 16 GB memory size. We use the PyTorch examples from the Transformer repository on GitHub[6] with the following parameters for each task:

- **Natural Language Inference** We fine-tuned the model for 2 epochs with batch size 32 and learning rate 5e-5. We use a max sequence length of size 128 for the premise.

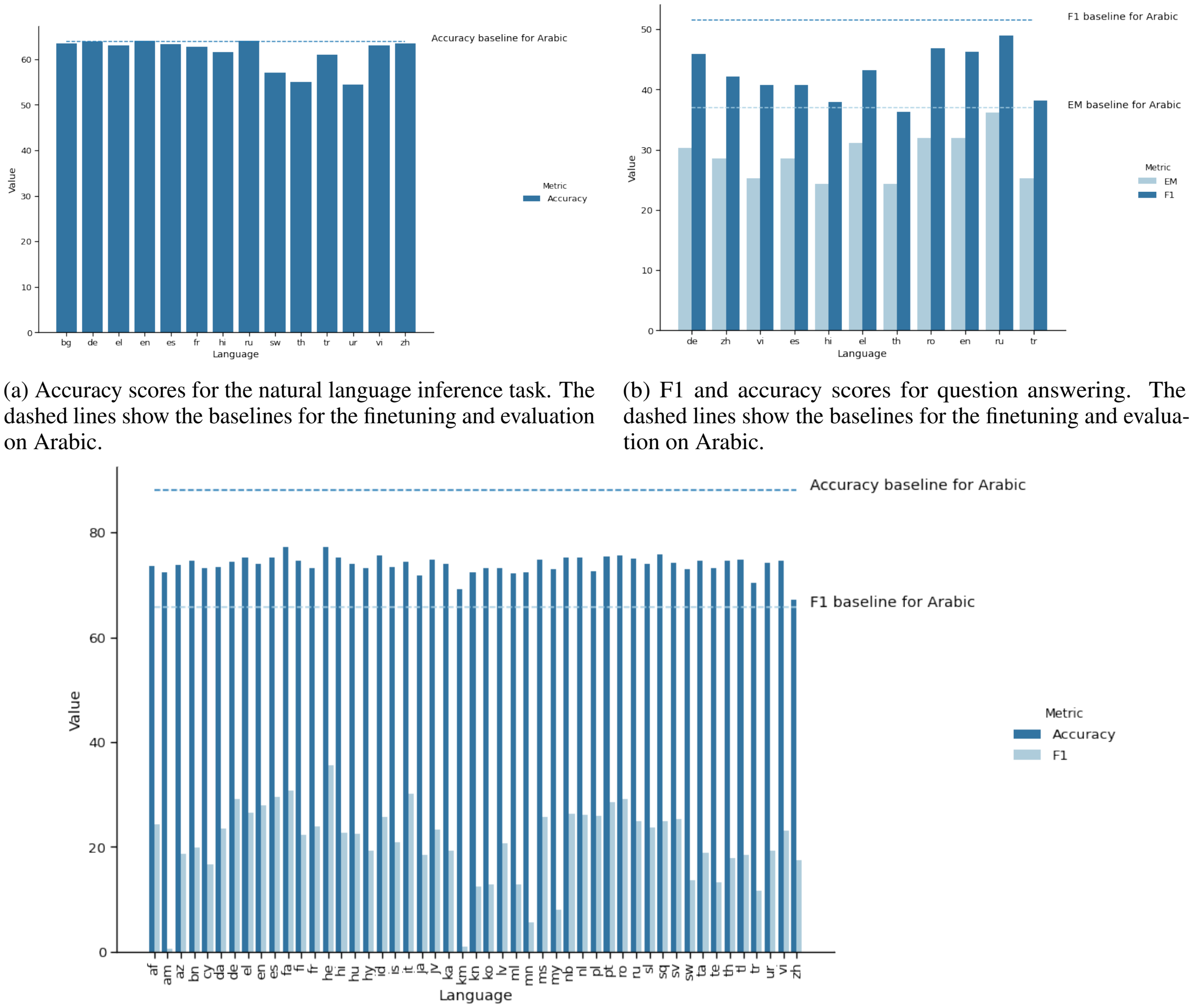
- **Question Answering** we fine-tune the models for two epochs with a batch size of 12. We

[3] `https://huggingface.co/bert-base-multilingual-cased`

[4] `https://huggingface.co/docs/transformers/main_classes/trainer`

[5] `https://colab.research.google.com`

[6] `https://github.com/huggingface/transformers/tree/main/examples/pytorch`



(a) Accuracy scores for the natural language inference task. The dashed lines show the baselines for the finetuning and evaluation on Arabic.

(b) F1 and accuracy scores for question answering. The dashed lines show the baselines for the finetuning and evaluation on Arabic.

(c) Accuracy and F1 scores for the named entity recognition task. The dashed lines show the baselines for the finetuning and evaluation on Arabic.

Figure 4: Results for question answering, natural language inference, and named entity recognition tasks.

also use a learning rate of 3e-5. For the context size, we use 384 max-size with a stride of size 128.

- **Named Entity Recognition** We fine-tune the model for two epochs with batch size 12 and learning rate 3e-5.

In Figure 4a, we show the results for the zero-shot evaluation on the natural language inference dataset XNLI. English and Russian achieve very similar results which approach the baseline for Arabic. The German language also achieves somewhat close results to the baselines. Urdu and Thai both achieve the worst results for natural language inference which are close to 50 % accuracy which is much lower than Chinese. Although Chinese and Thai are similar in pronunciation and other grammatical features, they belong to different language families. Note that this is just based on the 50,000 samples used for training. Increasing the training samples might result in different results, especially for the languages that are close in results. Furthermore, the results approach the baseline for fine-tuning and evaluating on Arabic. This might be the effect of using a machine-translated dataset for evaluation.

In Figure 4b, we show the results for the zero-shot evaluation on the question-answering dataset xQuAD. The dashed lines show the results of the baselines after training and evaluating on Arabic for exact match and F1 scores. We notice that the Russian language achieves the best scores for both the Exact match and F1 scores. These results correlate with the initial experiments in Section 5. Followed by the Romanian language which seems quite close to Russian in terms of structure. The

Thai language achieves the worst scores across all metrics which might be related to the structure of the language which is quite close to Chinese which does not use white-space tokenization.

In Figure 4c, we present the outcomes of the zero-shot evaluation conducted on the named entity recognition task. Overall, it is discernible that the outcomes, particularly the F1 scores, exhibit notable decrements when compared to the baseline, specifically within the Arabic language context. This discrepancy could be attributed to the dataset's substantial entity count, exceeding 100. Consequently, this abundance of entities introduces a degree of stochasticity into the cross-lingual comprehension process, complicating the derivation of definitive insights from the results.

9 Conclusion

In this research, we delved into the realm of cross-lingual zero-shot transfer, where we explored the application of knowledge from various languages to Arabic through the evaluation of multiple tasks. These tasks encompassed named entity recognition, natural language inference, and question answering. Initially, we employed an unsupervised approach to scrutinize the distinctions in morphology between Arabic and other languages. Subsequently, by employing supervised methods, we revealed certain connections between Arabic and other languages concerning their structure and writing systems. Our investigation demonstrated that superior results can be achieved by training models on languages other than Arabic and subsequently assessing their performance on Arabic, as opposed to direct training on Arabic. This phenomenon may be attributed to several factors, including the simplicity of the language, the resemblance of these languages to Arabic, and the distribution of the initial training data used for unsupervised learning. As a future direction, it could be interesting to look into more diverse tasks and more advanced transformer-based architectures.

Limitations

We highlight some limitations of our study. We summarize them as the following:

- **Data Quality** The quality and quantity of data available in the target languages, especially Arabic, can significantly impact the effectiveness of cross-lingual transfer. Limited or low-quality data can lead to sub-optimal results. For example, the XNLI dataset is machine-translated from English to Arabic which could result in some issues.

- **Language Distance** The success of cross-lingual transfer often depends on the linguistic distance between the source and target languages. If the source languages are distant from Arabic in terms of syntax, grammar, and vocabulary, the transfer may not be as effective.

- **Task Relevance** The paper discusses evaluating multiple tasks, including named entity recognition, natural language inference, and question answering. It's important to consider whether these tasks are representative of the general language understanding domain and whether the findings can be generalized to other tasks.

- **Bias and Fairness** The study doesn't explicitly mention considerations related to bias and fairness. Cross-lingual models can inherit biases from their training data, which can be problematic, especially in applications like named entity recognition.

- **Generalization** While the study shows promising results for certain tasks and languages, it's essential to assess the generalization of these findings to a broader range of languages and tasks. What works for one language pair may not hold for others. Also, there are variations of languages used in each task which might affect the final assumptions. Furthermore, this study focuses on using mBERT and whether this generalizes to more recent architectures is an interesting research question to be considered in future work.

- **Evaluation Metrics** The types of evaluation metrics could affect the insight we extract from such experiments. In our study, we focused on using multiple evaluation metrics, especially for question answering and named entity recognition. In our NER experiments, we highlight the huge difference between using the F1 score vs. using the accuracy score in the evaluation.

References

Khadige Abboud, Olga Golovneva, and Christopher DiPersio. 2022. Cross-lingual transfer for low-resource arabic language understanding. In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 225–237.

Ahmed Abdelali, Kareem Darwish, Nadir Durrani, and Hamdy Mubarak. 2016. Farasa: A fast and furious segmenter for arabic. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Demonstrations*, pages 11–16.

Ahmed Abdelali, Hamdy Mubarak, Shammur Absar Chowdhury, Maram Hasanain, Basel Mousi, Sabri Boughorbel, Yassine El Kheir, Daniel Izham, Fahim Dalvi, Majd Hawasly, et al. 2023. Benchmarking arabic ai with large language models. *arXiv preprint arXiv:2305.14982*.

Mohamed Alkaoud and Mairaj Syed. 2020. On the importance of tokenization in arabic embedding models. In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 119–129.

Badr AlKhamissi, Muhammad N ElNokrashy, and Mohamed Gabr. 2020. Deep diacritization: Efficient hierarchical recurrence for improved arabic diacritization. *arXiv preprint arXiv:2011.00538*.

Zaid Alyafeai, Maged S Alshaibani, Badr AlKhamissi, Hamzah Luqman, Ebrahim Alareqi, and Ali Fadel. 2023. Taqyim: Evaluating arabic nlp tasks using chatgpt models. *arXiv preprint arXiv:2306.16322*.

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. 2019. On the cross-lingual transferability of monolingual representations. *CoRR*, abs/1910.11856.

Kaj Bostrom and Greg Durrett. 2020. Byte pair encoding is suboptimal for language model pretraining. *arXiv preprint arXiv:2004.03720*.

Jonathan H Clark, Dan Garrette, Iulia Turc, and John Wieting. 2021. Canine: Pre-training an efficient tokenization-free encoder for language representation. *arXiv preprint arXiv:2103.06874*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Alexis Conneau, Guillaume Lample, Ruty Rinott, Adina Williams, Samuel R Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. *arXiv preprint arXiv:1809.05053*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, Swetha Ranganath, Laurie Crist, Misha Britan, Wouter Leeuwis, Gokhan Tur, and Prem Natarajan. 2022. Massive: A 1m-example multilingual natural language understanding dataset with 51 typologically-diverse languages.

Daniela Gerz, Ivan Vulić, Edoardo Ponti, Jason Naradowsky, Roi Reichart, and Anna Korhonen. 2018. Language modeling for morphologically rich languages: Character-aware modeling for word-level prediction. *Transactions of the Association for Computational Linguistics*, 6:451–465.

Valentin Hofmann, Janet B Pierrehumbert, and Hinrich Schütze. 2020. Generating derivational morphology with bert. *arXiv preprint arXiv:2005.00672*.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *International Conference on Machine Learning*, pages 4411–4421. PMLR.

Karima Kadaoui, Samar M Magdy, Abdul Waheed, Md Tawkat Islam Khondaker, Ahmed Oumar El-Shangiti, El Moatez Billah Nagoudi, and Muhammad Abdul-Mageed. 2023. Tarjamat: Evaluation of bard and chatgpt on machine translation of ten arabic varieties. *arXiv preprint arXiv:2308.03051*.

Muhammad Khalifa, Muhammad Abdul-Mageed, and Khaled Shaalan. 2021. Self-training pre-trained language models for zero-and few-shot multi-dialectal arabic sequence labeling. *arXiv preprint arXiv:2101.04758*.

Md Tawkat Islam Khondaker, Abdul Waheed, El Moatez Billah Nagoudi, and Muhammad Abdul-Mageed. 2023. Gptaraeval: A comprehensive evaluation of chatgpt on arabic nlp. *arXiv preprint arXiv:2305.14976*.

David King, Andrea Sims, and Micha Elsner. 2020. Interpreting sequence-to-sequence models for russian inflectional morphology. In *Proceedings of the Society for Computation in Linguistics 2020*, pages 409–418.

Stav Klein and Reut Tsarfaty. 2020. Getting the## life out of living: How adequate are word-pieces for modelling complex morphology? In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 204–209.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. *arXiv preprint arXiv:1804.10959*.

Wuwei Lan, Yang Chen, Wei Xu, and Alan Ritter. 2020. Gigabert: Zero-shot transfer learning from english to arabic. In *Proceedings of The 2020 Conference on Empirical Methods on Natural Language Processing (EMNLP)*.

Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. 2020. From zero to hero: On the limitations of zero-shot cross-lingual transfer with multilingual transformers. *arXiv preprint arXiv:2005.00633*.

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

Mai Oudah, Amjad Almahairi, and Nizar Habash. 2019. The impact of preprocessing on arabic-english statistical and neural machine translation. *arXiv preprint arXiv:1906.11751*.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. Mad-x: An adapter-based framework for multi-task cross-lingual transfer. *arXiv preprint arXiv:2005.00052*.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual bert? *arXiv preprint arXiv:1906.01502*.

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

Alexandre Rafalovitch, Robert Dale, et al. 2009. United nations general assembly resolutions: A six-language parallel corpus. In *Proceedings of the MT Summit*, volume 12, pages 292–299.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Aquia Richburg, Ramy Eskander, Smaranda Muresan, and Marine Carpuat. 2020. An evaluation of subword segmentation strategies for neural machine translation of morphologically rich languages. In *Proceedings of the The Fourth Widening Natural Language Processing Workshop*, pages 151–155.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Ashish Vaswani et al. 2017. *Attention is all you need.* Advances in neural information processing systems.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*.

A Appendix

In Tables 5, 6, and 7, we show off the results also for finetuning and evaluating on the same language and on Arabic on zero-shot fashion. Mostly, we don't see any correlation between achieving high evaluation scores on the same language and then on Arabic.

Table 4: Language Codes

Language	**Code**	**Language**	**Code**
Afrikaans	af	Dutch	nl
Khmer	km	Polish	pl
Kannada	kn	Portuguese	pt
Korean	ko	Romanian	ro
Latvian	lv	Russian	ru
Malayalam	ml	Slovenian	sl
Mongolian	mn	Albanian	sq
Malay	ms	Swedish	sv
Burmese	my	Swahili	sw
Norwegian Bokmål	nb	Tamil	ta
Chinese	zh	Telugu	te
Amharic	am	Thai	th
Arabic	ar	Filipino	tl
Azerbaijani	az	Turkish	tr
Bengali	bn	Urdu	ur
Welsh	cy	Vietnamese	vi
Danish	da	English	en
German	de	Spanish	es
Greek	el	Persian	fa
Hindi	hi	Finnish	fi
Hungarian	hu	French	fr
Armenian	hy	Hebrew	he
Indonesian	id	Italian	it
Icelandic	is	Japanese	ja
Javanese	jv	Georgian	ka

Table 5: Results for question answering. Exact match and F1 scores are shown as the metrics.

v

	de	**zh**	**vi**	**es**	**hi**	**el**	**th**	**ro**	**ar**	**en**	**ru**	**tr**
EM	30.25	40.34	34.45	41.18	26.89	34.45	40.34	40.34	36.97	43.70	41.18	31.09
F1	45.99	51.03	51.25	56.03	36.69	47.00	46.62	52.37	51.56	57.10	55.81	40.21
EM$_{ar}$	30.25	28.57	25.21	28.57	24.37	31.09	24.37	31.93	36.97	31.93	36.13	25.21
F1$_{ar}$	45.91	42.10	40.80	40.80	37.89	43.16	36.32	46.80	51.56	46.23	48.95	38.22

Table 6: Results for named entity recognition. Accuracy and F1 scores are shown as the metrics.

	af	**am**	**ar**	**az**	**bn**	**cy**	**da**	**de**	**el**	**en**	**es**	**fa**	**fi**	**fr**	**he**	**hi**	**hu**
Ac	91.2	73.6	88.1	89.6	89.3	89.7	91.9	91.4	90.6	92.4	89.5	91.9	89.0	90.5	88.5	90.8	89.8
F1	69.6	3.8	65.8	69.5	65.1	63.5	73.0	70.2	69.0	74.4	66.8	71.1	69.2	68.7	65.6	65.2	68.9
Ac$_{ar}$	73.5	72.4	88.1	73.7	74.5	73.2	73.4	74.4	75.2	73.9	75.2	77.2	74.6	73.2	77.2	75.2	73.9
F1$_{ar}$	24.3	0.6	65.8	18.7	19.9	16.7	23.6	29.1	26.6	27.9	29.5	30.8	22.2	24.0	35.6	22.7	22.6

	hy	**id**	**is**	**it**	**ja**	**jv**	**ka**	**km**	**kn**	**ko**	**lv**	**ml**	**mn**	**ms**	**my**	**nb**	**nl**
Ac	89.0	89.5	90.2	89.8	91.6	89.1	86.5	68.9	86.9	89.3	89.4	88.2	87.8	90.0	91.5	91.5	91.6
F1	65.7	68.3	68.3	68.5	84.9	66.8	66.0	3.0	62.1	68.3	68.2	65.4	62.0	69.4	76.4	70.9	70.9
Ac$_{ar}$	73.3	75.7	73.3	74.4	71.9	74.8	74.0	69.2	72.4	73.3	73.2	72.2	72.4	74.9	73.0	75.2	75.2
F1$_{ar}$	19.3	25.8	20.9	30.1	18.4	23.2	19.2	0.9	12.5	12.8	20.7	12.9	5.6	25.7	8.1	26.3	26.0

	pl	**pt**	**ro**	**ru**	**sl**	**sq**	**sv**	**sw**	**ta**	**te**	**th**	**tl**	**tr**	**ur**	**vi**	**zh**
Ac	88.5	90.3	89.2	89.8	89.1	90.2	91.5	87.2	87.4	87.8	91.9	89.6	88.5	89.7	89.5	91.5
F1	65.5	69.2	67.2	69.7	67.9	66.9	72.4	62.9	64.6	62.6	80.8	64.0	65.7	61.3	64.9	85.6
Ac$_{ar}$	72.5	75.4	75.7	75.0	73.9	75.7	74.1	73.1	74.7	73.3	74.5	74.8	70.4	74.1	74.6	67.2
F1$_{ar}$	25.8	28.6	29.1	24.8	23.7	24.8	25.3	13.7	18.9	13.2	17.9	18.4	11.7	19.2	23.1	17.4

Table 7: Results for natural language inference. Accuracy is shown as the metric.

	ar	**bg**	**de**	**el**	**en**	**es**	**fr**	**hi**	**ru**	**sw**	**th**	**tr**	**ur**	**vi**	**zh**
Ac	64.0	69.4	70.1	67.6	76.2	71.6	70.9	63.3	68.8	59.0	59.1	65.7	57.7	69.7	70.8
Ac$_{ar}$	64.0	63.5	63.9	63.1	64.1	63.3	62.8	61.6	64.0	57.1	55.0	60.9	54.4	63.1	63.5