A Focused Study to Compare Arabic Pre-training Models on Newswire IE Tasks

Wuwei Lan, Yang Chen, Wei Xu, Alan Ritter
Department of Computer Science and Engineering
The Ohio State University
{lan.105, chen.9279, xu.1265, ritter.1492}@osu.edu

Abstract

The Arabic language is a morphological rich language, posing many challenges for information extraction (IE) tasks, including Named Entity Recognition (NER), Part-of-Speech tagging (POS), Argument Role Labeling (ARL) and Relation Extraction (RE). A few multilingual pre-trained models have been proposed and show good performance for Arabic, however, most experiment results are reported on language understanding tasks, such as natural language inference, question answering and sentiment analysis. Their performance on the IE tasks is less known, in particular, the cross-lingual transfer capability from English to Arabic. In this work, we pre-train a Gigaword-based bilingual language model (GigaBERT) to study these two distant languages as well as zero-short transfer learning on the information extraction tasks. Our GigaBERT model can outperform mBERT and XLM-R_{base} on NER, POS and ARL tasks, with regarding to the per-language and/or zero-transfer performance. We make our pre-trained models publicly available at https://github.com/lanwuwei/GigaBERT to facilitate the research of this field.

1 Introduction

Recently, the pre-trained models (Peters et al., 2018; Devlin et al., 2019; Yang et al., 2019) have greatly improved performance for many NLP tasks, making "pre-training and fine-tuning" a new paradigm in this field. In addition to English language, these pre-trained models have enabled advances for many other languages, including AraBERT for Arabic (Antoun et al., 2020), CamemBERT for French (Martin et al., 2019), ERNIE for Chinese (Sun et al., 2019) and etc. Instead of costly pre-training language model for every langauge, Google releases a multilingual BERT (mBERT)[1] for 104 languages, but it shows lower performance compared to single-language pre-trained models. Some other multilingual pre-training models further improve the mBERT performance, for example, XLM (Lample and Conneau, 2019) introduces translation language model with bitext; XLM-R (Conneau et al., 2020) optimizes the BERT model and increases size of the pre-training data. However, all these pre-trained models focus on the downstream evaluations of 'high-level' natural language understanding (NLU) tasks, such as natural language inference, paraphrase identification, question answering, sentiment analysis, etc. Very few of them are evaluated on the 'low-level' information extraction tasks, such as Named Entity Recognition (NER), Part-of-Speech tagging (POS), Argument Role Labeling (ARL) and Relation Extraction (RE). Especially for morphologically rich languages (e.g., Arabic), where the language varieties pose many challenges for these pre-training models. The Arabic language has almost no shared scripts with English, which creates another challenge for cross-lingual transfer.

To address the above problems, we pre-train a bilingual language model primarily based on Giagword data (GiagBERT) for Arabic and English IE tasks, and systematically compare our pre-trained models with existing AraBERT (ElJundi et al., 2019), mBERT (Devlin et al., 2019) and XLM-R (Conneau et al., 2020). We find that the pre-trained models can do well for 'low-level' information extraction tasks, as well as the zero-shot transfer, even they are distant English and Arabic languages. Our bilingual GiagBERT performs better than mBERT and XLM-R_{base} (both support more than 100 languages) on NER, POS and ARL tasks, demonstrating that multilingual PTM actually sac-

[1] https://github.com/google-research/bert/blob/maste

rifices per-language performance. Our GigaBERT also outperforms monolingual AraBERT and potentially provides a good resource for Arabic NLP research.

2 Related Work

2.1 Monolingual Pre-training Models

There are only two publicly available pre-trained models for Arabic language: hULMonA (ElJundi et al., 2019) and AraBERT (ElJundi et al., 2019). The hULMonA is based on the AWD-LSTM architecture (Merity et al., 2017) and pre-trained with 600K Wikipedia articles, which shows state-of-the-art performance on Arabic sentiment analysis task. Recently, this model was outperformed by AraBERT model, a pre-trained Arabic BERT on large-scale news corpus as well as the Wikipedia data. Not only in sentiment analysis, the AraBERT also shows SOTA performance in question answering and named entity recognition.

2.2 Multilingual Pre-training Models

Several multilingual pre-trained models have been proposed to handle tens or over a hundred of languages within one model, where Arabic language is also included. The LASER model (Artetxe and Schwenk, 2019) utilizes the parallel data of 93 languages and pre-trains a BiLSTM based encoder-decoder, where the BiLSTM encoder is used for downstream evaluations. Similar to LASER, the MASS (Song et al., 2019) model also has a encoder-decoder framework, but utilizes both encoder and decoder for improving generation tasks. The mBERT (Devlin et al., 2019) is pre-trained on the Wikipedia dump of 104 languages with 12-layer Transformer (Vaswani et al., 2017) encoder. Compared to mBERT, the XLM model (Lample and Conneau, 2019) pre-trained BERT with only 15 languages (Arabic included), and has a Transition Language Model (TLM) objective in addition to improve the performance. Recently, XLM-R model (Conneau et al., 2020) shows that pre-training on large-scale high-quality data leads to significant performance gains for a wide range of cross-lingual transfer tasks. It also shows that the multilingual pre-trained models can performa better than single-language without sacrificing per-language performance.

3 Bilingual Language Model: GigaBERT

Our GigaBERT is based on BERT, a Bidirectional Transformer Encoder with Masked Language Model (MLM) and Next Sentence Prediction (NSP) pre-training objective (Devlin et al., 2019), but use a different setup for pre-training data selection, vocabulary set construction, subword units segmentation and hyper-parameters. In order to better understand the effects of different factors, we propose three versions of GigaBERT with varied pre-training data source and vocabulary setup (Table 1).

Architecture All three versions of GigaBERT use the $\text{BERT}_{\text{base}}$ configurations: 12 attention layers, each layer has 12 attention heads and 768 hidden dimensions. The max position embeddings have 512 and the hidden dimension of feed forward layer is 3072. The mBERT and $\text{XLM-R}_{\text{base}}$ (Conneau et al., 2020) also have the same $\text{BERT}_{\text{base}}$ architecture, which can be fairly compared with our GigaBERT.

Pre-training data We use the fifth edition of English Gigaword (LDC2011T07) and Arabic Gigaword (LDC2011T11) for pre-training GigaBERT-v0. We flatten[2] the raw Gigaword data and split English sentences with modified Stanford CoreNLP tool (Manning et al., 2014) and Arabic sentences with period, exclamation mark, and question mark. In addition, we add Wikipedia dump processed with WikiExtractor[3] for better coverage in GigaBERT-v1 and GigaBERT-v2. Since the pre-training data is unbalanced for two languages, we augment the Arabic part by two ways: (1) up-sample the Arabic data by repeating the Wikipedia data five times and the Gigaword data three times; (2) add the Arabic shuffled Oscar data (Ortiz Suárez et al., 2019), a large-scale multilingual dataset obtained by language identification and filtering of the Common Crawl corpus. These two ways are used in GigaBERT-v1 and GigaBERT-v2 respectively.

Vocabulary The vocabulary size is critical to the pre-training performance, as it affects the subword granularity and model parameters directly. Considering that our pre-training data has at most $\sim$10B tokens, while the original English BERT model has 30k vocabulary size for $\sim$3B tokens, the mBERT and XLM-R has about 5k and 10k subwords for Arabic in their vocabulary respectively,

[2] https://github.com/nelson-liu/flatten_gigaword
[3] https://github.com/attardi/wikiextractor

Models	Training Data		Vocabulary			Configuration	
	source	#tokens (en / ar)	tokenization	size	cased	architecture	#parameters
AraBERT	News	– / 2.5B	SentencePiece	64k	no	$\text{BERT}_{\text{base}}$	136M
mBERT	Wiki	2.5B / 153M	WordPiece	110k	yes	$\text{BERT}_{\text{base}}$	172M
$\text{XLM-R}_{\text{base}}$	CommonCrawl	55.6B / 2.9B	SentencePiece	250k	yes	$\text{BERT}_{\text{base}}$	270M
$\text{XLM-R}_{\text{large}}$	CommonCrawl	55.6B / 2.9B	SentencePiece	250k	yes	$\text{BERT}_{\text{large}}$	550M
GiagBERT-v0	Giga	3.6B / 1.1B	SentencePiece	50k	yes	$\text{BERT}_{\text{base}}$	125M
GigaBERT-v1	Giga, Wiki	6.1B / 1.3B	WordPiece	50k	yes	$\text{BERT}_{\text{base}}$	125M
GigaBERT-v2	Giga,Wiki,Oscar	6.1B / 4.3B	WordPiece	50k	no	$\text{BERT}_{\text{base}}$	125M

Table 1: Configuration comparisons for AraBERT (ElJundi et al., 2019), mBERT (Devlin et al., 2019), XLM-RoBERTa (Conneau et al., 2020) and GigaBERT (this work).

we decided to use 50k for our vocabulary size. For GigaBERT-v0, we segment the data using SentencePiece (Kudo and Richardson, 2018) with unigram language model. We build 30k cased English sub-words and 20k Arabic sub-words, then merge them to construct 50k bilingual vocabulary.[4] For GigaBERT-v1 and GigaBERT-v2, we did not distinguish Arabic and English sub-word units, instead, we train a unified 50k vocabulary with WordPiece model.[5] The vocabulary is cased for GigaBERT-v1 and uncased for GigaBERT-v2.

Pre-training objective Following the original BERT model (Devlin et al., 2019), we use the Masked Language Modeling (MLM) and the Next Sentence Prediction (NSP) to pre-train our GigaBERT. During MLM, 15% of the tokens are randomly masked out, the model is trained to predict the masked tokens with the rest of tokens. The NSP task is just to distinguish whether two sentences are continuous segments or not.

Hyper-parameters We pre-train GigaBERT with batch size of 1024 sentences and max sequence length of 128 for 1.2 million steps. We use Adam optimizer with learning rate of 1e-4 and warmup steps of 10k. The number of predictions per sentence is set to 20. We use whole word mask for GigaBERT-v0 and regular mask mechanism for GigaBERT-v1 and GigaBERT-v2.

4 Evaluation

We use named entity recognition, part-of-speech tagging, argument role labeling and relation extraction as downstream tasks for evaluating pre-trained models. All the evaluations follow the same fine-tuning procedure: the original sentences are fed into pre-trained model, then we extract the necessary hidden representations (i.e., all token representations for NER and POS, the argument span for ARL, and the entity span for RE) and apply linear classification. We evaluate performance for each language as well as the zero-shot transfer from English to Arabic, where the model is trained in English training set and evaluated in Arabic development and test set.

4.1 Downstream Tasks

Named Entity Recognition (NER) We use the nested named entity recognition dataset from ACE 2005 (LDC2006T06), where the train, dev and test examples for English are 7634, 1005, 1095 respectively, while for the Arabic part are 2683, 322, 238 respectively.[6] The evaluation metric is based on F_1 score.

Part-of-Speech Tagging (POS) The POS tagging dataset is from Universal Dependencies (UD) Treebanks v1.4 (Nivre et al., 2016), where the train, dev and test examples for English are 12543, 2002, 2077 respectively, while for the Arabic part are 6174, 786, 704 respectively. The evaluation metric is based on accuracy.

Argument Role Labeling (ARL) We use ACE 2005 dataset (LDC2006T06) for ARL task and randomly split train/dev/test for both languages, where the train, dev and test examples for English are 12836, 1340, 1681 respectively, while for the Arabic part are 6301, 908, 862 respectively. The evaluation metric is based on F_1 score.

Relation Extraction (RE) The dataset for RE

[4] 633 sub-words are shared by both languages, we add different UNK symbols to compose 50k vocabulary.

[5] We use the implementation of Hugging Face's tokenizers library: https://github.com/huggingface/tokenizers

[6] The Arabic train/dev/test is randomly split by ourselves while the English split is from official release.

Models	NER			POS			ARL			RE		
	en	ar	en→ ar	en	ar	en→ ar	en	ar	en→ ar	en	ar	en→ ar
AraBERT	-	78.6	-	-	97.6	-	-	81.6	-	-	**88.1**	-
mBERT	80.3	72.9	31.1	97.0	97.3	50.4	77.1	73.5	57.5	84.5	84.1	70.7
XLM-R_{base}	81.0	72.9	42.2	**97.8**	97.6	**59.5**	74.1	77.5	68.5	84.1	87.6	76.1
GigaBERT-v0	79.1	76.6	37.9	95.9	97.5	54.1	75.1	70.1	61.6	83.1	68.2	49.5
GigaBERT-v1	82.8	72.9	36.4	97.1	96.6	52.2	77.7	70.4	58.0	86.3	74.4	55.6
GigaBERT-v2	82.5	75.2	48.2	97.2	**97.8**	53.4	**79.2**	75.4	66.4	86.2	79.9	70.1
GigaBERT-v3	**83.4**	**83.1**	48.3	97.1	**97.8**	54.7	76.4	**83.5**	**68.9**	**86.6**	87.7	**76.4**
XLM-R_{large}	85.8	84.8	50.4	97.3	97.8	61.2	78.1	82.3	74.0	85.4	88.1	78.8

Table 2: Downstream evaluations for AraBERT (ElJundi et al., 2019), multilingual BERT (Devlin et al., 2019), XLM-RoBERTa (Conneau et al., 2020) and GigaBERT (this work). All models use $BERT_{base}$ architecture except the XLM-R_{large}. The GigaBERT-v3 is continued pre-training of GigaBERT-v2 for extra 100k steps with 512 max sequence length.

task is also from ACE 2005 (LDC2006T06). We randomly split train/dev/test for both languages, where the train, dev and test examples for English are 13761, 1365, 1619 respectively, while for the Arabic part are 7500, 1080, 874 respectively. The evaluation metric is F_1 score.

4.2 Experimental Setup

Pre-training We use the original BERT implementation[7] in our experiment. We deploy TensorFlow version 1.5 and TPU v2-8 in Google Cloud Platform, and set up storage buckets for data access. Based on the performance of downstream evaluation tasks, the best checkpoints are selected at around 1 million steps. We continue pre-training the best version of GigaBERT with max sequence length 512 for extra 100k steps.

Fine-tuning We run extensive grid search to find best hyper-parameters for each pre-trained model, the main hyper-parameters we explored are batch size (8, 16, 32), learning rate (1e-5, 2e-5, 3e-5, 5e-5, 1e-4, 2e-4), and the number of fine-tuning epochs (3, 7, 10).

4.3 Results and Analysis

We report performance for these IE tasks of each language as well as the zero-shot transfer performance in Table 2. As pre-training data increases from GigaBERT-v0 to GigaBERT-v2, the performance on downstream tasks improves. In particular, adding Oscar data in GigaBERT-v2 is more effective than up-sampling in GigaBERT-v1. Our GigaBERT-v2 performs best among all three versions, which also outperforms mBERT on NER, POS and ARL, and XLM-R_{base} on NER. Some training examples the in RE dataset are out of 128 max sequence length for pre-trained GigaBERT, which will be truncated during fine-tuning and causing low performance in the RE task. Therefore we select the best GigaBERT-v2 and continue pre-training it for extra 100k steps with max sequence length 512 to create GigaBERT-v3 in Table 2. The GigaBERT-v3 has the best performance on NER, ARL and RE among all the pre-trained models with $BERT_{base}$ architecture. We find that the zero-shot performance for every task is at least 10 points lower than per-language performance, the NER and POS tasks have even 30 points lower, indicating a large improvement space on the cross-lingual capability for pre-trained models. As for the per-language case, the AraBERT outperforms mBERT and XLM-R_{base}, but still has lower performance than our GigaBERT-v3, especially for the NER and ARL task, which indicates the potential usefulness of our GigaBERT for Arabic NLP study. The performance gap between GigaBERT-v3 and XLM-R_{large} comes from the model size and data size, given the limited compute resources, we leave the GigaBERT pre-training with $BERT_{large}$ architecture for future works.

5 Conclusion

We pre-trained a bilingual GigaBERT for Arabic and English and conducted a focused study of pre-trained models for IE tasks. The downstream evaluations show that our GigaBERT out-

[7] https://github.com/google-research/bert

performs state-of-the-art multilingual pre-trained models (mBERT and XLM-R_{base}) and the monloingual AraBERT on named entity recognition, part-of-speech tagging and argument role labeling. Our pre-trained GigaBERT provides a good resource for Arabic natural language processing and the cross-lingual transfer study between English and Arabic.

Acknowledgement

We thank Nizar Habash for his helpful advice and suggestions. This research is supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via the BETTER Program contract # 2019-19051600004. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

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

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics (TACL)*.

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 (ACL)*.

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 (NAACL-HLT)*.

Obeida ElJundi, Wissam Antoun, Nour El Droubi, Hazem Hajj, Wassim El-Hajj, and Khaled Shaban. 2019. hULMonA: The universal language model in Arabic. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations (EMNLP)*.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations (ACL)*.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, 'Eric Villemonte de la Clergerie, Djamé Seddah, and Benoît Sagot. 2019. Camembert: a tasty french language model. *ArXiv*.

Stephen Merity, Nitish Shirish Keskar, and Richard Socher. 2017. Regularizing and optimizing lstm language models. *arXiv preprint arXiv:1708.02182*.

Joakim Nivre, Željko Agić, Lars Ahrenberg, Maria Jesus Aranzabe, Masayuki Asahara, Aitziber Atutxa, Miguel Ballesteros, John Bauer, Kepa Bengoetxea, Yevgeni Berzak, Riyaz Ahmad Bhat, Eckhard Bick, Carl Börstell, Cristina Bosco, Gosse Bouma, Sam Bowman, Gülşen Cebiroğlu Eryiğit, Giuseppe G. A. Celano, Fabricio Chalub, Çağrı Çöltekin, Miriam Connor, Elizabeth Davidson, Marie-Catherine de Marneffe, Arantza Diaz de Ilarraza, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Marhaba Eli, Tomaž Erjavec, Richárd Farkas, Jennifer Foster, Claudia Freitas, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Moa Gärdenfors, Sebastian Garza, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta Gonzáles Saavedra, Matias Grioni, Normunds Grūzītis, Bruno Guillaume, Jan Hajič, Linh Hà Mỹ, Dag Haug, Barbora Hladká, Radu Ion, Elena Irimia, Anders Johannsen, Fredrik Jørgensen, Hüner Kaşıkara, Hiroshi Kanayama, Jenna Kanerva, Boris Katz, Jessica Kenney, Natalia Kotsyba, Simon Krek, Veronika Laippala, Lucia Lam, Phuong Lê H`ông, Alessandro Lenci, Nikola Ljubešić, Olga Lyashevskaya, Teresa Lynn, Aibek Makazhanov, Christopher Manning, Cătălina Mărănduc, David Mareček, Héctor Martínez Alonso, André Martins, Jan Mašek, Yuji Matsumoto, Ryan McDonald, Anna Missilä, Verginica Mititelu, Yusuke Miyao, Simonetta Montemagni, Keiko Sophie Mori, Shunsuke Mori, Bohdan Moskalevskyi, Kadri Muischnek, Nina Mustafina, Kaili Müürisep, Lu'o'ng Nguy˜ên Thị, Huy`ên Nguy˜ên Thị Minh, Vitaly Nikolaev, Hanna Nurmi, Petya Osenova, Robert Östling, Lilja Øvrelid, Valeria Paiva, Elena Pascual, Marco Passarotti, Cenel-Augusto Perez, Slav Petrov, Jussi Piitulainen, Barbara Plank, Martin Popel, Lauma Pretkalniņa, Prokopis Prokopidis, Tiina Puolakainen, Sampo Pyysalo, Alexandre Rademaker, Loganathan Ramasamy, Livy Real, Laura Rituma, Rudolf Rosa, Shadi Saleh, Baiba Saulīte, Sebastian Schuster, Wolfgang Seeker, Mojgan Seraji, Lena Shakurova, Mo Shen, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Aaron Smith, Carolyn Spadine, Alane Suhr, Umut Sulubacak, Zsolt Szántó, Takaaki Tanaka, Reut Tsarfaty, Francis Tyers, Sumire Uematsu, Larraitz Uria, Gertjan van Noord, Viktor Varga, Veronika Vincze, Lars Wallin, Jing Xian Wang, Jonathan North Washington, Mats Wirén, Zdeněk Žabokrtský, Amir Zeldes, Daniel Zeman, and Hanzhi Zhu. 2016. Universal dependencies 1.4. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous Pipeline for Processing Huge Corpora on Medium to Low Resource Infrastructures. In *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7)*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. Mass: Masked sequence to sequence pre-training for language generation. In *International Conference on Machine Learning (ICML)*.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Xuyi Chen, Han Zhang, Xin Tian, Danxiang Zhu, Hao Tian, and Hua Wu. 2019. Ernie: Enhanced representation through knowledge integration. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*.

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

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems (NeurIPS)*.