AJEMA GLOBAL RESEARCH NETWORK

AMERICAN Journal of Engineering, Mechanics and Architecture
Volume 2, Issue 9, 2024 ISSN (E): 2993-2637

Top Accurate Models for Handling Complex Arabic Linguistic Structures

Murteza Hanoon Tuama, Wahhab Muslim mashloosh, Yasir Mahmood Younus
Department of Computer Techniques Engineering, Imam Al-Kadhum University College, Baghdad, Iraq

Abstract: Arabic, a language rich in morphology but deficient in resources and syntactical exploration when compared to English, poses major hurdles for Applications of Arabic Natural Language Processing (NLP) include Question Answering, Named Entity Recognition (NER), and Sentiment Analysis (SA). (QA). However, recent advances in transformer-based models have demonstrated that language-specific BERT models, when pre-trained on large corpora, outperform in Arabic comprehension. These models have set new benchmarks and produced outstanding outcomes across a wide range of NLP tasks. In this study, we offer AraBERT, a BERT model built exclusively for Arabic, with the goal of replicating BERT's success in English. We compare AraBERT to Google's multilingual BERT and other cutting-edge techniques. The results revealed that the newly designed AraBERT outperformed most Arabic NLP.

Keywords: Natural Language Processing, Arabic language, Sentiment Analysis.

INTRODUCTION

Natural Language Understanding (NLU) tasks have significantly improved with the inclusion of contextualized text representation models, yielding top-tier Applications of Arabic Natural Language Processing (NLP) include Question Answering, Named Entity Recognition (NER), and Sentiment Analysis (SA).. However, these models neglected to take the surrounding context into account when creating an embedding for a word. Contextualized representations were added using models like ELMO in order to get over this restriction (Peters et al., 2018). Utilizing transfer learning to enhance the performance of downstream NLP/NLU tasks has received a lot of attention lately. This is achieved by employing a limited collection of cases to fine-tune large pre-trained language models. Utilizing this technique has improved job performance significantly. Using language models that have been pre-trained in an unsupervised manner—a process known as self-supervised learning—offers the main advantage. It's important to understand this technique's limits, though. The requirement for sizable corpora in the pre-training phase is one such drawback. Moreover, the computational expense is substantial, as the existing models need the use of more than 500 TPUs or GPUs running for many weeks (Conneau et al., 2019; Raffel et al., 2019). The application of such models is restricted to English and a few other languages due to these issues. In an effort to close this disparity, multilingual models are trained to simultaneously acquire representations for more than 100 languages; yet, because of their small vocabulary and sparse representation of the data, multilingual models still lag behind monolingual models. Shared representations can help languages with similar vocabulary and structure (Conneau et al., 2019), but not all languages can benefit from them. For example, Arabic differs greatly from most other Latin-based languages in terms of its morphological and

syntactic structures. In this study, we address the process of translating the Arabic language, or "arabert," from the BERT Transformer model (Devlin et al., 2018). Three distinct Arabic NLU downstream tasks are used to evaluate ARABERT: (i) Question Answering (QA), (ii) Named Entity Recognition (NER), and (iii) Sentiment Analysis (SA). The results of the trial show that ARABERT performs better than several baselines, including previous single- and multilingual methods.

We explore two datasets for downstream tasks: Dialectal Arabic (DA) and Modern Standard Arabic (MSA). Our input may be summed up as follows:

- A method for applying the BERT model to three downstream NLU tasks: named entity recognition, question answering, and sentiment analysis;
- ARABERT is applied to these tasks using a large Arabic corpus.
- ARABERT is accessible to the general public on well-known NLP libraries.

The remainder of this essay is structured as follows: An overview of earlier studies on the linguistic representation of Arabic and English is given in Section 2. Section 3 goes into the methodology used to develop ARABERT. The benchmark dataset used in the evaluation and the downstream tasks are described in Section 4. Section 5 describes the experimental design and examines the findings. In conclusion, Section 6 offers suggestions for future study subjects and draws conclusions.

RELATED WORKS

Put Word Inclusions Together Mikolov et al. (2013) first developed meaningful word representations using their word2vec technique. Subsequently, research has concentrated on word2vec variations such as GloVe (Pennington et al., 2014) and Fast Text (Mikolov et al., 2017). Despite their significant improvements, these early models still required contextual information, which was addressed by ELMO (Peters et al. 2018). Better word and phrase representations and bigger structures are the results of increased performance on a range of activities. Many language comprehension models have since been created, such as ALBERT (Lan et al., 2019), T5, and others. (Raffel et al., 2019), RoBERTa (Liu et al., 2019), XLNet (Yang et al., 2019), BERT (Devlin et al., 2018), and ULMFit (Howard and Ruder, 2018). These models enhance performance by experimenting with various correlation techniques, altered model topologies, and bigger training corpora.

Isolated Arabic representations NLP researchers tried to reproduce word2vec's (Mikolov et al., 2013) success with English in order to generate language-specific embeddings. After being tried by Soliman et al. (2017), the Fasttext model (Bojanowski et al., 2017) performed better in Arabic than word2vec, thanks to training on Wikipedia data. While word embeddings for Arabic learnt on more than 250 million tweets were proposed by (Abu Farha and Magdy, 2019; Abdul-Mageed et al., 2018), (Erdmann et al., 2018) provided methods to train multi-dialect word embeddings on a relatively small and noisy corpus to manage dialectal heterogeneity in Arabic.

Contextualized Representations in Arabic A multilingual BERT developed by Google (Devlin et al., 2018) covers over 100 languages and performs well for most of them, particularly non-English languages. However, pre-training monolingual BERT has been demonstrated to perform better than multilingual BERT for non-English languages, as the publically accessible BERTs (Martin et al., 2019; de Vries et al., 2019) and the Italian BERT Alberto (Polignano et al., 2019). ElJundi et al. (2019) reported that Arabic-specific contextualized representations models, such as hULMonA, employed the ULMfit structure and outperformed BERT on English natural language processing tests.

ARABERT

Techniques In order to perform better on a number of Arabic natural language understanding challenges, we have developed an Arabic language representation model in this study. We utilize the stacked Bidirectional Transformer Encoder (BERT) model as the foundation for ARABERT (Devlin et al., 2018). In many different NLP tasks across several languages, this model is regarded as the foundation for the majority of state-of-the-art outcomes. We employ the BERTbase configuration, comprising of about 110 million parameters, 12 attention heads, 12 encoder blocks, 768 hidden dimensions, and 512 maximum sequence length. To improve the model's fit to the Arabic language, we also included more preprocessing before the pretraining phase. Here, we outline the pre-training setup, the ARABERT pre-training dataset, the fine-tuning technique, and the advised Arabic preprocessing.

Pre-training Setup

To achieve the original BERT pre-training goal, we use whole-word masking in addition to the Masked Language Modeling (MLM) task, where 15% of the N input tokens are chosen for replacement. Ten percent is a random token, ten percent is the [MASK] token, and eighty percent is the [MASK] token in place of the original token. Wholeword masking improves the pre-training challenge by forcing the model to predict the complete word instead of just partial recommendations. Furthermore, by letting the model comprehend the connection between two phrases, we use the Next Sentence Prediction (NSP) task, which is useful for numerous language understanding tasks such as Question Answering.

Pre-training Dataset

Zhu et al. (2015) state that 3.3 billion words from the English Wikipedia and the Book Corpus were used to train the first BERT. We personally collected stories from Arabic news sources because there are less items in the Arabic Wikipedia Dumps than there are in the English ones. Furthermore, we employed two Arabic corpora that are accessible to the public: (1) the Open Source International Arabic News Corpus (Zeroual et al., 2019), which is a modern corpus comprising over 5 million articles sourced from 10 major news sources across eight countries; and (2) the 1.5 billion words Arabic Corpus (El-Khair, 2016). This comprises 3.5 million articles (about 1 billion tokens) from 24 Arab countries' 31 news sources. The ultimate size of the pre-training dataset, once superfluous phrases are eliminated, is 70 million sentences, or around 24 terabytes of text. This dataset can serve as a representative sample for a wide range of topics covered in the Arab world because it includes news from various media outlets in various Arab regions. It is important to highlight that we kept Latin-character phrases since, in order to prevent information loss, it is standard procedure to refer to recognized entities, scientific terms, and technical terms in their native tongue.

Segmentation of Sub-Word

Units Because of its intricate concatenative structure, The lexical sparsity of Arabic is well known (Al-Sallab et al., 2017). Words may take on several forms without changing their meaning. The English counterpart of "the," for instance, is "ال - Al," the definite article, which is always appended to other nouns but does not constitute a part of the word itself. Consequently, tokens will show up twice when using a BERT-compatible tokenization: once with and once without the "Al-" prefix. For instance, "كتاب- kitAb" and "الكتاب-AlkitAb" are examples of terms that need to be included in the lexicon, which results in a great deal of needless repetition. In order to circumvent this problem, we first divide the words into stems, prefixes, and suffixes using Farasa (Abdelali et al., 2016). "اللغة - Alloga," for example, becomes "ال+لغ+ة - Al+ log +a." Next, we trained a Sentence Piece (an unsupervised utilizing the segmented pre-training dataset, text tokenizer and detokenizer (Kudo, 2018) in unigram mode. As a result, we were able to generate about 60K tokens of subword vocabulary. Sentence Piece was trained again on non-segmented text to produce ARABERTv0.1, a second version of the suggested tokenization

method that does not need segmentation, in order to assess its effectiveness. The final vocabulary size was 64k tokens, with an additional 4K tokens for possible pre-training.

Optimizing Sequence Categorization

To fine-tune AraBERT for sequence classification, we use the final hidden state of the initial token, which corresponds to the word embedding of the distinct "[CLS]" token appended at the start of each sentence. Next, we integrate a simple feed-forward layer and determine the probability distribution across the expected output classes by applying the traditional Softmax approach. Throughout the fine-tuning phase, the classifier and the pre-trained model weights collaborate to maximize the log-probability of the correct class.

Identification of Named Entities The IOB2 format for the NER task (Ratnaparkhi, 1998) is used to label each token in the sentence. The initial word of the entity is indicated by the “B” tag, the subsequent words are shown by the “I” tag, and the fact that the tagged word is not a desired named entity is indicated by the “O” tag. In order to label the tokens using various text classification techniques, we thus treat the system as a multi-class classification process. Furthermore, we used the AraBERT tokenizer to ensure that the model was only supplied the first sub-token of each word.

Answering Questions Given a question and a passage that provides the solution, the model in the QA has to select a part of text that contains the answers. This is achieved by anticipating a "start" token and a "end" token, with the condition that the "end" token come first. During training, each token's final embedding in the text is obtained by two classifiers, each of which applies a single set of weights to each token. To create a probability distribution for every token, a softmax layer is given the output embeddings and the dot product of the classifier. The process is then repeated for the "end" token, and the token with the highest probability of becoming a "start" token is selected.

EVALUATION

Sentiment analysis, named entity recognition, and question answering are the three downstream tasks in Arabic language understanding that we used to test ARABERT. We benchmarked ARABERT against the multilingual BERT version and against other state-of-the-art performance on each job.

Sentiment Analysis

The following Arabic sentiment datasets, which span several genres, domains, and dialects, were used to assess ARABERT.

- HARD: Elnagar et al. (2018)'s Hotel Arabic evaluations Dataset includes 93,700 hotel evaluations written in dialectal Arabic in addition to Modern Standard Arabic (MSA). Reviews are categorized as either favorable or negative, with a rating of 1 or 2, a rating of 4 or 5, and a value of 3 for neutral reviews that were disregarded.
- ASTD: Nabil et al. (2015) compiled 10,000 tweets in both MSA and Egyptian dialect from the Arabic Sentiment Twitter Dataset. We conducted our tests using the ASTD-B, or balanced form of the dataset.
- ArSenTD-Lev: Four thousand tweets written in Levantine dialect with annotations for sentiment, topic, and sentiment target make up the Arabic Sentiment Twitter Dataset for LEVantine (Baly et al., 2018). The fact that the tweets in this dataset originate from various domains and cover a range of subjects makes it difficult to use.
- LABR: 63,000 Arabic-language book evaluations may be found in the Large-scale Arabic Book evaluations dataset (Aly and Atiya, 2013). Ratings for the reviews range from 1 to 5. The imbalanced two-class dataset, where evaluations with ratings of 1 or 2 are regarded as unfavorable and those with ratings of 4 or 5 as positive, is used as a baseline for our model.

- AJGT: There are 1,800 tweets written in Jordanian dialect in the Arabic Jordanian General Tweets dataset (Alomari et al., 2017). Positive or negative annotations were manually made to the tweets.

Baselines: In Arabic, sentiment analysis is a common NLP activity. Earlier methods depended on sentiment lexicons like ArSenL (Badaro et al., 2014), a large-scale MSA word lexicon created by combining the English SentiWordNet with the Arabic WordNet. Numerous methods for Arabic-specific processing were investigated for recurrent and recursive neural networks (Al Sallab et al., 2015; Al-Sallab et al., 2017; Baly et al., 2017). Pre-taught word embeddings were used to train Convolutional Neural Networks (CNN) (Dahou et al., 2019a). A hybrid model was presented by (Abu Farha and Magdy, 2019), in which LSTMs handled sequence and context interpretation while CNNs handled feature extraction. According to Howard and Ruder's (2018) ULMfit architecture, the hULMonA model (ElJundi et al., 2019) is an Arabic language model that produces state-of-the-art outcomes. We compare the results from hULMonA and ARABERT.

Named Entity Recognition

The purpose of this assignment is to locate and extract identified things from the text. The work is structured as a word-level categorization (or tagging) task, with classes corresponding to pre-established groups including people, places, businesses, occasions, and expressions of time. We employ the Arabic NER corpus (ANERcorp) for assessment (Benajiba and Rosso, 2007). 16.5K entity references are included in this dataset, which are split up into 4 categories: person (39%), organization (30.4%), place (20.6%), and miscellaneous (10%).

Baselines: The CoNLL 2003 (Sang and De Meulder, 2003) dataset has been the focus of English-language advances in the NER problem. Conditional Random Fields (CRF) were first used to address NER (Lafferty et al., 2001). Subsequently, CRFs were applied to Bi-LSTM models, showing notable gains over standalone CRFs (Huang et al., 2015; Lample et al., 2016). Next, contextualized embeddings were applied to Bi-LSTM-CRF structures, exhibiting further improvements (Peters et al., 2018). In conclusion, sizable pre-trained transformers demonstrated a minor enhancement, establishing the present benchmark for performance (Devlin et al., 2018). Regarding Arabic, we contrast ARABERT's performance with both BERT multilingual and the Bi-LSTM-CRF baseline, which established the prior state-of-the-art performance (El Bazi and Laachfoubi, 2019).

Responding to Inquiries

One of the objectives of artificial intelligence is open-domain question answering (QA), which may be accomplished by utilizing knowledge acquisition and natural language comprehension (Kwiatkowski et al., 2019). Large datasets like the Stanford Question Answering Dataset (SQuAD) have been made available, which has stimulated research in English quality assurance (Rajpurkar et al., 2016). However, the absence of large datasets and the unique difficulties that come with Arabic QA have hampered study in this area. These difficulties include:

- Name spelling variations (e.g., Syria can be spelled as "سوريا - sOriyA" or "سوريةƒ - sOriyT" in Arabic).
- Renaming (for example: "عبد العزيز - AbdulAzIz" appears in the query, but "عبد العزيز - Abdul AzIz" appears in the response).
- The dual form "المثنى," which can take on many forms (for example, "قلمان" - "qalamAn" or "قلمين" - "qalamyn" - "two pencils").
- Grammatical gender variation: all nouns, animate and inanimate objects are classified under two genders either masculine or feminine (ex: "كبير - "kabIr" and " كبيرة" - "kabIrT".

The Arabic Reading Comprehension Dataset (ARCD) (Mozannar et al., 2019) is the dataset used to assess ARABERT. The job involves determining the span of an answer in a document for a

given query. In addition to 2966 machine-translated questions and answers from the SQuAD, renamed ArabicSQuAD, ARCD includes 1395 questions based on Wikipedia articles. We use 50% of ARCD for testing and the entirety of Arabic-SQuAD for training.Baselines Multilingual BERT had previously achieved state of the art results on ARCD.

EXPERIMENTS

Experimental Setup

Regarding We employed the original TensorFlow implementation of BERT in our research. After being split up into TFRecords and sharded, the pre-training data was put on Google Cloud Storage. The masking probability was set at 15%, the duplication factor to 10, and a random seed of 34. The model underwent 1,250,000 steps of pre-training on a TPUv2-8 pod. In order to reduce the training time, sequences of 128 tokens were used to train the first 900K steps, then sequences of 512 tokens were used to train the remaining steps. The completion of downstream tasks determined when to end the pre-training. We adopt the same methodology as the German BERT open-source project (DeepsetAI, 2019). The Adam optimizer was utilized, with a batch size of 512 and 128 for sequence lengths of 128 and 512, respectively, and a learning rate of 1e-4. Training covered all tokens throughout 27 epochs in 4 days. Adjusting For each job, independent fine-tuning was carried out using the same setup. Because of time and computing limitations, we do not do a thorough grid search to find the ideal hyper-parameters. When available, we make use of the splits that the authors of the dataset provide. and, when not, the typical 80% and 20%.

Results

Table 1 compares state-of-the-art findings and the multilingual BERT model (mBERT) with the experimental results of using AraBERT to numerous Arabic NLU downstream tasks.

Sentiment Analysis Table 1's results for Arabic sentiment analysis demonstrate that both AraBERT variants perform better than mBERT and other cutting-edge methods on the majority of evaluated datasets. Despite having been trained on MSA, AraBERT performed admirably on dialects that had never been encountered before.

Table 1: AraBERT's performance on downstream Arabic tasks in comparison to mBERT and earlier state-of-the-art systems

Task	Metric	prev. SOTA	mBERT	AraBERTv0.1/ v1
SA (HARD)	Acc.	95.7*	95.7	96.2 / 96.1
SA (ASTD)	Acc.	86.5*	80.1	92.2 / 92.6
SA (ArsenTD-Lev)	Acc.	52.4*	51	58.9 / 59.4
SA (AJGT)	Acc.	92.6**	83.6	93.1 / 93.8
SA (LABR)	Acc.	87.5†	83	85.9 / 86.7
NER (ANERcorp)	macro-F1	81.7††	78.4	84.2 / 81.9
QA (ARCD)	Exact Match	Mbert	34.2	30.1 / 30.6
	macro-F1		61.3	61.2 / 62.7
	Sent. Match		90	93.0 / 92.0

Previous generation of the BiLSTM-CRF model's cutting-edge output

Named Entity Recognition Results in Table 1 show that AraBERTv0.1 achieved an F1 score of 84.2, 2.53 points higher than the Bi-LSTM-CRF model, making AraBERT the new state-of-the-art for NER on ANERcorp. Tokenized prefixes and suffixes in AraBERT testing produced outcomes resembling those of the Bi-LSTM-CRF model. We think that this occurred because the start token (B-label) is frequently linked to the suffixes. As an illustration, the phrase "الجامعة" with the label B-ORG is changed to "جامعة" ,"ال", and so on with the labels B-ORG and I-ORG, respectively, giving the model false initial signals. The results of our multilingual BERT testing

were not as good as those of the baseline model, hence it was useless.Responding to Inquiries Although Table 1's data indicate an increase in the F1-score, the exact match scores were much lower. After looking over the results again, I found that most of the incorrect responses were only one or two words off from the correct answer, with no discernible difference in the answer's semantics. Tables 2 and 3 provide examples. Additionally, we report a 2% absolute improvement in sentence match score over the prior state-of-the-art, mBERT. Sentence Match (SM) calculates the proportion of predictions that match the ground truth response in the same sentence.

To develop a BiLSTM-CRF model for analyzing sentiment in Arabic social discourse, we can follow the following steps. Prepare the data, build the model, train it, evaluate it, and display the results. By using the BiLSTM-CRF model, which is considered one of the effective models in natural language processing. As in the following flowchart:

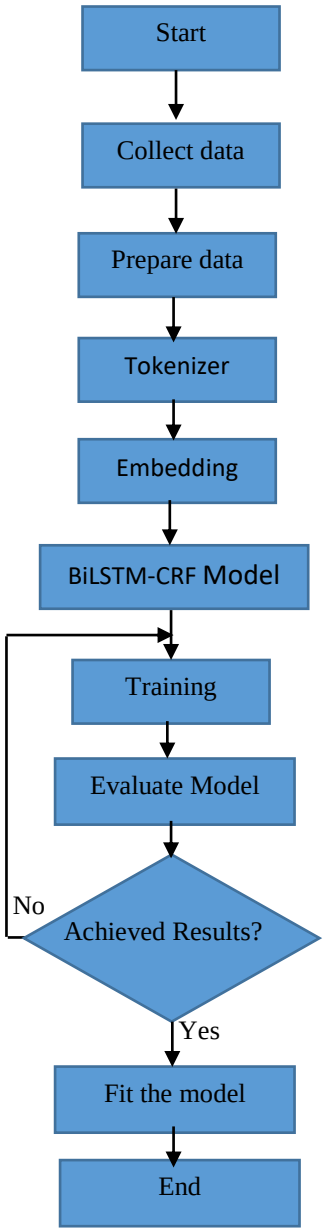

Table 2: An illustration of an incorrect result from the ARCD test set is that the preposition is the only thing that differs : "في ـ - In".

Question	أين تاسست منظمة الامم المتحدة؟ **where was the united nations established**?
Ground Truth Prdicted Answer	In San Francisco- **في سان فرانسيسكو** San Francisco – **سان فرانسيسكو**

Table 3: An additional illustration of a false result from the ARCD test set is when the anticipated response omits "introductory" terms.

Question	ماهو النظام الخاص بدولة استراليا؟ **What is the type of government ?in Austria**
Ground Truth Prdicted Answer	Austria is a federal republic- **استراليا جمهورية فيدرالية** A federal republic – **جمهورية فيدرالية**

DISCUSSION

In the tasks of question answering, named entity identification, and sentiment analysis, AraBERT demonstrated cutting-edge performance. This strengthens the theory that pretrained language models on a single language only outperform multilingual models in terms of performance. There are several reasons for this performance increase. First, the increase in performance may be clearly attributed to data size. Compared to the 4.3G Wikipedia used for the multilingual BERT, AraBERT consumed about 24GB of data. Secondly, the vocabulary size utilized in the multilingual BERT is 2k tokens, whereas 64k tokens were utilized in the development of AraBERT. Third, there is greater variation in the pre-training distribution due to the huge sample quantity.Regarding the fourth point, performance was lowered on the NER job but increased on the SA and QA tasks when pre-segmentation was implemented prior to BERT tokenization. It should be mentioned that the pre-processing done on the pre-training data takes Arabic language complexity into account. Therefore, by eliminating superfluous, redundant tokens that accompany certain common prefixes, the effective vocabulary was raised. Additionally, by lowering the language complexity, the model was able to learn more effectively. We think these elements contributed to achieving state-of-the-art performance on eight distinct datasets and three distinct challenges. The obtained findings show that a monolingual model, as opposed to a broad language model trained on Wikipedia crawls, like multilingual BERT, better understands the advantage we obtained in the datasets analyzed.

CONCLUSION

AraBERT raises the bar for Arabic language downstream work in a number of areas. Additionally, it is 300 MB less than multilingual BERT. We expect that by making our AraBERT models available to the public, they will be able to be utilized as the new standard for a variety of Arabic natural language processing tasks. Additionally, we hope that our work will serve as a foundation for developing and enhancing future Arabic language understanding models. Our current project is to release a version of AraBERT that is independent of third-party tokenizers. Additionally, we are now training models with a deeper comprehension of the many dialects that exist within the Arabic language across various Arabic-speaking nations.

REFERENCES

1. Darwish, K., Mubarak, H., Durrani, N., and Abdulali, A. (2016). Farasa: An ardent and swift Arabic segmenter. Pages 11–16 in Proceedings of the 2016 Conference of the Association for Computational Linguistics' North American Chapter: Demonstrations.
2. Elaraby, M., Alhuzali, H., and Abdul-Mageed, M. (2018). You tweet what you say: An Arabic dialect dataset at the city level. The Eleventh International Conference on Language Resources and Evaluation (LREC 2018) proceedings
3. Magdy, W., and Abu Farha, I. (2019). Mazajak: An Arabic emotion analyzer available online. In August, Florence, Italy, Proceedings of the Fourth Arabic Natural Language Processing Workshop, pp 192–198. The Computational Linguistics Association.
4. Yang, Z., Kulshreshtha, A., Nemade, G., Lu, Y., Hall, J., Fiedel, N., Thoppilan, R., Adiwardana, D., Luong, M.-T., So, D. R., and Le, Q. V. (2020). Toward an open-domain chatbot that resembles a human.
5. Hajj, H., Badaro, G., Baly, R., El-Hajj, W., Al Sallab, A., and Shaban, K. (2015). Arabesque sentiment analysis using deep learning models. Pages 9–17 in the Proceedings of the Second Workshop on Arabic Natural Language Processing.
6. El-Hajj, W., Badaro, G., Baly, R., Hajj, H., Shaban, K. B., and Al-Sallab, A. (2017). Aroma: A low-resource language recursive deep learning model for opinion mining in Arabic. ACM Transactions on Low-Resource and Asian Language Information Processing 16(4):1–20.
7. Shaalan, K., ElSherif, H. M., and Alomari, K. M. (2017). Machine learning sentiment analysis of Arabic tweets. Applied Intelligent Systems, International Conference on Industrial, and Engineering, and Other Applications, pages 602–610. Springer.
8. Atiya, A. and M. Aly (2013). LABR: A comprehensive dataset of Arabic book reviews. In Volume 2: Short Papers, Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, August, Sofia, Bulgaria, pp 494–498. The Computational Linguistics Association.
9. El-Hajj, W., Habash, N., Hajj, H., Baly, R., and Badaro, G. (2014). a comprehensive Arabic emotion vocabulary for opinion mining in Arabic. In Arabic natural language processing (ANLP) workshop proceedings, EMNLP 2014, pages 165–173.
10. Shaban, K. B., El-Hajj, W., Habash, N., Hajj, H., and Baly, R. (2017). For efficient sentiment analysis in Arabic, morphologically enhanced recursive deep models and a sentiment treebank are used. 16(4):1–21 in ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP).
11. Baly R., Khaddaj A., Hajj H., El-Hajj W., and Shaban K. B. (2018). Arsentd-lev is a multi-topic corpus for target-based sentiment analysis in Arabic-Levantine tweets. OSACT 3: The Third Workshop on Open Source Arabic Corpora and Processing Tools, p. 37.
12. Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzman, F., Grave, E., Ott, M., Zettle-Moyer, L., and Stoyanov, V. (2019). Unsupervised cross-lingual representation learning at scale. J. D. Lafferty, A. McCallum, and F. C. Pereira (2001).
13. Conditional random fields are probabilistic models for segmenting and labeling sequence data. In ICML. Martin L., Muller B., Suarez P. J. O., Dupont Y., Romary L., Eric Villemonte de la Clergerie, Seddah D., and Sagot B. (2019). Camembert, a delectable French language model.
14. Zeroual, I., D. Goldhahn, T. Eckart, and A. Lakhouaja (2019). OSIAN: Open source international Arabic news corpus created and integrated into the CLARIN infrastructure. In the Proceedings of the Fourth Arabic Natural Language Processing Workshop, pp 175-182, Florence, Italy, August. Association of Computational Linguistics.

15. Erdmann A., Zalmout N., & Habash N. (2018). Addressing noise in multilingual word embeddings. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 558-565.
16. Martin L., Muller B., Suarez P. J. O., Dupont Y., Romary L., Eric Villemonte de la Clergerie, Seddah D., and Sagot B. (2019). Camembert, a delectable French language model.
17. Kodo, T. (2018). Subword regularization is the process of improving neural network translation models by introducing several subword possibilities.
18. Rajpurkar P., Zhang J., Lopyrev K., & Liang P. (2016).Squad: Over 100,000 questions for machine understanding of text. arXiv preprint: 1606.05250.
19. Polignano, M., Basile, P., de Gemmis, M., Semeraro, G.,and Basile, V. (2019). AlBERTo: Italian BERT Language Understanding Model for NLP Challenging Tasks Based on Tweets. In Proceedings of the Sixth Italian Conference on Computati