Using Bidirectional Long Short-Term Memory and Conditional Random Fields for Labeling Arabic Named Entities: A Comparative Study

Sa'aD A. AlzbouN
saalzboon16@cit.just.edu.jo

Saja Khaled Tawalbeh
sajatawalbeh91@gmail.com

Mohammad AL-Smadi
masmadi@just.edu.jo

Yaser Jararweh
yijararweh@just.edu.jo

Dept. of Computer Science, Jordan University of Science and Technology - Irbid, Jordan

***Abstract*—Named entity recognition (NER) is considered as one of the important tasks of natural languages processing (NLP). This paper presents two approaches that were developed for Arabic named entity recognition (ANER). The first approach is based on a traditional machine learning method of using the conditional random fields (CRF) trained with predefined set of syntactic and morphological features. Whereas, the second approach is based on the bidirectional long short-term memory with a conditional random fields layer (Bi-LSTM-CRF). Both approaches were evaluated using a reference dataset for ANER. Evaluation results show that the Bi-LSTM-CRF deep neural network overcomes the traditional CRF model with 15% of enhancement based on the F1 measure.**

***Index Terms*—Arabic Named-Entity Recognition, Machine learning, Deep neural network, LSTM, CRF, Bidirectional LSTM.**

I. Introduction

Named entity recognition (NER) is a natural languages processing task that labels or classifies entities in a given text according to predefined labels or tags (such as person, location, organization, etc.). NER is a very important task in the field of natural languages processing (NLP) where many applications require semantic labeling techniques and NER as part of their methodology. For instance, NER is used in many NLP and semantic computing applications such as: sentiment analysis and aspect-based sentiment analysis [1], affective news analysis [2], entity linking and disambiguation [3], [4], paraphrase identification and semantic text similarity evaluation [5], machine translation [6], and semantic search and information retrieval [7].

Majority of the previous research used traditional supervised machine learning for NER. Examples include: using models of conditional random fields (CRF) [8], support vector machine (SVM) trained on predefined features [4], [9], [10], and using neural networks such as in [11]. A neural network model is considered to be less depending on features engineering but rely on using word embedding only to learn the text and to train the model using a large amount of tagged data. However, neural networks have some limitations such as relying on simple feed forward for neurons learning, and depending on word embedding only to learn the text features.

In this paper, we propose two models for ANER out of Modern Standard Arabic (MSA) dataset. (i) The first model is a baseline approach where we applied traditional machine learning by using the conditional random fields (CRF) [8] classifier trained on morphological and syntactic features, (ii) whereas, in the second model we applied a deep neural network by using a Bidirectional Long Short-Term Memory with CRF (BI-LSTM-CRF) model [12]. Research contributions are summarized as follows: (a) propose a deep learning based NER for Arabic text in comparison with traditional supervised ML techniques represented by the baseline CRF model. (b) compare the performance of the baseline CRF model with the Bi-LSTM-CRF model evaluated using the same reference dataset.

Both proposed models were evaluated using the WikiFANE Gold 2014 dataset [13]. Evaluation results show that the Bi-LSTM-CRF model outperforms the baseline model on all the targeted categories i.e. PERSON, LOCATION, and ORGANIZATION.

The remainder of this paper is present as follows: Section 2 sheds the light on related work, Section 3 describes the named entity recognition models used in this research, Section 4 discusses the experimentation setup and evaluation results, whereas Section 5 concludes this research and provides pos-

978-1-5386-9588-3/18/$31.00 ©2018 IEEE

sibilities for future work.

II. Related Work

Reference [12] presents a sequence tagging study using a variety of LSTM based models, the sequence tagging problems presented in this paper are named entity recognition (NER), part of speech tagging (POS), and chunking. Focusing on NER. The research implemented a Bi-LSTM-CRF model for NER trained with external lexicons for word embedding and additional word features. The model was evaluated using the CoNLL 2003 dataset [14] and achieved promising results with accuracy= 84.26% with only word features and 90.10% using external Senna lexicon and Gazetteer in addition to the word features.

References [15], [16] provide a comparison study for the impact of using a various set of features (syntactic, morphological, lexical and contextual features) for training two types of machine learning approaches namely, Conditional Random Fields (CRF) and Support Vector Machines (SVM) for Arabic NER. The approaches were evaluated using the ACE 2003 Broadcast News data and achieved good results.

Reference [17] presents a NER approach applied on Twitter messages through using bidirectional long Short-Term Memory (Bi-LSTM). Proposed model enabled an automatic training using orthographic features without the need for hand-crafted features (feature engineering). The aim of this research is to tackle the problem of having noisy and slang tweet messages.

Reference [18] proposes a novel hybrid approach using bidirectional LSTM (BI-LSTM) networks architecture and Conventional Neural Networks (CNN) architecture for NER. Proposed models were trained using word-Level and character-Level features as well as using external lexicons. Proposed approaches were evaluated using the CoNLL-2003 [14] and OntoNotes5.0 [19] datasets and achieved F1 score 91.62% and 86.28% for the CoNLL-2003 and OntoNotes5.0 datasets, respectively.

Reference [20] presents a novel approach for NER problems based on Gated Convolutional Neural Networks (GCNN). Comparing the proposed model GCNN for NER with NER models that depends on RNN, the GCNN provides better results and achieves enhanced performance in the effectiveness of training. GCNN model was evaluated using three datasets: (i) MSRA [21], (ii) CityU [21], and (iii) CoNLL-2003 [14]. The GCNN achieved F1 score of 91.23%, 90.65%, and 91.24% for the MSRA, CityU, and CoNLL-2003 datasets respectively.

Reference [22] proposes language independent NER system depending on deep neural networks (CharWNN) which is based on word and character-level embedding in addition to pre-trained word embedding. Proposed model was evaluated using the datasets HAREM I [23] for Portuguese language and SPA CoNLL2002 [24] for Spanish language and achieved results of F1= 82.21% for SPA CoNLL-2002, and 71.23% for HAREM I.

Reference [25] presents two approaches for NER, the first one is based on bidirectional LSTM with CRF (Bi-LSTM-CRF), whereas the second one labels segments of the input text using a transition-based approach. The models were trained with character and word level representations that capture morphological and orthographic features of the input text. The models are designed to be language independent and was evaluated using datasets with different languages (i.e. English, German, Spanish, and Dutch), CoNLL-2002 [24] and CoNLL-2003 [14] and achieved what the authors claimed to be the highest reported results for NER models evaluated using the same dataset.

Reference [26] presents a deep neural network for NER of Modern Standard Arabic and Egyptian dialectal Arabic tweets. The implemented model is based on a deep neural network with word embedding and character representations using CNN as an input used to train a Bi-LSTM-CRF model. The model was trained to label nine categories of entities namely: PERSON, ORGANIZATION, LOCATION, PRODUCT, TITLE, GROUP, EVENT, TIME, and OTHER. The proposed approach was evaluated with dataset of tweets and achieved F1 score of 70.09% for the overall categories.

Reference [27] proposes a Bi-LSTM-CRF model trained with character-representations and word-representations for ANER out of tweets. The proposed approach were evaluated using a Twitter dataset [28] and achieved results of 85.71% for the F1 score.

Reference [29] Proposed a deep active learning approach for NER based on CNN-CNN-LSTM. The CNN layers were used to encode the input text based on character and word levels, whereas the LSTM layer was used as a decoder to label the NER tags. The approach was evaluated using English and Chinese NER datasets: CoNLL-2003 English [14] and OntoNotes5.0 [19]. As reported in the paper, the active deep learning contribution enabled the implemented models to achieve state-of-the-art results with less input data.

Based on related work, it is found that most of the developed approaches for NER are either using traditional machine learning with selected predefined features (e.g. using CRF or SVM) or using a deep neural network with adapted versions of LSTM and CNN. Therefore, this paper presents a comparative study of both directions. More specifically, a bidirectional LSTM with a layer of CRF is compared to a traditional CRF with predefined syntactic and morphological features.

III. Proposed Models for ANER

Following the work presented in [13], [26], [27], this paper discusses the outcome of using (i) CRF with predefined features and (ii) Bi-LSTM-CRF with word embedding models for ANER.

A. The CRF Model

Conditional Random Fields (CRF) is an undirected graph model in which each word in the input sentence is represented using a corresponding tag in the output sentence [30]. The CRF model depends on the neighboring words/tags to predict the tag of the current word, the prediction of the tag can be done using a sentence-level representation as a whole instead of the individual word representation [31].

The CRF model was trained to label three categories: Person (PER), Location (LOC), Organization (ORG), and for other categories, the Object (O) class was used. Fig. 1. illustrates an Arabic named entity recognition using the CRF model. The IOB encoding is used to represent the NER tags where B-tag indicates the beginning of an entity and I-tag indicates an intermediate entity and the O-tag represents the other non entities in the sequence (i.e. objects).

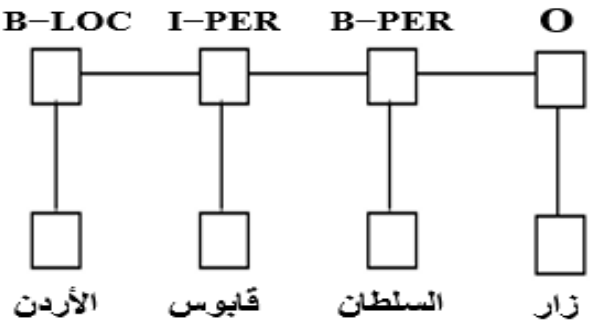

Fig. 1. An example of using the CRF Model for ANER.

In total 21 fine-grained entities were used to represent the three main categories of entities (i.e. PER, LOC, and ORG), with 36207 instances. See TABLE I for the detailed representation of entities.

TABLE I
THE FINE-GRAINED CLASSES FOR ARABIC NER USED TO TRAIN OUR MODELS

Main class	Sub-class
Person (PER)	Athlete (1293)
	Politician (6453)
	Scientist (2178)
	Businessperson (864)
	Artist (2514)
	Religious-PER (2710)
	Police (1324)
	Group (3565)
	Engineer (259)
Organization (ORG)	Government (1785)
	Non-Governmental (2014)
	Educational (1250)
	Media (1194)
	Commercial (2182)
	Sports (1134)
	Religious-ORG (194)
	Entertainment (160)
	Medical-Science (185)
Location (LOC)	Water-Body (2928)
	Land-Region-Natural (954)
	Celestial (1067)
Total (3)	21 (36207)

B. Bi-LSTM-CRF Model

In sequence labeling tasks such as named entity recognition, the accuracy of the recognition can be enhanced using the sequence around the word being predicted. Therefore, using the Bidirectional Long Short-Term Memory (Bi-LSTM) model

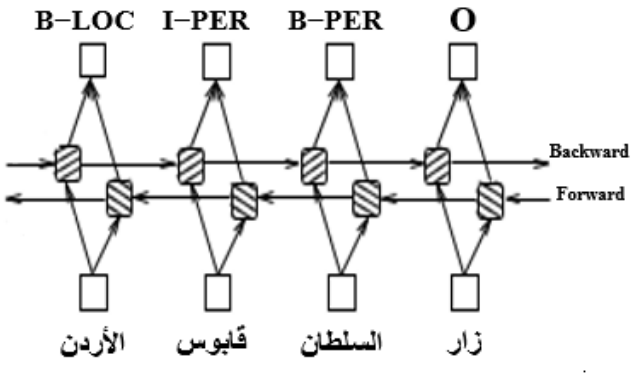

Fig. 2. An example of using the Bi-LSTM Model for ANER.

[12], [25] enhances the recognition accuracy. The Bi-LSTM model is capable to learn form past and futuristic input features at a specific period of time (i.e. a window approach). The Bi-LSTM model can learn the past input features through the forward pass state and the futuristic input features through the backward pass state. The back-propagation through time (PPTT) algorithm was used in the model training [32].

Fig. 2. illustrates how an Arabic sequence is tagged for ANER using a Bi-LSTM model. Each word of the sentence is tagged with one of the three entities mentioned earlier or with 'O' for non-entity recognition.

Following the findings from literature, the Bi-LSTM model can be combined with a CRF layer to enhance the model results [12], [25]. This model inherits the ability of learning past and futuristic input features from the Bi-LSTM model and uses a sentence-level tag to predict the possible tags with the help of the CRF layer. The same three categories of entities (see Table I.) were used to train the Bi-LSTM-CRF model. Fig. 3. illustrates an example of using Bi-LSTM-CRF model for Arabic named entity recognition.

The Bi-LSTM-CRF model was trained based on the pseudo code presented in Algorithm 1. As presented in the Algorithm 1, for each epoch out of 10 epochs, the data was processed in batches and for each batch out of 100, the four steps for Bi-LSTM-CRF are as follows: (1) the Bi-LSTM forward pass, (2) the Bi-LSTM backward pass, (3) the CRF forward and backward passes, then (4) the parameters for the Bi-LSTM-CRF are updated.

IV. EXPERIMENTATION AND SETUP

A. ANER Dataset

A reference dataset based on the work of [13] was used to evaluate our developed models. Out of the dataset categories only three were selected (i.e. PER, LOC, ORG). The dataset is annotated with fine-grained subcategories as presented in Table I. Regarding the training, validation, and testing datasets,

Algorithm 1 The Bi-LSTM-CRF model training algorithm.

```
for each epoch do
    for each batch do
        LSTM model forward pass
        LSTM model backward pass
        CRF layer forward and backward pass
        update parameters
    end for
end for
```

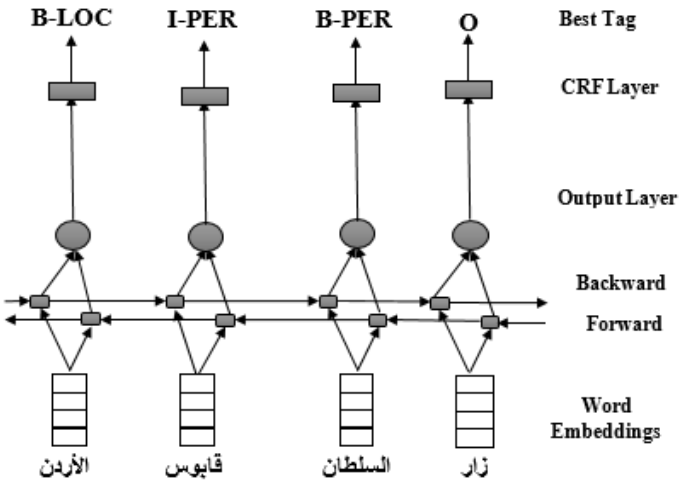

Fig. 3. An example of using the Bi-LSTM-CRF model for ANER.

Table II and Table III show the size of them represented by the number of sentences, tokens, and labels.

TABLE II
CRF MODEL TRAINING AND TESTING DATA

WikiFANE Gold 2014 (500k)		
training	sentence #	12565
	token #	400343
testing	sentence #	3131
	token #	98219
	label #	21

TABLE III
BI-LSTM-CRF MODEL TRAINING, VALIDATION AND TESTING DATA

WikiFANE Gold 2014 (500k)		
training	sentence #	12000
	token #	313290
validation	sentence #	2565
	token #	87053
testing	sentence#	3131
	token #	98219
	label #	21

B. Data Preprocessing

In order to achieve high accuracy results, the dataset was cleaned and tokenized. Content such as symbols (. , !), special characters, replication letters, non-Arabic words and non-Arabic letters, were removed.

The MADAMIRA tool was used for data tokenization [33]. Both implemented models were trained and tested using the same cleaned and tokenized dataset.

C. Features Extraction

In order to train and test the CRF model a set of syntactic and morphological features were extracted. MADAMIRA tool [33] was used for feature extraction. MADAMIRA is a morphological and syntactic analysis tool for Arabic language developed based on the tools MADA [34] and AMIRA [35].

For the syntactic features the part-of-speech (POS) tag was extracted. Whereas, for the morphological features a set of features based on number, voice, and gender were extracted. Table IV summarizes the features that were extracted out of the dataset.

TABLE IV
EXTRACTED FEATURES FROM MADAMIRA TOOL

	MADAMIRA Feature	
Part-of-Speech Tagging (POS)	syntactic	Noun Verb Adj ... etc
Number (num)	morphological	Singular (S) Plural (P) Dual (d) Not applicable (na) Undefined (u)
Person (per)	morphological	1st (1) 2nd (2) 3rd (3) Not applicable (na)
Voice (vox)	morphological	Active (a) Passive (p) Not applicable (na) Undefined (u)
Gender (gen)	morphological	Feminine (f) Masculine (m) Not applicable (na)

D. Word Embedding

As discussed in [11], word embedding plays an important role in enhancing classification results when it comes to using deep neural networks (DNN). In our proposed model (Bi-LSTM-CRF), the word2vec algorithm [36] based on the Keras implementation [37] was implemented with a 50-dimensional embedding vector to represent the input words.

E. Models Implementation and Training

The CRFSuite tool [38] was used to implement the CRF model. As presented in Table V, the Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) method was applied as an optimizer, coefficient for L1 regularization (C1) is set to 0.1, coefficient for L2 regularization (C2) is set to 0.1, Maximum number of iterations were 25000.

The Keras library from TensorFlow [37] was used to implement the Bi-LSTM-CRF model where, the RMSprop was

implemented as an optimizer, the learning rate was set to 0.1, 300 hidden layers were used, the batch size was set to 100, and 10 Epochs were trained (see Table V).

TABLE V
THE PARAMETER USED FOR TRAINING THE TWO MODELS.

	Models Parameter	
CRF Model	Optimizer	lbfgs
	C1	0.1
	C2	0.1
	Max iterations	25000
BI-LSTM-CRF Model	Optimizer	RMSprop
	Learning rate (Lr)	0.1
	Hidden layer	300
	Epoch	10
	Batch size	100

The implemented models (i.e. CRF and Bi-LSTM-CRF) were trained using the same dataset [13], with the same size of training and testing data (see the dataset subsection). Tables II and III summarizes the size of the training, validation and testing datasets.

V. Results and Findings

A. Evaluation Measure

In order to evaluate the performance of our models (i.e. CRF and Bi-LSTM-CRF), the evaluation measures precision, recall, and F1-Score were computed.

The precision measure represents the name entities that the system extracted correctly out of the overall extracted named entities, and it can be computed as:

$$Precision = \frac{NumberOfCorrectlyExtractedEntities}{NumberOfNamedEntitiesExtracted}$$

Whereas, the recall represents the name entities that the system extracted correctly out of the overall number of named entities in the dataset, and it can be computed as:

$$Recall = \frac{NumberOfCorrectNamedEntities}{TotalNumberOfNamedEntities}$$

Finally, the F1-measure is computed based on the precision and recall values as follows:

$$F1 - Score = 2 * \frac{(Precision * Recall)}{(Precision + Recall)}$$

B. Results

As presented in Tables VI and VII, achieved results by the proposed approaches are promising. Focusing on the F1 results for the targeted named entities (i.e. without the O category results), it can be seen that the Bi-LSTM-CRF outperforms the CRF model in extracting all the categories with an overall F1-score of F1= 73.0% for the Bi-LSTM-CRF model and F1=58.00% for the CRF one. More precisely, an enhancement in F1-score of 15% is achieved by the Bi-LSTM-CRF model over the baseline CRF model.

TABLE VI
THE PERFORMANCE RESULTS OF THE PROPOSED CRF MODEL

	Precision	Recall	F1	Support	Accuracy
PER	62.75%	52.50%	57.16%	4008	
ORG	70.34%	39.45%	51.11%	1927	
LOC	74.50%	45.25%	56.35%	956	
Overall	75.00%	48.05%	**58.00**%	6891	65.16%

TABLE VII
THE PERFORMANCE RESULTS OF THE PROPOSED BI-LSTM-CRF MODEL

	Precision	Recall	F1	Support	Accuracy
PER	70.50%	75.05%	72.05%	4008	
ORG	69.34%	77.45%	73.11%	1927	
LOC	71.50%	76.25%	74.35%	956	
Overall	70.00%	76.05%	**73.00**%	6891	75.73%

The results also show that the Bi-LSTM-CRF achieved better results in terms of approach classification accuracy with overall accuracy = 75.73% for the Bi-LSTM-CRF and 65.16% for the CRF one. Findings go inline with literature where in Reference [12] the authors found that Bi-LSTM-CRF outperforms CRF for NER out of the English dataset (i.e. CoNLL2003).

VI. Conclusion and Future Work

In this paper, we evaluated and compared two models for ANER, a traditional machine learning approach based on CRF trained using morphological and syntactic features and a DNN model based on Bi-LSTM-CRF trained with word level representations. Both approaches were evaluated using a reference dataset [13] for fine-grained named entities of the Person, Location and Organization categories (see Table I). Evaluation results show that the Bi-LSTM-CRF model outperforms the traditional CRF model with 15% of enhancement in the F1-score (see Tables VI and VII).

For future work, we plan to investigate the impact of using external lexicons as well as character-level representations for training the DNN model.

References

[1] A.-S. Mohammad, A.-A. Mahmoud, J. Yaser, and Q. Omar, "Enhancing aspect-based sentiment analysis of arabic hotels reviews using morphological, syntactic and semantic features," *Information Processing & Management*, 2018.

[2] M. AL-Smadi, M. Al-Ayyoub, H. Al-Sarhan, and Y. Jararweh, "An aspect-based sentiment analysis approach to evaluating arabic news affect on readers," *Journal of Universal Computer Science (J.UCS)*, vol. 22, pp. 630–649, 2016.

[3] O. Al-Qawasmeh, M. Al-Smadi, and N. Fraihat, "Arabic named entity disambiguation using linked open data," in *2016 7th International Conference on Information and Communication Systems (ICICS)*, April 2016, pp. 333–338.

[4] G. Luo, X. Huang, C.-Y. Lin, and Z. Nie, "Joint entity recognition and disambiguation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 879–888.

[5] M. AL-Smadi, Z. Jaradat, M. AL-Ayyoub, and Y. Jararweh, "Paraphrase identification and semantic text similarity analysis in arabic news tweets using lexical, syntactic, and semantic features," *Information Processing & Management*, vol. 53, no. 3, pp. 640–652, 2017.

[6] H. Toda and R. Kataoka, "A search result clustering method using informatively named entities," in *Proceedings of the 7th annual ACM international workshop on Web information and data management*. ACM, 2005, pp. 81–86.

[7] B. Babych and A. Hartley, "Improving machine translation quality with automatic named entity recognition," in *Proceedings of the 7th International EAMT workshop on MT and other Language Technology Tools, Improving MT through other Language Technology Tools: Resources and Tools for Building MT*. Association for Computational Linguistics, 2003, pp. 1–8.

[8] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," 2001.

[9] L. Ratinov and D. Roth, "Design challenges and misconceptions in named entity recognition," in *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*. Association for Computational Linguistics, 2009, pp. 147–155.

[10] A. Passos, V. Kumar, and A. McCallum, "Lexicon infused phrase embeddings for named entity resolution," *arXiv preprint arXiv:1404.5367*, 2014.

[11] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *Journal of Machine Learning Research*, vol. 12, no. Aug, pp. 2493–2537, 2011.

[12] Z. Huang, W. Xu, and K. Yu, "Bidirectional lstm-crf models for sequence tagging," *arXiv preprint arXiv:1508.01991*, 2015.

[13] F. Alotaibi and M. Lee, "A hybrid approach to features representation for fine-grained arabic named entity recognition," in *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, 2014, pp. 984–995.

[14] E. F. Tjong Kim Sang and F. De Meulder, "Introduction to the conll-2003 shared task: Language-independent named entity recognition," in *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*. Association for Computational Linguistics, 2003, pp. 142–147.

[15] Y. Benajiba, M. Diab, and P. Rosso, "Arabic named entity recognition using optimized feature sets," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2008, pp. 284–293.

[16] Y. Benajiba, I. Zitouni, M. Diab, and P. Rosso, "Arabic named entity recognition: using features extracted from noisy data," in *Proceedings of the ACL 2010 conference short papers*. Association for Computational Linguistics, 2010, pp. 281–285.

[17] N. Limsopatham and N. H. Collier, "Bidirectional lstm for named entity recognition in twitter messages," 2016.

[18] J. P. Chiu and E. Nichols, "Named entity recognition with bidirectional lstm-cnns," *arXiv preprint arXiv:1511.08308*, 2015.

[19] S. Pradhan, A. Moschitti, N. Xue, H. T. Ng, A. Björkelund, O. Uryupina, Y. Zhang, and Z. Zhong, "Towards robust linguistic analysis using ontonotes," in *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, 2013, pp. 143–152.

[20] C. Wang, W. Chen, and B. Xu, "Named entity recognition with gated convolutional neural networks," in *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*. Springer, 2017, pp. 110–121.

[21] G.-A. Levow, "The third international chinese language processing bakeoff: Word segmentation and named entity recognition," in *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, 2006, pp. 108–117.

[22] C. N. d. Santos and V. Guimaraes, "Boosting named entity recognition with neural character embeddings," *arXiv preprint arXiv:1505.05008*, 2015.

[23] D. Santos and N. Cardoso, "Reconhecimento de entidades mencionadas em português," *Linguateca, Portugal*, vol. 7, no. 7, p. 1, 2007.

[24] E. F. Tjong Kim Sang and F. De Meulder, "Introduction to the conll-2003 shared task: Language-independent named entity recognition," in *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*. Association for Computational Linguistics, 2003, pp. 142–147.

[25] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," *arXiv preprint arXiv:1603.01360*, 2016.

[26] M. Attia, Y. Samih, and W. Maier, "Ghht at calcs 2018: Named entity recognition for dialectal arabic using neural networks," in *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, 2018, pp. 98–102.

[27] M. Gridach, "Character-aware neural networks for arabic named entity recognition for social media," in *Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing (WSSANLP2016)*, 2016, pp. 23–32.

[28] K. Darwish, "Named entity recognition using cross-lingual resources: Arabic as an example," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, 2013, pp. 1558–1567.

[29] Y. Shen, H. Yun, Z. C. Lipton, Y. Kronrod, and A. Anandkumar, "Deep active learning for named entity recognition," *arXiv preprint arXiv:1707.05928*, 2017.

[30] A. McCallum and W. Li, "Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons," in *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*. Association for Computational Linguistics, 2003, pp. 188–191.

[31] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," 2001.

[32] M. Boden, "A guide to recurrent neural networks and backpropagation," *the Dallas project*, 2002.

[33] A. Pasha, M. Al-Badrashiny, M. T. Diab, A. El Kholy, R. Eskander, N. Habash, M. Pooleery, O. Rambow, and R. Roth, "Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of arabic." in *LREC*, vol. 14, 2014, pp. 1094–1101.

[34] N. Habash, O. Rambow, and R. Roth, "Mada+ tokan: A toolkit for arabic tokenization, diacritization, morphological disambiguation, pos tagging, stemming and lemmatization," in *Proceedings of the 2nd international conference on Arabic language resources and tools (MEDAR), Cairo, Egypt*, vol. 41, 2009, p. 62.

[35] M. Diab, "Second generation amira tools for arabic processing: Fast and robust tokenization, pos tagging, and base phrase chunking," in *2nd International Conference on Arabic Language Resources and Tools*, vol. 110, 2009.

[36] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[37] F. Chollet *et al.*, "Keras: The python deep learning library," *Astrophysics Source Code Library*, 2018.

[38] N. Okazaki, "Crfsuite: a fast implementation of conditional random fields (crfs)," 2007. [Online]. Available: http://www.chokkan.org/software/crfsuite/