A Semi-Supervised BERT Approach for Arabic Named Entity Recognition

Chadi Helwe[1], **Ghassan Dib**[2], **Mohsen Shamas**[2], **Shady Elbassuoni**[2]
[1]Télécom Paris, Institut Polytechnique de Paris
[2]Department of Computer Science, American University of Beirut
chadi.helwe@telecom-paris.fr
{gid01, mys12, se58}@aub.edu.lb

Abstract

Named entity recognition (NER) plays a significant role in many applications such as information extraction, information retrieval, question answering, and even machine translation. Most of the work on NER using deep learning was done for non-Arabic languages like English and French, and only few studies focused on Arabic. This paper proposes a semi-supervised learning approach to train a BERT-based NER model using labeled and semi-labeled datasets. We compared our approach against various baselines, and state-of-the-art Arabic NER tools on three datasets: AQMAR, NEWS, and TWEETS. We report a significant improvement in F-measure for the AQMAR and the NEWS datasets, which are written in Modern Standard Arabic (MSA), and competitive results for the TWEETS dataset, which contains tweets that are mostly in the Egyptian dialect and contain many mistakes or misspellings.

1 Introduction

In recent years, researchers have become increasingly interested in developing deep learning solutions for Arabic Natural Language Processing applications. Arabic is considered one of the most spoken languages in the world. However, compared to any non-Arabic language such as English, it is considered a much more challenging language because of its high ambiguity and rich morphology.

In this paper, we tackle the problem of Arabic Named Entity Recognition (NER) using a semi-supervised learning approach. NER is the task of extracting, locating, and classifying named entities in a given piece of text. The named entity can be a proper noun, a numerical expression representing type unit or monetary value, or a temporal value that represents time. In this work, we focus on recognizing proper nouns only and classifying them into one of three classes: a person, a location, or an organization in a BIO (beginning, inside, outside) format.

NER is a particularly difficult task for Arabic. First, there is no capitalization in the Arabic script, commonly used in non-Arabic languages such as English to detect named entities. Second, Arabic can be ambiguous; for instance, a lot of named entities are also used as common nouns and adjectives. Arabic is also known for its rich morphology. Finally, one major issue that hinders Arabic NLP research, including NER, is the lack of sufficient resources. Such resources include Arabic corpora and gazetteers that can be leveraged to perform the NLP tasks. Even if some of these resources are present, they are usually limited in scope or not publicly available.

To overcome the aforementioned challenges related to Arabic NLP, we propose a semi-supervised deep learning approach for Arabic NER inspired by the work of Yalniz et al. (Yalniz et al., 2019). The idea is to train two BERT-based models: a teacher model and a student model. BERT stands for Bidirectional Encoder Representations from Transformers. The BERT teacher model is trained on a small labeled data set and then applied on a huge semi-labeled dataset to predict the classes of its unlabeled tokens. The output is then used to train a student model with the same architecture as the teacher model, and then the student model is fine-tuned using the small labeled dataset used to train the teacher model.

To evaluate our approach, we used three different Arabic NER benchmarks, namely AQMAR (Mohit et al., 2012), NEWS (Darwish, 2013) and TWEETS (Darwish, 2013). We compared our approach to various baselines and state-of-the-art NER tools and we outperformed all of them in the case of AQMAR and NEWS datasets and achieved comparable performance in the case of the TWEETs dataset.

The paper is organized as follows. In Section 2, we review the related work. Section 3 describes our semi-supervised learning approach. In Section 4, we evaluate our proposed approach on different datasets. Finally, we conclude and present future directions in Section 5.

2 Related Work

Many approaches have been proposed in the literature to perform Arabic NER. These approaches can be categorized into three main categories: machine-learning-based approaches, rule-based approaches, and hybrid approaches.

In a survey on Arabic NLP (Shaalan, 2014), the authors reviewed a set of machine-learning-based Arabic NER approaches. Some approaches utilized conditional random fields (CRF) (Abdul-Hamid and Darwish, 2010; Benajiba and Rosso, 2007; Benajiba et al., 2007), while others relied on support-vector machines (SVM) (Abdelali et al., 2016; Benajiba et al., 2008b; Koulali and Meziane, 2012; Pasha et al., 2014). Other approaches relied on meta-classifiers (AbdelRahman et al., 2010; Benajiba et al., 2008a; Benajiba et al., 2010). All these approaches utilized different combinations of features such as lexical, contextual, morphological, gazetteer, syntactic and POS features. To date, there are a few works that studied deep learning for the task of Arabic NER. Gridach (Gridach, 2016) utilized character-level neural networks and conditional random fields, in a fully-supervised fashion. However, this approach was trained and tested using only one dataset and was not evaluated on multiple datasets as in our case to assess its generalization capabilities. Helwe and Elbassuoni (Helwe and Elbassuoni, 2019) proposed a semi-supervised learning approach based on an algorithm called co-training, which was adapted to the context of deep learning for the task of Arabic NER. Their method makes use of a small amount of labeled data, which is augmented with partially labeled data that is automatically generated from Wikipedia. Their model is based on an ensemble of two BI-LSTMs. We used the same training, validation, and testing datasets from (Helwe and Elbassuoni, 2019) to evaluate our approach. Antoun et al. (Antoun et al., 2020) pre-trained a BERT model for Arabic called AraBERT, which was evaluated on different tasks such as sentiment analysis and NER. In our approach, we used their pre-trained model and re-trained it in a semi-supervised fashion for the task of Arabic NER.

Many rule-based approaches have been proposed for Arabic NER. Most of these approaches relied on different combinations of features including lexical triggers (Abuleil, 2004; Al-Shalabi et al., 2009), morphological analyzers (Elsebai et al., 2009; Maloney and Niv, 1998; Mesfar, 2007), regular expressions and gazetteers (Shaalan and Raza, 2007), and transliteration (Samy et al., 2005). Most of these reviewed methods however were trained and tested using very limited data, typically less than a hundred documents, thus it is not clear how well they can generalize to other datasets. Moreover, none of these approaches were evaluated on any established benchmarks for the task of Arabic NER. The only exceptions are the approaches by Shaalan and Raza (Shaalan and Raza, 2007), which were trained and tested on the Automatic Content Extraction (ACE) (Doddington et al., 2004) and the Treebank Arabic datasets, and the approach by Elsebai et al. (Elsebai et al., 2009), which was trained and tested using more than 500 news articles.

Another type of approaches commonly used for NER is the hybrid approaches, which combines machine-learning-based and rule-based techniques such as (Abdallah et al., 2012; Oudah and Shaalan, 2012; Shaalan and Raza, 2009). The advantage of our approach over the above mentioned approaches is that we build a more robust machine-learning-based model by training a BERT neural network in a semi-supervised fashion using fully labeled and semi-labeled datasets. We compared our approach to the state-of-the-art approaches (i.e,. those with the highest reported performance from the list above, namely MADAMIRA (Pasha et al., 2014), FARASA (Abdelali et al., 2016) and Deep Co-learning (Helwe and Elbassuoni, 2019)) and outperformed them on two different MSA datasets.

3 Approach

To train a robust model for Arabic NER using deep learning, a sufficiently large training data is needed. Given the lack of such data in the case of Arabic, we propose a semi-supervised learning approach. Our approach is based on a teacher-student learning mechanism inspired by (Yalniz et al., 2019). It relies on

two datasets for training: a fully labeled dataset and a partially labeled dataset. Each instance of these datasets is a sentence composed of word tokens and their labels (person, organization, location or other) if they exist. Figure 1 shows an example instance of the fully labeled dataset. As can be seen from the figure, every token is associated with a label. Figure 2 shows an example instance of the partially labeled dataset, where some of the tokens are labeled and some are not.

The core model in our approach is a pre-trained Arabic BERT model called AraBERT. In brief, our semi-supervised approach works as follows: a BERT teacher model is trained on the fully labeled training dataset to classify the non-labeled tokens of the partially labeled dataset. The best instances from these weakly labeled sentences are then chosen to train a BERT student model, which will be later fine-tuned using the fully labeled training dataset. In the remaining of this section, we first describe the pre-trained AraBERT model. We then describe our proposed semi-supervised learning approach for Arabic NER.

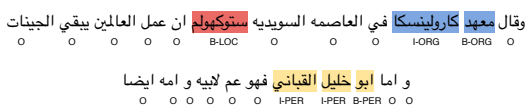

Figure 1: Instance of the Fully Labeled Dataset

Figure 2: Instance of the Partially labeled Dataset

3.1 AraBERT Model

AraBERT is a pretrained Arabic language model developed by Antoun et al. (Antoun et al., 2020) based on a transformer architecture called BERT. The BERT model consists of a stack of transformer blocks which was pre-trained on two tasks: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP).

The task of MLM consists of training the model to predict a masked word given the other words in a sentence. The task's dataset is constructed by choosing 15% of its tokens to be masked by replacing: 80% with the [MASK] token, 10% with a random token, and 10% with the original token. While the task of NSP consists of training the model to learn the relationship between two sentences by taking as input two sentences A and B and predicting if sentence B follows sentence A.

The AraBERT model was pre-trained on a large dataset of 70M Arabic sentences with 3B words. The training data was collected from different publicly available corpora such as the Arabic Wikidumps, the 1.5B words Arabic Corpus, the OSIAN Corpus, and a corpus of Assafir news articles. In addition to the publicly available datasets, the authors augmented the dataset by manually crawling news websites such as Al-Akhbar, Annahar, AL-Ahram, and AL-Wafd.

3.2 Semi-Supervised Learning Model for Arabic NER

Algorithm 1 Semi-Supervised Learning Model for Arabic NER

Require: Labeled Data D^l
Require: Semi Labeled Data D^{sl}
Require: Confidence Threshold τ
Train BERT teacher model $BERT_{teacher}$ with D^l
$pred\_D^{sl} \leftarrow$ Predict the non-labeled tokens from the semi labeled data D^{sl} with $BERT_{teacher}$
for sentence i of $pred\_D^{sl}$ **do**
  $confidence \leftarrow$ Compute the confidence of $pred\_D^{sl}[i]$
  if $confidence >= \tau$ **then**
    $chosen\_D^{sl} \leftarrow$ Save $D^{sl}[i]$
  end if
end for
Train BERT student model $BERT_{student}$ with $chosen\_D^{sl}$
Fine-tune student BERT model $BERT_{student}$ with D^l
return $BERT_{student}$

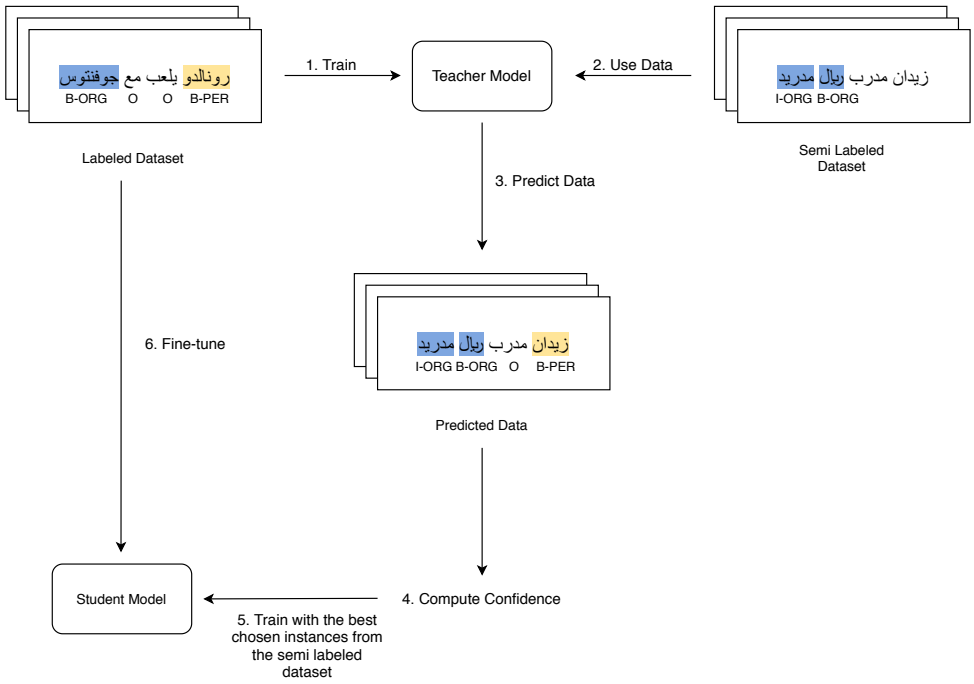

Figure 3: Semi-Supervised Learning Approach

Our semi-supervised learning approach is shown in Figure 3 and summarized in Algorithm 1. In our approach, we make use of two different datasets, one that is fully labeled but limited in size and one that is large but partially labeled by an automatic technique. First, we train a BERT teacher model $BERT_{teacher}$ with the labeled dataset D_l. Second, we predict the labels of the non-labeled tokens of the semi-labeled (i.e., partially labeled) dataset D_{sl} using our trained BERT teacher model $BERT_{teacher}$ and then save them into $pred\_D_{sl}$. Third, we compute the average confidence score of the predicted labels of each instance (sentence) of $pred\_D_{sl}$, and we check if it is higher than a predefined threshold τ. If the condition is met, we pick the instances from $pred\_D_{sl}$ and save them into $chosen\_D_{sl}$. This condition is required to choose the best instances from the data annotated by the teacher model. The average confidence score for each sentence i is computed as follows:

$$s_i = \frac{1}{n} \sum_{j=1}^{n} \underset{0<=l<=6}{\arg\max}\, tok_j^l$$

where n is the number of unlabeled tokens in a sentence i, tok_j^l is the probability of the unlabeled token tok_j in sentence i belonging to label $l \in \{$B-PER, I-PER, B-ORG, I-ORG, B-LOC, I-LOC, O$\}$. That is, the average confidence score s_i is computed using only the non-labeled tokens. For example, in Figure 3, only "Real" and "Madrid" are labeled while the others are not. To label the remaining tokens, we use the teacher model which labels "Zidane" as B-PER and "trainer" as O. The newly labeled tokens are used to compute the average confidence score to check if this instance should be added or not into the chosen dataset $chosen\_D_{sl}$. We then train another BERT model $BERT_{student}$, called the student model that has the same architecture of the teacher model, with the chosen instances $chosen\_D_{sl}$. Finally, we fine-tune the student model using the labeled dataset D_l.

4 Evaluation

4.1 Datasets

In this paper, the datasets used for training, validation, and testing are the same as those used by Helwe and Elbassuoni (Helwe and Elbassuoni, 2019). We adopted six different datasets such that one set is used for training, one set is used for validation, three sets are used for testing and one set which is partially

annotated is used to train our semi-supervised model described in the previous section. First, there is the training dataset (ANERCorp dataset (ANE, 2007)), which consists of 114,926 labeled tokens (about 10,880 articles), and it was used to train the teacher model and fine-tune the student model. Then there is the validation data, the NewsFANE_Gold corpus (Alotaibi and Lee, 2014), which consists of 71,067 labeled sentences (about 1,360 articles), and we used it for validation to fine-tune the hyperparameters of the model. Our approach was then tested on three different Arabic NER benchmarks. The first dataset we evaluated our model on is the AQMAR dataset, an annotated corpus for the task of ArabicNER. AQMAR by Mohit et al. (Mohit et al., 2012) consists of 2,456 sentences from 28 articles from Arabic Wikipedia. The articles belong to four domains, particularly history, science, sports, and technology. The second dataset is the NEWS dataset, which is also an annotated corpus for the task of Arabic NER constructed by Darwish (Darwish, 2013). The NEWS dataset consists of 292 sentences retrieved from the RSS feed of the Arabic (Egypt) version of news.google.com from October 6, 2012. The corpus contains news from different sources and covers international and local news related to politics, finance, health, sports, entertainment, and technology. The third and final dataset we used for evaluation is the TWEETS dataset, also constructed by Darwish (Darwish, 2013). The TWEETS dataset consists of 982 tweets randomly selected from tweets posted between November 23, 2011 and November 27, 2011. The tweets were retrieved from Twitter API using the query lang: ar (language=Arabic). Finally, we used a semi-labeled dataset in order to train our semi-supervised model. The semi-labeled data consists of 1,617,184 labeled and unlabeled tokens. Each line contains a set of tokens and their labels if they exist. This dataset was automatically generated by annotating all the entities in randomly selected Wikipedia articles using an LSTM neural network model (Helwe and Elbassuoni, 2019). This model takes as input the summary of the entity's Wikipedia article and classifies it into one of four classes: person, location, organization, or other.

4.2 Experiment

In this section, we evaluate our semi-supervised approach for the task of Arabic NER. We tested our approach described in Section 3 on three different datasets and compared it with various approaches. More precisely, we compared our approach to both FARASA (Abdelali et al., 2016) and MADAMIRA (Pasha et al., 2014), which are well-known Arabic NER tools as based on recent evaluations. In addition, we compared our approach to the deep co-learning approach from (Helwe and Elbassuoni, 2019) and a fully supervised AraBERT model. The fully supervised AraBERT model was trained solely using the ANERCorp dataset and validated using the NewsFANE Gold corpus. This allows us to evaluate the benefit of training an AraBERT model in a semi-supervised fashion using the semi-labeled dataset. The fully supervised AraBERT model was trained for 20 epochs with a batch size of 32, a dropout of 0.2 with early stopping, and we used ADAM as the optimization algorithm. All the hyperparameters were tuned based on the validation set. In order to experiment with our proposed semi-supervised learning BERT approach, we used the fully supervised AraBERT model as the teacher model. We applied the latter model, called the teacher model, on the semi-labeled dataset to predict the non-labeled tokens. We set the threshold τ to a value of 0.95. This threshold is a parameter that was tuned based on the validation set. To choose the instances that satisfy the threshold condition, we computed each instance's average confidence score. We then trained a student model with an architecture similar to the teacher model with the chosen instances of the semi-labeled dataset. Then we fine-tuned the pre-trained student model with the training set, which is the ANERCorp dataset in our case. We realize that fine-tuning the student model on a clean labeled dataset is significant to achieve a better performance after being pre-trained on a large semi-labeled dataset. The training configuration used in the semi-supervised learning approach is similar to the fully supervised AraBERT models' training configuration. All experiments were run on an Ubuntu machine with a 24 GB RAM, a CPU Intel Core I7 and a GPU NVIDIA GeForce GTX 1080 TI 11GB.

4.3 Results

In this section, we evaluate our AraBERT semi-supervised model for the task of Arabic NER. We tested our approach, as mentioned above, on three different datasets and compared the results against different

Arabic NER tools and approaches. To calculate all the F-measures reported in this section, we used the CoNLL evaluation script (Tjong Kim Sang and De Meulder, 2003).

4.3.1 AQMAR Dataset

The first dataset we evaluated our model on is the AQMAR dataset. As can be seen from Table 1, MADAMIRA and FARASA, which are machine learning tools that use feature engineering, have very low F-measure than the deep learning approaches. The Deep Co-learning approach scores a slightly higher F-measure than the AraBERT Fully Supervised since it is a semi-supervised learning method that used the semi-labeled dataset during training. Our approach scores an F-measure of 65.5, which is the highest.

Model	LOC	ORG	PER	Avg
MADAMIRA	39.4	15.1	22.3	29.2
FARASA	60.1	30.6	52.5	52.9
Deep Co-learning	67.0	38.2	65.1	61.8
AraBERT Fully Supervised	63.6	31.0	70.9	61.5
AraBERT Semi-Supervised	68.4	34.6	74.4	**65.5**

Table 1: The F-measure of the various models and the Arabic NER tools on AQMAR

4.3.2 NEWS Dataset

The second dataset is the NEWS dataset. As shown in Table 2, the results of the different approaches and tools are similar to the AQMAR dataset results. The MADAMIRA and FARASA have low scores compared to the deep learning approaches. The Deep Co-learning has a higher F-measure than the AraBERT model trained in a fully supervised fashion, while our approach outperforms all the different approaches and tools, with an F-measure of 78.6.

Model	LOC	ORG	PER	Avg
MADAMIRA	39.4	15.1	22.3	29.2
FARASA	73.1	42.1	69.5	63.9
Deep Co-learning	81.6	52.7	82.4	74.1
AraBERT Fully Supervised	74.2	54.2	85.1	73.2
AraBERT Semi-Supervised	80.5	60.8	89.5	**78.6**

Table 2: The F-measure of the various models and the Arabic NER tools on NEWS

4.3.3 TWEETS Dataset

The third and final dataset we used for evaluation is the TWEETS dataset. As can be seen from Table 3, the MADAMIRA and FARASA tools performed poorly compared to the deep learning approaches with an F-measure of 24.6 and 39.9, respectively. Only in this dataset, the Deep Co-learning approach has the highest score, which is better than the AraBERT trained in a fully supervised fashion and to the AraBERT trained in a semi-supervised fashion with an F-measure of 59.2. The reason behind this result is that the AraBERT model was pre-trained on MSA corpora, which highly differ in nature from tweets that are mostly in the Egyptian dialect and contain mistakes or misspellings.

We conclude that our semi-supervised approach is making a significant improvement in the performance of the Arabic NER task when the texts are written in MSA. To have better results on other types of

Model	LOC	ORG	PER	Avg
MADAMIRA	40.3	8.9	18.4	24.6
FARASA	47.5	24.7	39.8	39.9
Deep Co-learning	65.3	39.7	61.3	**59.2**
AraBERT Fully Supervised	57.9	30.7	60.9	54.0
AraBERT Semi-Supervised	63.3	42.1	59.4	57.3

Table 3: The F-measure of the various models and the Arabic NER tools on TWEETS

Arabic texts like tweets, we need to study the performance of our approach when pre-trained on this type of Arabic texts.

5 Conclusion

This paper presented a new approach to detect and classify named entities in any Arabic text. Our approach consists of training an already pre-trained BERT model for Arabic NER in a semi-supervised fashion. We made use of two datasets. The first dataset was fully labeled, while the second dataset was partially labeled. We evaluated our approach on three datasets. It outperforms all other Arabic NER tools and approaches on two testing datasets, namely NEWS and AQMAR datasets. For the TWEETS dataset, Helwe and Elbassuoni's deep co-learning approach (Helwe and Elbassuoni, 2019) scores a higher F-measure than our method because the BERT model was pre-trained and trained on mainly MSA corpora that do not contain mistakes and misspellings.

In future work, we plan to pre-train the BERT model on tweets to make it more suitable for text that could contain misspellings and mistakes and which is not necessarily written in MSA. We believe that this will result in an improved performance of our approach on the TWEETS datasets. Finally, we plan to apply our semi-supervised BERT-based learning approach to other NLP tasks such as part-of-speech tagging and dependency parsing.

References

Sherief Abdallah, Khaled Shaalan, and Muhammad Shoaib. 2012. Integrating rule-based system with classification for arabic named entity recognition. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 311–322. Springer.

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. Association for Computational Linguistics, San Diego, California.

Samir AbdelRahman, Mohamed Elarnaoty, Marwa Magdy, and Aly Fahmy. 2010. Integrated machine learning techniques for arabic named entity recognition. *IJCSI*, 7:27–36.

Ahmed Abdul-Hamid and Kareem Darwish. 2010. Simplified feature set for arabic named entity recognition. In *Proceedings of the 2010 Named Entities Workshop*, pages 110–115. Association for Computational Linguistics.

Saleem Abuleil. 2004. Extracting names from arabic text for question-answering systems. In *Coupling approaches, coupling media and coupling languages for information retrieval*, pages 638–647. LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE.

Riyad Al-Shalabi, Ghassan Kanaan, Bashar Al-Sarayreh, Khalid Khanfar, Ali Al-Ghonmein, Hamed Talhouni, and Salem Al-Azazmeh. 2009. Proper noun extracting algorithm for arabic language. In *International conference on IT, Thailand*.

Fahd Alotaibi and Mark G Lee. 2014. A hybrid approach to features representation for fine-grained arabic named entity recognition. In *COLING*, pages 984–995.

2007. Anercorp. http://www1.ccls.columbia.edu/ ybenajiba/downloads.html.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. Arabert: Transformer-based model for arabic language understanding. *arXiv preprint arXiv:2003.00104*.

Yassine Benajiba and Paolo Rosso. 2007. Anersys 2.0: Conquering the ner task for the arabic language by combining the maximum entropy with pos-tag information. In *IICAI*, pages 1814–1823.

Yassine Benajiba, Paolo Rosso, and José Miguel Benedíruiz. 2007. Anersys: An arabic named entity recognition system based on maximum entropy. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 143–153. Springer.

Yassine Benajiba, Mona Diab, and Paolo Rosso. 2008a. Arabic named entity recognition using optimized feature sets. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 284–293. Association for Computational Linguistics.

Yassine Benajiba, Mona Diab, Paolo Rosso, et al. 2008b. Arabic named entity recognition: An svm-based approach. In *Proceedings of 2008 Arab International Conference on Information Technology (ACIT)*, pages 16–18.

Yassine Benajiba, Imed Zitouni, Mona Diab, and Paolo Rosso. 2010. Arabic named entity recognition: using features extracted from noisy data. In *Proceedings of the ACL 2010 conference short papers*, pages 281–285. Association for Computational Linguistics.

Kareem Darwish. 2013. Named entity recognition using cross-lingual resources: Arabic as an example. In *ACL (1)*, pages 1558–1567.

George R Doddington, Alexis Mitchell, Mark A Przybocki, Lance A Ramshaw, Stephanie Strassel, and Ralph M Weischedel. 2004. The automatic content extraction (ace) program-tasks, data, and evaluation. In *LREC*, volume 2, page 1.

Ali Elsebai, Farid Meziane, and Fatma Zohra Belkredim. 2009. A rule based persons names arabic extraction system. *Communications of the IBIMA*, 11(6):53–59.

Mourad Gridach. 2016. 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)*, pages 23–32.

Chadi Helwe and Shady Elbassuoni. 2019. Arabic named entity recognition via deep co-learning. *Artificial Intelligence Review*, 52(1):197–215.

Rim Koulali and Abdelouafi Meziane. 2012. A contribution to arabic named entity recognition. In *ICT and Knowledge Engineering (ICT & Knowledge Engineering), 2012 10th International Conference on*, pages 46–52. IEEE.

John Maloney and Michael Niv. 1998. Tagarab: a fast, accurate arabic name recognizer using high-precision morphological analysis. In *Proceedings of the Workshop on Computational Approaches to Semitic Languages*, pages 8–15. Association for Computational Linguistics.

Slim Mesfar. 2007. Named entity recognition for arabic using syntactic grammars. In *Natural Language Processing and Information Systems*, pages 305–316. Springer.

Behrang Mohit, Nathan Schneider, Rishav Bhowmick, Kemal Oflazer, and Noah A Smith. 2012. Recall-oriented learning of named entities in arabic wikipedia. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 162–173. Association for Computational Linguistics.

Mai Oudah and Khaled F Shaalan. 2012. A pipeline arabic named entity recognition using a hybrid approach. In *COLING*, pages 2159–2176.

Arfath Pasha, Mohamed Al-Badrashiny, Mona T Diab, Ahmed El Kholy, Ramy Eskander, Nizar Habash, Manoj Pooleery, Owen Rambow, and Ryan Roth. 2014. Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of arabic. In *LREC*, volume 14, pages 1094–1101.

Doaa Samy, Antonio Moreno, and Jose M Guirao. 2005. A proposal for an arabic named entity tagger leveraging a parallel corpus. In *International Conference RANLP, Borovets, Bulgaria*, pages 459–465.

Khaled Shaalan and Hafsa Raza. 2007. Person name entity recognition for arabic. In *Proceedings of the 2007 Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources*, pages 17–24. Association for Computational Linguistics.

Khaled Shaalan and Hafsa Raza. 2009. Nera: Named entity recognition for arabic. *Journal of the American Society for Information Science and Technology*, 60(8):1652–1663.

Khaled Shaalan. 2014. A survey of arabic named entity recognition and classification. *Computational Linguistics*, 40(2):469–510.

Erik F Tjong Kim Sang and Fien De Meulder. 2003. 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*, pages 142–147. Association for Computational Linguistics.

I Zeki Yalniz, Hervé Jégou, Kan Chen, Manohar Paluri, and Dhruv Mahajan. 2019. Billion-scale semi-supervised learning for image classification. *arXiv preprint arXiv:1905.00546*.