CAraNER: The COVID-19 Arabic Named Entity Corpus

Abdulmohsen Al-Thubaity [1], **Sakhar Alkhereyf** [1], **Wejdan Alzahrani** [2], **Alia Bahanshal** [1]

[1] King Abdulaziz City for Science and Technology, Riyadh, Saudi Arabia
[2] King Saud University, Riyadh, Saudi Arabia
{aalthubaity,salkhereyf,abahanshal}@kacst.edu.sa
444203310@student.ksu.edu.sa

Abstract

Named Entity Recognition (NER) is a well-known problem for the natural language processing (NLP) community. It is a key component of different NLP applications, including information extraction, question answering, and information retrieval. In the literature, there are several Arabic NER datasets with different named entity tags; however, due to data and concept drift, we are always in need of new data for NER and other NLP applications. In this paper, first, we introduce Wassem, a web-based annotation platform for Arabic NLP applications. Wassem can be used to manually annotate textual data for a variety of NLP tasks: text classification, sequence classification, and word segmentation. Second, we introduce the COVID-19 Arabic Named Entities Recognition (CAraNER) dataset extracted from the Arabic Newspaper COVID-19 Corpus (AraNPCC). CAraNER has 55,389 tokens distributed over 1,278 sentences randomly extracted from Saudi Arabian newspaper articles published during 2019, 2020, and 2021. The dataset is labeled by five annotators with five named-entity tags, namely: Person, Title, Location, Organization, and Miscellaneous. The CAraNER corpus is available for download for free. We evaluate the corpus by finetuning four BERT-based Arabic language models on the CAraNER corpus. The best model was AraBERTv0.2-large with 0.86 for the F1 macro measure.

1 Introduction

Named entity recognition (NER) is a classical sequence classification problem where each word in a given sentence is assigned to one of a predefined list of tags such as person name (شاكر Shaker, بايدن Biden), location (القاهرة Cairo, واشنطن Washington), and organization (الأمم المتحدة United Nations, نادي الاتحاد Al-Ittihad Club).

NER is a key component and a fundamental task for many NLP applications, including information extraction (Liu et al., 2021; Nasar et al., 2021), question answering (Xu et al., 2021; Peng et al., 2021), content recommendations (Harrando and Troncy, 2021; Grewal and Lin, 2018), customer support (Brahma et al., 2021; Bozic et al., 2021), and information retrieval (Aliwy et al., 2021). It is one of the earliest tasks of NLP using classical statistical algorithms such as maximum entropy (Chieu and Ng, 2003) and has been developed for many years. However, it is still relevant in the current time where we are using transformer-based language models such as Bidirectional Encoder Representations from Transformers (BERT) (Liu et al., 2022).

Despite the recent advances in the NLP systems due to the usage of deep learning models, especially transformer-based language models, the need for new annotated datasets for developing NER systems is still crucial, where each domain and application requires its own dataset and tags. In general, NER systems, from our perspective, face three challenges:

First, the widespread use of NLP applications in different domains necessitates the usage of texts from these domains, which are probably different in their genre, style, and vocabularies, from the available annotated NER datasets. Out-of-vocabulary (OOV) will be the first challenge the NER system will face. For instance, if we need high-performance NER models, a NER dataset for the legal domain can not be used for the medical domain, and a NER dataset for Moroccan newspapers will not be the best choice for NER applications for UAE newspapers.

Second, unlike fixed tagset applications such as part-of-speech tagging or word segmentation, each NER application requires different tagsets. Most of the NER available datasets concentrate on person names, location, and organization tags with slight differences among them on other tags, such as the availability of geopolitical entities tags for government entities such as the Ontonotes 5 NER

Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP), pages 1 - 10
December 8, 2022 ©2022 Association for Computational Linguistics

dataset (Weischedel et al., 2013). Consider the need for a NER dataset for a food delivery chatbot; in this case, we may need a tagset containing tags for: a) the person's name to know who ordered the food, b) different tags for food items to direct the order to a relevant restaurant, and c) address to know the delivery location. Or suppose a system to analyze newspaper articles for military clashes; such a system, in addition to time, location, and the number of injuries, will need different tags to identify different kinds of weapons, for example.

Third, even if there is a dataset for particular domains or genres, there are always new topics, interests, and concepts introduced to those domains and genres that may degrade the models trained on older datasets. Such a challenge is well known in the machine learning community as data and concept drift (Celik and Vanschoren, 2021; Maheswari et al., 2022; Mei et al., 2022). Consider, for instance, a NER system trained on annotated texts from newspapers during the 2000s and then applied to newspaper texts during the COVID-19 pandemic; will this system perform well?

The contribution of this paper is in three folds. First, we introduce *Wassem* (وَسِّم in Arabic, "annotate" in English), a platform for Arabic textual data annotation based on the Django framework. Second, we used Wassem to prepare COVID-19 Arabic Named Entity Recognition dataset (CAraNER): a NER dataset annotated with six tags (Person, Organization, Location, Title, Miscellaneous, and Other) covering 1,278 sentences, randomly extracted from Saudi Arabian newspapers part of Arabic Newspapers COVID-19 Corpus (AraNPCC) (Al-Thubaity et al., 2022). The COVID-19 part of the corpus name "CAraNER" is a temporal reference to the COVID-19 period as the AraNPCC corpus covers one year before the COVID-19 pandemic and two years after the emergence of the pandemic (i.e. 2019 - 2021).

Third, using CAraNER, we evaluate four BERT-based Arabic language models, namely bert-base-multilingual-cased (Devlin et al., 2019) (baseline), AraBERTv0.2-large (Antoun et al., 2020), CAMeLBERT-MSA (Inoue et al., 2021), and GigaBERT-v4 (Lan et al., 2020).

The rest of this paper is organized as follows: In section 2, we review the related work on Arabic NER. We briefly describe the main building blocks for Wassem in section 3. Section 4 describes the process of the CAraNER dataset construction and annotation and its basic statistics. Section 5 illustrates and discusses the result of fine-tuning four language models using CAraNER. We conclude the paper in section 6.

2 Related Work

Previous works on NER can be divided into two categories: building named-entity-tagged corpora and building NER models. In this section, we focus on previous work on building named-entity tagged Arabic corpora. Previous works on building named-entity corpora cover a variety of languages, genres, and domains. These studies focused on many tag sets that differ according to the application and domain requirements. Some corpora in the literature cover general-purpose tag sets from broad domains such as newswire and Wikipedia. In contrast, others focus on specific tag sets, such as the medical domain. Most previous studies include the four named-entity tags: Person, Location, Organization, and Miscellaneous.

The interest in building Arabic NER corpora dates back to the 2000s. One of the earliest studies for building an Arabic named-entity annotated corpus is the ACE 2004 Multilingual Training Corpus (Mitchell et al., 2005). The ACE 2004 corpus is developed by LDC and contains text in Arabic, Chinese, and English, covering a variety of genres. It was annotated for many NLP tasks, including named entity recognition and relation extraction. The ACE 2004 entity tags are Person (PER), Geo-Political Entity (GPE), Organization (ORG), and Facility (FAC). The size of the Arabic portion of ACE 2004 is around 10K tokens and collected from newswire texts.

Another LDC-licensed multilingual corpus is Ontonotes 5 (Weischedel et al., 2013), which is collected from various genres, including newswire and conversational telephone speech in three languages: Arabic, Chinese, and English. The Arabic portion of Ontonotes 5 contains around 300K tokens. Similar to our corpus, the Arabic Ontonotes 5 corpus was collected only from newswire sources in Modern Standard Arabic (MSA) and annotated with 18 entity types.

The ANERcorp corpus (Benajiba et al., 2007) is collected from Modern Standard Arabic media texts. It contains around 150K tokens tagged with four entity types: person, organization, location, and miscellaneous.

For genres other than newswire, Mohit et al.

(2012) developed the American and Qatari Modeling of Arabic (AQMAR) corpus for Wikipedia articles. It consists of 74K tokens tagged with domain-specific categories covering four topics: technology, science, history, and sports. Salah and Zakaria (2018) developed the Classical Arabic Named Entity Recognition Corpus (CANERCorpus) for text for the Islamic Hadith. It contains around 72K tokens tagged with categories relevant to the field, such as "Prophet".

Darwish and Gao (2014) have developed the first NER dataset for Arabic Tweets. Their dataset comprises 5,069 tweets tagged with three tags, namely: person, location, and organization. Recently, Jarrar et al. (2022) released the Nested Arabic Named Entity Corpus (Wojood). Wojood comprises 550K tokens from Modern Standard Arabic (MSA) and different Arabic dialects. Wojood annotated with 21 entity types, including person, organization, location, product, and unit. Wojood is the largest Arabic NER dataset and the first Arabic NER dataset using nested tagging.

The most important factor that may distinguish the CAraNER dataset is that it was sampled from the COVID-19 period. Regarding the CAraNER size ($\sim$50K tokens), we are working to increase its size to reach a level that can produce good results using state-of-the-art machine learning algorithms, specifically neural language models.

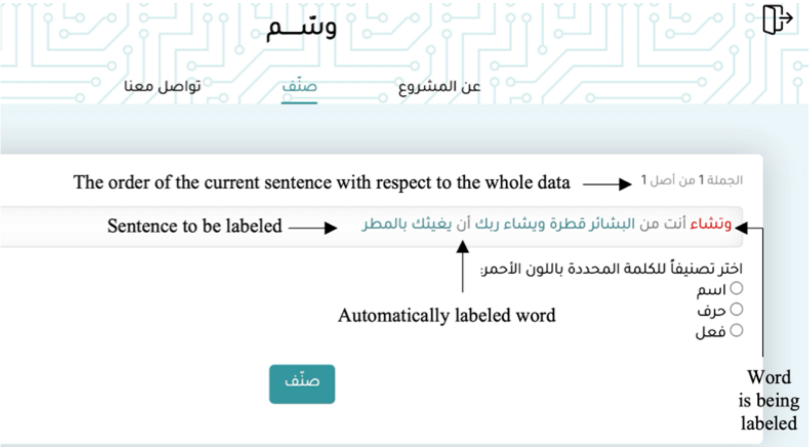
3 Annotation Platform

The motivation behind the development of *Wassem* is the shortage of open source annotation platforms suitable for Arabic NLP annotation tasks such as word segmentation and diacritization. We designed Wassem to help in the following tasks: a) Text and sentence annotation for applications such as text classification and sentiment analysis, b) Sequence annotation for applications like NER and POS tagging, c) Subword annotation for applications like Arabic words segmentation, and d) for Arabic word diacritization.

Wassem has four main functions, which are described as follows:

a. Annotation task initialization: The system administrator is responsible for this function. Four steps are needed to complete this process as follows: First, the administrator needs to define the list of tags used for the annotation task, with a brief description for each tag if they do not exist before in the database. Also, the administrator can attach a list of words with fixed tags such that the corresponding tag for each word in the list does not change when the context changes; this accelerates the annotation process. For example, for POS tagging, this list may include particles and prepositions such as "إلى" (to), "عن" (about), "لكن" (but) or part of the most frequent words in the data that have the same characteristic, i.e., they have fixed tags such as "الله" (Allah), "قال" (Said), "إلى" (To). The system will automatically annotate words with their corresponding tags in the list such that the user does not need to consider them during manual annotation. Such lists of words and their fixed tags can be used in the future for other annotation tasks. Second, the system currently provides manual annotation on the document level and word level. Based on the type of annotation task, the system administrator should determine the level of the task. The difference between the two levels is the text unit that will be annotated with the tag. Hence, in the case of the document level, a label will be assigned to the entire sentence/text, for example, sentiment analysis on the document level. In contrast, for the word-level annotation, each word/token in the text will be labeled with a tag, for example, POS tagging. Moreover, for the case of word-level annotation, the administrator should specify if the task is a segmentation, diacritization, or tagging the whole token (e.g., NER). Third, the administrator needs to provide a description of the annotation tasks and identify the minimum number of annotators who can participate in the task. The system can automatically assign a final label using the majority vote if there are three or more annotators. Fourth, the administrator should upload the raw data that will be tagged if it was not in the system before (i.e., used previously on other annotation tasks), and link this annotation task with the appropriate list of tags.

b. Annotation task assignment: After creating the annotation task, the administrator should assign it to the annotators. The administrator can add new annotators or select annotators already existing in the system. Wassem's website provides a link to a registration form for volunteer annotators where they need to pro-



وسّم
عن المشروع
صنّف
تواصل معنا
The order of the current sentence with respect to the whole data
Sentence to be labeled
Automatically labeled word
Word is being labeled
صنّف

الجملة 1 من أصل 1

وتشاء أنت من البشائر قطرة ويشاء ربك أن يغيثك بالمطر

اختر تصنيفاً للكلمة المحددة باللون الأحمر:

○ اسم
○ حرف
○ فعل

Figure 1: An example of a word-level annotation task (POS tagging) on Wassem.

vide their contact information, gender, age, and educational background. Such information will help the administrator to identify the best annotators for each annotation task.

c. **Annotation process:** When the annotator starts the annotation process for the first time, the system will welcome her/him and provide them with a description of the annotation task and a description for each tag. Then the system will display the data for the annotator to start annotating it. To keep the user concentrated on the annotation task, the system uses three colors for the words in the displayed sentence during annotation:

- Red: highlights the word that is being annotated.
- Gray: indicates the word that was labeled automatically by the system using a predefined list of words and their tags.
- Green: for the rest of the words.

Figure 1 shows an example of word-level annotation for a simple Arabic POS annotation task.

d. **Exporting data:** Finally, the annotated data can be exported in a CSV file format. The system applies the majority voting approach to determine the final tags for each example. If there is no agreement (i.e., tie), the final tag will be set as "*No_agr*". In the case of uncompleted tasks (i.e., some annotators have not completed their tasks yet), the system will set the tag as "*Not_Annot*" to the examples which are not annotated yet.

4 Data

In this section, we describe our work to prepare the raw data for the annotation process, the tagset used for annotation, the annotators' training and annotation process, and the final data after annotation. We made the dataset is available for free download on GitHub [1].

4.1 Data Preparation

The raw CAraNER data is randomly selected sentences from 826,323 Modern Standrad Arabic (MSA) texts that constitute Saudi Arabia newspapers in AraNPCC (Al-Thubaity et al., 2022). AraNPCC comprises more than 1.7 million texts automatically collected from the newspapers of 12 Arab countries for one year before the COVID-19 pandemic and two years during the pandemic (from 1 January 2019 until 31 December 2021). We focus on the Saudi part of the AraNPCC corpus as we had the chance to hire annotators only from Saudi Arabia, who are more familiar with the local named entities such as town names.

To prepare the data for annotation, we followed the following steps:

4.1.1 Texts Selection

There are 8 Saudi newspapers in AraNPCC. The texts from these eight newspapers are categorized into 19 classes: health, corona, culture, economy, international, local, opinion, society, sport, politics, technology, journal, last page, lifestyle, main, religion, story, women, and not classified. Each Saudi newspaper has its own classification system, so not all these classes are available in every Saudi newspaper. For each year (2019, 2020, and 2021) and

[1] https://github.com/kacst-ncdaai/caraner

from each of the eight newspapers, we randomly selected 6 texts from each class. Finally, we got 1,271 texts comprising 322,907 tokens. The number of tokens exceeds our need for this stage of the project; however, it will allow us to extend our work in the future.

4.1.2 Preprocessing

Instead of annotating the entire text, we preprocessed each text and divided it into sentences. Building a NER system based on sentences will allow all ML algorithms to handle the input data easily and will make the data more diverse. For each text, we carried out the following steps to achieve our goal:

a) Replace each new line marker “\n” with a space.

b) Replace each URL with a special marker “<link>”.

c) Remove Arabic diacritics.

d) Replace repeated punctuation marks such as “!”, “?”, “.” and “-” with a single punctuation mark of its kind.

e) Separate punctuation marks from words by a space and parentheses from words and numbers by a space.

f) The above step will affect the dots that come after the title abbreviations of Doctor “د.” (Dr.), Engineer “م.” (Eng.), Professor “أ.” or “ا.” (Prof.), which will negatively affect the process of sentence segmentation. So, we replace each dot after these abbreviations with a special marker “/”.

g) We use the *sent_tokenize(text)* function in the NLTK python package to segment the text into sentences. This step will produce a list of sentences.

h) Select sentences with a length of more than 10 characters.

i) Replace “/” that comes after “د، م، ا، أ” with a dot “.” on the selected sentences and save them in a list. This step allows us to preserve these abbreviations.

Applying the above steps for all texts produced 8,371 sentences comprising 370,138 words. We shuffled these sentences randomly and saved them in 75 text files, each file compromising approximately 5,000 words. Note that the preprocessing steps increase the number of words due to the application of step “e” mentioned above. Dividing the produced sentences into separate files (75) allows managing the annotation process as batches and handle any misconceptions or mistakes by the annotators during the revision of the annotation process for each batch before the beginning of the next batch.

4.2 Tagset

For CAraNER we choose the following tags:

- **PER:** person names such as “محمد” (Mohamad); nicknames such as “أخو نورة” (Nora’s brother) and “الجاحظ” (Al-Jahiz).

- **TIT:** job title such as “رئيس مجلس الوزراء” (Prime Minister); military and civilian ranks “فريق أول بحري” (Admiral); academic or professional title such as “المهندس” (engineer); political or social title such as “صاحب السمو الملكي” (His Royal Highness).

- **LOC:** countries such as “مصر” (Egypt); regions, provinces, cities, and villages such as “بيروت” (Beirut); landmarks and sites such as “غار حراء (Cave of Hira).

- **ORG:** government and commercial organizations and bodies such as “الهيئة السعودية للبيانات والذكاء الاصطناعي” (Saudi Data and Artificial Intelligence Authority); sports clubs such as “فريق نيوكاسل” (Newcastle United Football Club); international bodies such as “المنظمة العربية للتربية والثقافة والعلوم” (Arab League Educational, Cultural and Scientific Organization); countries and capitals as political entities such as “المغرب” (Morocco) in such a following context: “أعربت المغرب عن استنكارها ....” (Morocco has expressed its disapproval ....).

- **MIS:** For other named entities (miscellaneous). It includes but is not limited to diseases such as “كوفيد-١٩” (COVID-19), medicines and chemical compounds such as “كلوروكين” (Chloroquine); events such as “إكسبو ٢٠٢٠” (Expo 2020); Currencies such as “درهم إماراتي (Arab Emirates Dirham); beliefs and ideologies such as

“الديمقراطية” (Democracy); products such as “إيباد برو” (iPad Pro); measurement units such as “كيلو جرام” (Kg); regulations and laws such “قانون كرة اليد الدولي” (international handball federation regulations); tribes such as “تغلب” (Taghlib).

Since the named entities can be in chunks with more than one word, we adopt the most used tagging format for NER: Inside–Outside–Beginning (IOB) format such that tags will be prefixed either with “I” or “B”. The non-named entities will be tagged as “O”.

In addition to these tags, we use the “N” tag to indicate when the annotator can not determine the right tag for a given word. This tag helps us track the annotators’ learning curve and highlight the difficulties they may face during the annotation process. In total, the annotators will work on 12 tags, namely: *B_PER, I_PER, B_TIT, I_TIT, B_LOC, I_LOC, B_ORG, I_ORG, B_MIS, I_MIS, O,* and *N*. The N tag does not appear in the final revised tags for the dataset, as it is revised by other annotators.

4.3 Annotation Process

We have hired five annotators for the annotation process of CAraNER. All annotators are Saudi nationals, two males and three females, in the final semester of their university undergraduate study, and all were around 21 years old.

We followed the following steps to train the annotators:

- We introduced the problem of NER to the annotators.

- We introduced different examples of each tag and discussed them with annotators.

- We asked each annotator, based on their first impression, to annotate three short sentences and ask the other annotators if they agreed or disagreed and why. This discussion allowed us to clarify several issues regarding the annotation process to the annotators.

- We provided the annotators with 25 sentences and asked them to annotate them. Furthermore, we asked the annotators not to discuss the annotation process with each other to reduce cognition bias.

- We reviewed the annotation results with the annotators, gave them our feedback, and answered their questions and ambiguities regarding tags.

- We train the annotators on Wassem.

The training process for annotators took more than two weeks. After the annotators’ training, we provide each annotator with one batch at the beginning of the week. Then, we ask them to annotate the batch during the week using Wassem unless they feel tired, bored, or sick. We do so to assure the quality of the annotation. In the following week, we annotate the same five batches as the previous week, but each annotator will annotate another batch. By the end of the second week, each batch will be annotated by two annotations. Within five weeks, the annotators were able to annotate 27 batches.

After annotating a batch, we asked all annotators to discuss the disagreement cases and to agree on a decision regarding a disagreement case.

The data shows that there are 2,949 disagreement cases (5.3%) during the annotation process, i.e., the annotators agreed on 94.7% of annotation examples. Furthermore, the annotated data shows that there were 506 cases (0.9%) where a single annotator could not determine the tag for a given word. 21 of these words were shared between two annotators. All these cases were resolved each week during the process of annotation revision.

4.4 Statistics

The statistics on the data show that the CAraNER corpus comprises 55,389 tokens distributed over 1,278 sentences containing 3,813 named entities. Table 1 illustrates the distribution of named entities and examples of each named entity type. Note that the percentage of words that have “O” tags in the dataset is 84.5%.

5 Evaluation

NER is usually treated as a sequence labeling problem. In the literature, early studies used CRF models (Konkol and Konopík, 2013). Later, deep learning sequence models such as LSTMs have been used in many studies for NER (Zhang and Yang, 2018). More recently, pre-trained language models have been used to model NER, and they outperform previous models (Yohannes and Amagasa, 2022).

Tag	%	Examples
PER	19.3%	ماكاريوس، خالد بن فهد، وليد الصمعاني، سلمان بن عبدالعزيز، كرستيانو رونالدو، نوف بنت خالد الجريوي، أبو فراس الحمداني، محمد بن سلمان بن عبدالعزيز، رجب طيب أردوغان، جو بايدن. Makarios, Khalid bin Fahd, Walid Al-Samaani, Salman bin Abdulaziz, Cristiano Ronaldo, Nouf bint Khalid Al-Jeriwi, Abu Firas Al-Hamdani, Mohammed bin Salman bin Abdulaziz, Recep Tayyip Erdogan, Joe Biden
TIT	21.4%	السفير السوفيتي، الرئيس القبرصي، الأمير، معالي وزير العدل، الدكتور، فضيلة الشيخ، الرئيس التنفيذي لنوفا، المهندسة، خادم الحرمين الشريفين Soviet Ambassador, Cypriot President, Emir, Honorable Minister of Justice, Dr., Sheikh, CEO of Nova, Engineer, Custodian of the Two Holy Mosques
LOC	12.9%	المسجد الحرام، معدن شمام، وادي الفرع، كولومبيا، لوسون، القاهرة، بحيرة مالاوي، ولاية كارولاينا الشمالية، مسرح نورد، مأرب Al-Masjid Al-Haram, Ma'aden Shemam, Wadi Al-Fara', Columbia, Lawson, Cairo, Lake Malawi, North Carolina, Nord Theater, Ma'rib
ORG	25.0%	فيسبوك، جماعة الإخوان، قوات الأمن البيئي، وزارة الداخلية، إدارة سقيا زمزم، الهيئة السعودية للبيانات والذكاء الاصطناعي، المركز الوطني للوقاية من الأمراض ومكافحتها، مليشيات الحوثي، ليفربول، الكونجرس الأمريكي Facebook, Brotherhood, Environmental Security Forces, Ministry of Interior, Zamzam Water Department, Saudi Authority for Data and Artificial Intelligence, National Center for Disease Prevention and Control, Houthi militias, Liverpool, US Congress
MIS	21.4%	ماء زمزم، فيروس كورونا المستجد، العلمانية، الفضة، قبيلة باهلة، دوري أبطال آسيا، توكلنا،السكري، مهرجان أفلام السعودية، نظام حماية الأجور للعمالة المنزلية Zamzam water, the emerging coronavirus, secularism, silver, the Bahla tribe, the Asian Champions League, Tawakkalna, diabetes, the Saudi Film Festival, the wage protection system for domestic workers.

Table 1: Named entities distribution with examples from the CAraNER corpus.

We evaluate the CAraNER dataset by fine-tuning four BERT-based language models: bert-base-multilingual-cased (baseline), AraBERTv0.2-large, CAMeLBERT-MSA, and GigaBERT-v4. All of these models are based on BERT-base except AraBERT, which is based on BERT-large.

We fine-tuned the language models on the Google Colab platform using Tesla GPUs. We considered the following for experimentation setup for all models:

- From Huggingface, we used *transformers* v4.21.1, *AutoTokenizer*, and *BertForTokenClassification* libraries.
- We use *AdamW* for optimization with learning rate = 3e-5.
- We split data into 80% for training and 20% for testing (randomly selected).
- We select the number of Epochs = 16.
- We set the value for Max_grad_norm = 1.0.
- We set sentence_max_length = 295 (length of the longest sentence in the corpus).
- We choose batch size = 4.

Model	Acc.	Prec.	Recall	F1
mBERT	0.95	0.78	0.77	0.77
AraBERT	**0.97**	**0.86**	**0.86**	**0.86**
CAMeLBERT	0.96	0.83	**0.86**	0.84
GigaBERT	0.96	0.81	0.8	0.8

Table 2: Performance measures (accuracy, macro averaged precision, recall, and F-1) for the fine-tuned language models. mBERT: *bert-base-multilingual-cased.*

Table 2 shows the performance measures (macro avg) for the four fine-tuned language models. We consider the macro F1 measure when comparing the models. The results suggest that the

Tag	mBERT	AraBERTv0.2-large	CAMeLBERT-MSA	GigaBERT-v4
B-LOC	0.63	**0.75**	0.71	0.71
B-MIS	0.61	**0.74**	0.69	0.64
B-ORG	0.69	0.83	**0.85**	0.8
B-PER	0.89	**0.97**	0.94	0.91
B-TIT	0.86	**0.92**	0.88	0.86
I-LOC	0.61	0.76	**0.78**	0.74
I-MIS	0.58	**0.64**	**0.64**	0.55
I-ORG	0.79	0.9	**0.91**	0.82
I-PER	0.94	**1**	0.98	0.97
I-TIT	0.9	0.91	**0.93**	0.85
O	0.98	**0.99**	0.98	0.98

Table 3: F1 measure for each named entity tag. mBERT: *bert-base-multilingual-cased*.

AraBERTv0.2-large language model outperforms the other models followed by CAMeLBERT-MSA.

The superiority of the AraBERT over other models can be explained by the fact that AraBERTv0.2-large is much larger than the other models. In particular, AraBERTv0.2-large has 371M parameters, whereas the other models are based on the smaller model BERT-base, which has less than half this number of parameters. However, we observe that CAMeLBERT-MSA achieved a comparable performance by only using less than half of the model size. This relatively good performance of CAMeLBERT-MSA can be attributed to the size of the data on which the model was trained compared to the other models.

From these results, we observe that all models achieved an accuracy score of more than 95%. This can be attributed to the fact that most of the words have an “O” tag, which makes it easy to achieve such a high accuracy score. In particular, only 3,813 (∼7%) out of 55,389 tokens are named entities, and 51,576 (∼93%) of the tokens are not.

Table 3 shows the F1 score of the four fine-tuned models on CAraNER for each tag. The results show that all models have the same relative performance order for named entity tags. We observe that the best performance was on the PER tag, followed by the TIT, ORG, LOC, and MIS tags, respectively. The relatively high performance on the PER and TIT tags is probably due to the repetition of public figures’ person names and their titles in newspapers. For the ORG and LOC tags, the errors were due to wrong identification for the beginning of their named entities. The low performance of the MIS tag can be attributed to the diversity of named entities it contains.

6 Conclusion

In this paper, we introduced a web-based annotation platform for Arabic NLP (Wassem) and a new dataset for Arabic NER (CAraNER) annotated with five tags (PER, TIT, LOC, ORG and MIS) using Wassem. Experimentation on four BERT-based language models shows that fine-tuning AraBERTv0.2-large on CAraNER gives the best results among the other models, with a 0.86 macro F-1 score. Also, the relatively good performance of CAMeLBERT-MSA (0.84 macro F-1 score) may suggest that using large and diverse datasets for pre-training smaller language models (i.e., BERT-base) gives similar performance to larger models (i.e., BERT-large) pre-trained on smaller datasets. In the future, we plan to double the size of CAraNER to improve the performance and experiment with different Arabic language models.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments and feedback. Also, they would like to thank the annotators for this corpus during their summer internships at KACST, namely, Abdulaziz Alghuwainem, Amjad Alshalhoub, Dalal Alqahtany, Dana AlDukhail, and Marwan Alobathani.

References

Abdulmohsen Al-Thubaity, Sakhar Alkhereyf, and Alia O. Bahanshal. 2022. AraNPCC: The Arabic newspaper covid-19 corpus. In *Proceedinsg of the 5th Workshop on Open-Source Arabic Corpora and Processing Tools with Shared Tasks on Qur’an QA and Fine-Grained Hate Speech Detection*, pages 32–

40, Marseille, France. European Language Resources Association.

Ahmed Aliwy, Ayad Abbas, and Ahmed Alkhayyat. 2021. NERWS: Towards improving information retrieval of digital library management system using named entity recognition and word sense. *Big Data and Cognitive Computing*, 5(4):59.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *LREC 2020 Workshop Language Resources and Evaluation Conference 11–16 May 2020*, page 9.

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.

Bojan Bozic, Jayadeep Kumar Sasikumar, and Tamara Matthews. 2021. KnowText: Auto-generated knowledge graphs for custom domain applications. In *The 23rd International Conference on Information Integration and Web Intelligence*, pages 350–358.

Aditya Kiran Brahma, Prathyush Potluri, Meghana Kanapaneni, Sumanth Prabhu, and Sundeep Teki. 2021. Identification of food quality descriptors in customer chat conversations using named entity recognition. In *8th ACM IKDD CODS and 26th COMAD*, pages 257–261.

Bilge Celik and Joaquin Vanschoren. 2021. Adaptation strategies for automated machine learning on evolving data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(9):3067–3078.

Hai Leong Chieu and Hwee Tou Ng. 2003. Named entity recognition with a maximum entropy approach. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003*, pages 160–163.

Kareem Darwish and Wei Gao. 2014. Simple effective microblog named entity recognition: Arabic as an example. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2513–2517.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ajeet Grewal and Jimmy Lin. 2018. The evolution of content analysis for personalized recommendations at Twitter. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 1355–1356.

Ismail Harrando and Raphaël Troncy. 2021. Improving media content recommendation with automatic annotations. In *KaRS 2021, 3rd Edition of Knowledge-aware and Conversational Recommender Systems*.

Go Inoue, Bashar Alhafni, Nurpeiis Baimukan, Houda Bouamor, and Nizar Habash. 2021. The interplay of variant, size, and task type in Arabic pre-trained language models. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 92–104.

Mustafa Jarrar, Mohammed Khalilia, and Sana Ghanem. 2022. Wojood: Nested Arabic named entity corpus and recognition using BERT. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2022), Marseille, France, June*.

Michal Konkol and Miloslav Konopík. 2013. Crf-based czech named entity recognizer and consolidation of czech ner research. In *International conference on text, speech and dialogue*, pages 153–160. Springer.

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

Chenguang Liu, Yongli Yu, Xingxin Li, and Peng Wang. 2021. Named entity recognition in equipment support field using tri-training algorithm and text information extraction technology. *IEEE Access*, 9:126728–126734.

Hao Liu, Qinjun Qiu, Liang Wu, Wenjia Li, Bin Wang, and Yuan Zhou. 2022. Few-shot learning for name entity recognition in geological text based on GeoBERT. *Earth Science Informatics*, pages 1–13.

A Uma Maheswari, N Revathy, T Guhan, B Praveen, and R Magesh Kumar. 2022. A systematic bpclstm algorithm for concept drift detection incorporated sentiment mining. In *2022 International Conference on Computer Communication and Informatics (ICCCI)*, pages 1–8. IEEE.

Songzhu Mei, Cong Liu, Qinglin Wang, and Huayou Su. 2022. Model provenance management in mlops pipeline. In *2022 The 8th International Conference on Computing and Data Engineering*, pages 45–50.

Alexis Mitchell, Stephanie Strassel, Shudong Huang, and Ramez Zakhary. 2005. Ace 2004 multilingual training corpus. *Linguistic Data Consortium, Philadelphia*, 1:1–1.

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.

Zara Nasar, Syed Waqar Jaffry, and Muhammad Kamran Malik. 2021. Named entity recognition and relation extraction: State-of-the-art. *ACM Computing Surveys (CSUR)*, 54(1):1–39.

Keqin Peng, Chuantao Yin, Wenge Rong, Chenghua Lin, Deyu Zhou, and Zhang Xiong. 2021. Named entity aware transfer learning for biomedical factoid question answering. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*.

Ramzi Esmail Salah and Lailatul Qadri Binti Zakaria. 2018. Building the Classical Arabic named entity recognition corpus (canercorpus). In *2018 Fourth International Conference on Information Retrieval and Knowledge Management (CAMP)*, pages 1–8. IEEE.

Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, et al. 2013. Ontonotes release 5.0 ldc2013t19. *Linguistic Data Consortium, Philadelphia, PA*, 23.

Gezheng Xu, Wenge Rong, Yanmeng Wang, Yuanxin Ouyang, and Zhang Xiong. 2021. External features enriched model for biomedical question answering. *BMC bioinformatics*, 22(1):1–19.

Hailemariam Mehari Yohannes and Toshiyuki Amagasa. 2022. Named-entity recognition for a low-resource language using pre-trained language model. In *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*, pages 837–844.

Yue Zhang and Jie Yang. 2018. Chinese ner using lattice LSTM. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1554–1564.