Cross-Dialectal Named Entity Recognition in Arabic

Niama Elkhbir[†], **Urchade Zaratiana**[*†], **Nadi Tomeh**[†], **Thierry Charnois**[†]
[⋆] FI Group, [†] LIPN, CNRS UMR 7030, France
{elkhbir,zaratiana,tomeh,charnois}@lipn.fr

Abstract

In this paper, we study the transferability of Named Entity Recognition (NER) models between Arabic dialects. This question is important because the available manually-annotated resources are not distributed equally across dialects: Modern Standard Arabic (MSA) is much richer than other dialects for which little to no datasets exist. How well does a NER model, trained on MSA, perform on other dialects? To answer this question, we construct four datasets. The first is an MSA dataset extracted from the ACE 2005 corpus. The others are datasets for Egyptian, Moroccan, and Syrian which we manually annotate following the ACE guidelines. We train a span-based NER model on top of a pretrained language model (PLM) encoder on the MSA data and study its performance on the other datasets in zero-shot settings. We study the performance of multiple PLM encoders from the literature and show that they achieve acceptable performance with no annotation effort. Our annotations and models are publicly available (https://github.com/niamaelkhbir/Arabic-Cross-Dialectal-NER).

1 Introduction

The Arabic language, encompassing Classical Arabic (CA), Modern Standard Arabic (MSA), and various Dialects of Arabic (DA), stands out for its linguistic diversity and intricate morphology. This linguistic complexity presents a unique challenge for Natural Language Processing (NLP) tasks, particularly in the field of named entity recognition (NER). Modern Standard Arabic serves as the formal reference, and many research efforts have been dedicated to MSA NER. The literature on MSA NER methods has witnessed an evolution from rule-based methods, to machine learning models based on hand-crafted features and subsequently deep learning models incorporating rich contextual representations. Notably, pretrained transformer-based language models have recently driven significant advancements in Arabic NER.

Arabic, however, has more than 20 distinct dialects and around 100 regional variants, which are widely used in everyday communication, particularly in digital spaces. This emphasizes the urgent need for NLP models capable of effectively handling this linguistic diversity. However, these dialects exhibit significant linguistic variation, including differences in spelling, morphology, and syntax, making it exceptionally challenging to develop a unified global modeling approach. Additionally, there is no standardized spelling for these dialects. In addition, the scarcity of annotated dialectal data has been a major obstacle to progress in the field of dialectal NER.

Our research is driven by the goal of bridging the linguistic gap between MSA and Arabic dialects, specifically in the context of entity recognition. Given the substantial time required for the annotation process and leveraging the success of cross-lingual transfer learning, our work focuses on exploring knowledge transfer in the context of NER, transferring knowledge from MSA to various dialects.

Our contributions in this article are two-fold:

- We introduce a NER dataset manually annotated for three dialects: Moroccan, Egyptian, and Syrian. This dataset is used for evaluation purposes;
- We propose an efficient span-based NER model trained on already-available MSA data and analyze its transferability to other dialects.

2 Dataset and Annotation

In this section, we introduce our datasets for Modern Standard Arabic and Arabic Dialects (Moroccan, Egyptian, Syrian), their construction, and annotation guidelines.

2.1 Modern Standard Arabic Dataset

Our dataset for Modern Standard Arabic is sourced from the Arabic Corpus ACE 2005 (Walker and Consortium, 2005). The ACE corpus comprises a rich collection of text data from diverse sources, including newswires, broadcast news, and weblogs. This corpus includes annotations for seven distinct entity types, namely Persons (PER), Organizations (ORG), Geographical/Social/Political Entities (GPE), Locations (LOC), Facilities (FAC), Vehicles (VEH), and Weapons (VEH). In addition to entity types, it annotates three entity mention types: Names (NAM), Nominal Constructions (NOM), and Pronouns (PRO). The corpus offers annotations for both flat and nested entities, further including coreference information.

The MSA dataset we use in this work is based on ACE 2005. In its construction, we make the following choices:

- **Focus on NAM and NOM entities**: we opted to concentrate exclusively on the recognition of named entities and nominal constructions while excluding pronouns. ACE 2005 is notable for its detailed annotation, including pronouns, which is uncommon in the typical named entity recognition task that primarily deals with nominal entities and names. Pronoun usage exhibits considerable variation, displaying nuanced distinctions not only between dialects but even within distinct regions of the same dialect. Consequently, accurately annotating pronouns across dialects presents practical challenges and potential ambiguity, due to their strong contextual reliance and the absence of comprehensive dialect-specific guidelines. The inclusion of pronouns is therefore left to future work. For clarity, named entities include examples such as جون (*John*) and رام الله (*Ramallah*), while nominal entities include examples like المحامي (*The lawyer*) and ميناء (*Port*). Pronominal entities, which we chose to exclude, include terms such as هم (*they*), بعض (*some*), and كثيرون (*many*).

- **Focus on flat entities**: we opted to concentrate exclusively on flat entities, omitting nested entities and coreference resolution. This choice simplifies the task significantly by reducing complexity in both annotation and modeling. Nesting and coreference, while valuable areas of study, introduce intricate challenges, especially in dialectal Arabic, where linguistic variations are prevalent. Focusing on flat entities streamlines our research process, making it more scalable for testing across dialects.

Considering these two methodological decisions, we constructed our MSA dataset from the ACE 2005 corpus by randomly selecting 500 sentences. We provide detailed statistics about these sentences in the first columns of Tables 1 and 2.

This dataset will be used to train a model and study its transferability to other dialects. It will also be used to evaluate models that are trained on other dialects.

We also extracted an additional 350 MSA sentences to train an MSA model and evaluate it on the 500 sentences for reference. More details can be found in the results section (5)

2.2 Annotation Guidelines for Dialects

We introduce concise yet comprehensive annotation guidelines that were used in the annotation of our dialectal datasets. These guidelines closely follow the ACE guidelines that were used for the MSA dataset. The detailed reference is provided by the Linguistic Data Consortium (LDC) guidelines[1].

1. PER (Person): This entity type is used for individual human beings. It includes:

   - Names and surnames of individuals. *Example*: ميت رومني (*Mitt Romney*)
   - Group of people. *Example*: العائلة (*The family*).
   - Saints and other religious figures. *Example*: آللّه (*God*).

2. ORG (Organization): This entity type is used for corporations, agencies, and other groups of people defined by an organization structure. It includes:

   - Commercial organizations. *Example*: ميكروسوفت (*Microsoft*)
   - Government organizations. *Example*: البحرية الملكية (*Royal Navy*).

[1] https://www.ldc.upenn.edu/collaborations/past-projects/ace/annotation-tasks-and-specifications

- Educational organizations. *Example*: جامعة ستانفورد (*Stanford University*).
- Political parties. *Example*: الحزب الليبرالي (*Liberal Party*).
- Media. *Example*: وكالة انسا (*ANSA agency*).

3. LOC (Location): This entity type is used for geographical entities such as mountains, rivers, seas, and regions that aren't politically defined. *Example*: شمال نيو مكسيكو (Northern New Mexico).

4. GPE (Geographical/Social/Political Entity): This entity type is used for geographical regions that have a political distinction. This includes countries, states, provinces, and cities. *Example*: أمريكا (*America*).

5. VEH (Vehicle): This entity type is used for entities that are primarily designed for transporting goods or people from one place to another. *Example*: عربة (*vehicle*).

6. WEA (Weapon): This entity type is used for devices used with intent to inflict damage or harm.
   - Exploding. *Example*: قنابل (*Bombs*).
   - Chemical. *Example*: الغاز (*Gas*).
   - Underspecified. *Example*: سلاح (*Weapon*).

7. FAC (Facility): This entity type is used for buildings or structures. It includes buildings, houses, factories, stadiums, office buildings, gymnasiums, prisons, museums, space stations, barns, parking garages and airplane hangars, streets, highways, airports, ports, train stations, bridges, and tunnels. *Example*: المطار (*The airport*).

We adhere to these guidelines by annotating the smallest constituent of flat entities. For example, consider the entity بطل الولايات المتحدة (*United States champion*). In this case, we annotate الولايات المتحدة (*United States*) as GPE and بطل (*champion*) as PER. If our task involved nested entities, we would have provided additional annotations for the entire nested entity بطل الولايات المتحدة as PER.

Stat	**MSA**	**Mor.**	**Egy.**	**Syr.**
Sentences	500	378	353	361
Tokens	14168	6780	6533	6034
Entities	3030	970	831	956

Table 1: Dialect Dataset Statistics. **MSA**: Modern Standard Arabic, **Mor.**: Moroccan, **Egy.**: Eyptian, **Syr.**: Syrian.

Ent	**MSA**	**Mor.**	**Egy.**	**Syr.**
FAC	143	83	63	71
GPE	923	249	229	331
LOC	160	191	142	89
ORG	413	112	77	109
PER	1269	278	264	307
VEH	52	45	50	41
WEA	70	12	6	8

Table 2: Dialect Dataset Statistics by Entity Type. **MSA**: Modern Standard Arabic, **Mor.**: Moroccan, **Egy.**: Eyptian, **Syr.**: Syrian.

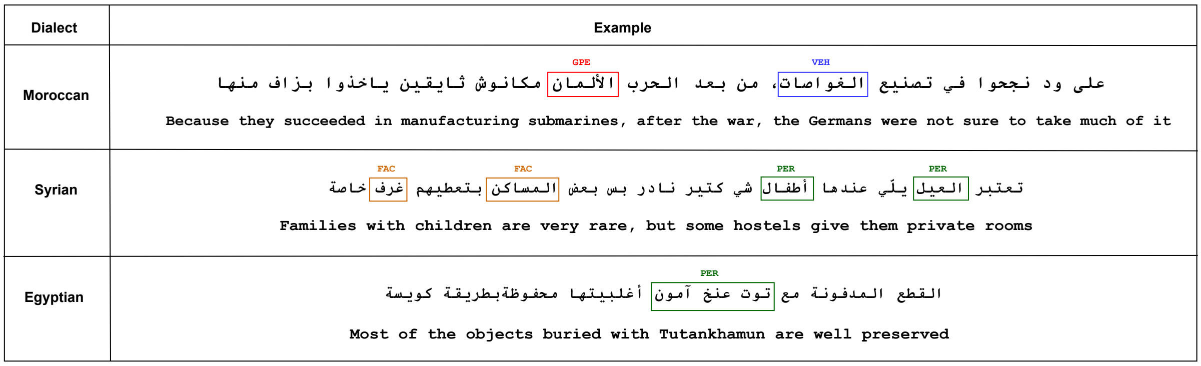
2.3 Annotation Process of the Dialect Datasets

Our dataset for Arabic Dialects is sourced from the xP3x corpus (Muennighoff et al., 2022). The xP3x corpus comprises a vast collection of prompts and datasets across 277 languages, covering 16 distinct NLP tasks. This corpus comprises pairs of sentences and their translations in various languages.

3 Task Definition and Model

In this study, we opted to work with three distinct Arabic dialects: Moroccan, Egyptian, and Syrian. For each dialect, we selected randomly 500 sentences from the xP3x corpus and tokenized them by whitespaces before presenting them for annotation. Notably, our annotation process was overseen by a single annotator, a proficient Moroccan Arabic speaker, with a deep understanding of Egyptian and Syrian dialects as well. The limited dataset size made the use of a single annotator optimal, as this approach ensured consistency, coherence, and a manageable workload, minimizing inter-annotator discrepancies and maintaining unified annotation styles.

In this study, we chose to investigate three distinct Arabic dialects: Moroccan, Egyptian, and Syrian. We randomly selected 500 sentences from the xP3x corpus for each dialect and tokenized them using whitespace. Our annotation process, carried out using Label Studio as the annotation tool, was supervised by a single proficient annotator, fluent



Dialect	Example
Moroccan	GPE VEH على ود نجحوا في تصنيع الغواصات، من بعد الحرب الألمان مكانوش ثايقين ياخذوا بزاف منها Because they succeeded in manufacturing submarines, after the war, the Germans were not sure to take much of it
Syrian	FAC FAC PER PER تعتبر العيل يلّي عندها أطفال شي كتير نادر بس بعض المساكن بتعطيهم غرف خاصة Families with children are very rare, but some hostels give them private rooms
Egyptian	PER القطع المدفونة مع توت عنخ آمون أغلبيتها محفوظةبطريقة كويسة Most of the objects buried with Tutankhamun are well preserved

Figure 1: Example of annotations from our Dialect Dataset.

in Moroccan Arabic and possessing a strong grasp of Egyptian and Syrian dialects. Given the limited dataset size, employing a single annotator was advantageous for maintaining consistency, coherence, and manageable workloads, thereby reducing inter-annotator discrepancies and ensuring uniform annotation styles.

After the annotation process, we only retained sentences containing entities for our experiments. For a comprehensive overview of the dataset's statistics, please consult Tables 1 and 2. To visualize examples from our dataset, please refer to Figure 1.

Named Entity Recognition involves identifying and categorizing named entities within text into predefined entity categories. Formally, we frame the task of NER as a span classification problem. Given an input sequence: $\boldsymbol{x} = \{x_i\}_{i=1}^{L}$, our objective is to classify all potential spans within the sequence, defined as:

$$\boldsymbol{y} = \bigcup_{i=1}^{L} \bigcup_{j=i}^{L} s_{ijc} \tag{1}$$

Here, i, j, and c correspond to the start position, end position, and span type, respectively. The probability of a specific span classification $\boldsymbol{y}$ given the input sequence $\boldsymbol{x}$ is represented as:

$$p_\theta(\boldsymbol{y}|\boldsymbol{x}) = \frac{\exp \sum_{s_{ijc} \in \boldsymbol{y}} \phi_\theta(s_{ijc}|\boldsymbol{x})}{\mathcal{Z}_\theta(\boldsymbol{x})} \tag{2}$$

In this equation, $\phi_\theta(.)$ is the span scoring function, and $\mathcal{Z}_\theta(\boldsymbol{x})$ is the partition function. During training, our objective is to minimize the negative log-likelihood of the gold span classifications.

Training loss During training, our assumption allows us to bypass the need to explicitly evaluate the partition function $\mathcal{Z}_\theta(\boldsymbol{x})$ to compute the loss. The loss for a single sample $(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{T}$ is simply the sum of loss for all spans in the input:

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{y}) = -\sum_{c_{ij} \in \boldsymbol{y}} \log p(c_{ij}|\boldsymbol{x}) \tag{3}$$

where,

$$p(c_{ij}|\boldsymbol{x}) = \frac{\exp \phi_\theta(c_{ij}|\boldsymbol{x})}{\sum_{c' \in \mathcal{C}} \exp \phi_\theta(c'_{ij}|\boldsymbol{x})} \tag{4}$$

This loss is minimized over the training set using a stochastic gradient descent algorithm.

Decoding During inference, our aim is to determine:

$$\boldsymbol{y}^* = \arg\max_{\boldsymbol{y} \in \mathcal{Y}} \sum_{s_{ijc} \in \boldsymbol{y}} \phi_\theta(s_{ijc}|\boldsymbol{x}) \tag{5}$$

In other words, we seek to identify the span labeling configuration that achieves the highest score. For unconstrained span classification, a straightforward approach is to assign the label with the highest score to each individual span, as follows:

$$s_{ijc^*} = \arg\max_{c} \phi_\theta(s_{ijc}|\boldsymbol{x}) \tag{6}$$

Nonetheless, this decoding approach is not optimal since it may result in structural constraint violations. In our context of flat entities, overlapping entity spans are strictly prohibited. A more efficient solution, as presented in our prior research (Zaratiana et al., 2022a,b)[2], employs a two-stage decoding process. Initially, spans predicted as non-entities are filtered out, followed by the application of a maximum independent set algorithm to the remaining spans to determine the optimal set of entity spans.

[2] https://github.com/urchade/Filtered-Semi-Markov-CRF

Token and Span Representations We compute the span score $\phi_\theta(s_{ijc}|\boldsymbol{x})$ by performing a linear projection of the span representation, which is derived from a $1D$ convolution applied to token representations obtained from a transformer-based model (eg. BERT):

$$\boldsymbol{s}_{ijc} := w_c^T \texttt{Conv1D}_k([\boldsymbol{h}_i; \boldsymbol{h}_{i+1}; \ldots; \boldsymbol{h}_j]) \quad (7)$$

Here, $h_i \in \mathbb{R}^D$ represents the token representation at position i, k signifies the size of the convolutional filter (corresponding to the span length), and $w_c \in \mathbb{R}^D$ denotes a learned weight matrix associated with span label c.

4 Experimental Setup

Token Encodings To encode our input tokens, we use 8 diverse pretrained language models, i.e trained on diverse dataset sources: Arabic MSA dataset (ARBERTv2 and CAMeLBERT-MSA), Arabic dialect dataset (MARBERTv2 and CAMeLBERT-DA), Mixture of MSA and Arabic dialect (AraBERTv2 and CAMeLBERT-Mix), and multilingual dataset (mBERT and mDeBERTa). We detail them below:

- ARBERTv2: (Abdul-Mageed et al., 2021): A large-scale pretrained masked language model for MSA with 12 attention layers, 12 heads, 768 hidden dimensions, and 163M parameters, trained on 61GB of Arabic text.

- MARBERTv2 (Abdul-Mageed et al., 2021): A large-scale pretrained masked language model for both DA and MSA, trained on 1B Arabic tweets (128GB text, 15.6B tokens), using the same architecture as ARBERT (BERT-base) without next sentence prediction.

- AraBERTv2 (Antoun et al., 2020): The dataset consists of 77GB Arabic text from diverse sources. It uses the same architecture as BERT-Base.

- CAMeLBERT-DA (Inoue et al., 2021): A collection of pretrained BERT models for Arabic dialects, trained on a diverse dataset of 54GB, totaling 5.8 billion tokens.

- CAMeLBERT-Mix (Inoue et al., 2021): A collection of pretrained BERT models for Arabic, including MSA, DA, and CA, trained on a diverse dataset of 167GB, totaling 17.3 billion tokens.

- CAMeLBERT-MSA (Inoue et al., 2021): A collection of pretrained BERT models for MSA, trained on a diverse dataset of 107GB, totaling 12.6 billion tokens.

- mBERT (Devlin et al., 2019): The multilingual version of BERT pretrained on the top 104 languages with the largest Wikipedia using a masked language modeling (MLM) objective.

- mDeBERTa: A multilingual version of DeBERTa (He et al., 2020) trained with CC100 multilingual data.

Hyperparameters We train all our models up to convergence. We use a training batch size of 12 and a validation batch size of 32. We employed a learning rate of `2e-5` for the pre-trained parameters and a learning rate of `3e-3` for the other parameters. We used a batch size of 8 and trained all the models to convergence (near 0 training loss). For testing, we use the last model, given the limited availability of validation data in our dataset. To manage the complexity of the task, we impose a constraint on the maximum span length, setting it to a maximum width of $K = 10$. This constraint significantly reduces the number of segments from L^2 to LK. The pretrained transformer models were loaded from HuggingFace's Transformers library, we used AllenNLP for data preprocessing. We trained all the models on a server equipped with V100 GPUs.

Evaluation Metrics We adopt the standard NER evaluation methodology, calculating precision (P), recall (R), and F1-score (F), based on the exact match between predicted and actual entities.

5 Results

The main results of our experiments are shown in Figure 2. We conducted two primary experiments: firstly, training on Modern Standard Arabic, and evaluating on dialects, and secondly, reversing this configuration, training on individual dialects and assessing on MSA. For both scenarios, we used the complete dataset outlined in Table 1. In addition, we conducted MSA-to-MSA experiments, where we evaluated our model on the MSA dataset specified in Table 1, while the training set consisted of a random selection of 350 sentences drawn from the original Arabic ACE dataset, using the same preprocessing steps detailed in Section 2.1.

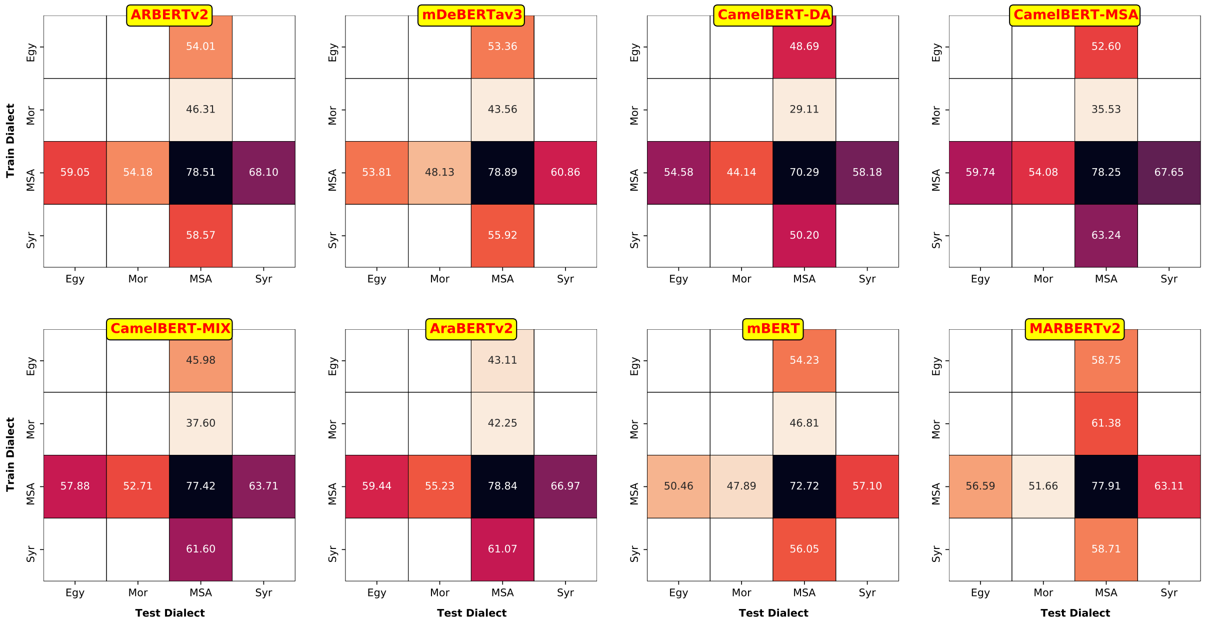


Figure 2: Comparative performance of models across different training and testing settings in terms of F1 score.

MSA-to-MSA The performance metrics reveal that MSA-to-MSA settings consistently yield the highest accuracy across all tested configurations, a result that aligns with expectations given that Modern Standard Arabic often serves as the benchmark for Arabic language tasks. Interestingly, most backbone models such as ARBERTv2, mDeBERTav3, CAMeLBERT-MSA (Inoue et al., 2021), CAMeLBERT-Mix (Inoue et al., 2021), AraBERTv2 and MARBERTv2 demonstrate comparable performance, suggesting that their architecture and training data are well-suited for MSA-centric tasks. Two models, however, diverge from this trend. CAMeLBERT-DA (Inoue et al., 2021) exhibits an 8% drop in performance compared to the other language models, which can be attributed to its focus on dialectal data during training. This specialization likely limits its ability to generalize effectively to MSA. Similarly, mBERT performs less well. As a multilingual model, mBERT may suffer from language interference or tokenization issues, given its training on a diverse corpus where Arabic is not the dominant language.

Test	Best Model	Avg. F1
Egyptian	CAMeLBERT-MSA	59.74
Moroccan	AraBERTv2	55.24
Syrian	ARBERTv2	68.10

Table 3: Best-Performing Language Model for test Dialect (F1-score).

Train	Best Model	Avg. F1
Egyptian	MARBERTv2	58.75
Moroccan	MARBERTv2	61.38
Syrian	CAMeLBERT-MSA	63.24

Table 4: Best-Performing Language Model for train Dialect (F1 score).

MSA to Dialects When training models on the MSA dataset, the observed performance metrics indicate a hierarchical trend among the tested Arabic dialects. The best performances are systematically obtained with the Syrian dialect, followed by the Egyptian dialect, and finally the Moroccan dialect. This gradient could be indicative of the linguistic similarities and differences between MSA and these dialects. The Syrian dialect may share more syntactic and semantic features with MSA, allowing models trained on MSA to generalize more easily to Syrian. On the other hand, the Moroccan dialect appears to be the most divergent from MSA among the tested dialects, resulting in the lowest performance scores. This could be due to unique lexical, grammatical, or even phonological features that are not adequately captured when a model is trained solely on MSA data.

Dialects to MSA Similar to the MSA to dialects scenario, the best test performance on MSA is obtained when models are trained on the Syrian dialect, followed by the Egyptian dialect and finally the Moroccan dialect. This pattern aligns well with the earlier observation that models trained on MSA perform best on the Syrian dialect, thereby suggesting a mutual linguistic affinity between Syrian and MSA. Models trained on Egyptian also perform relatively well, reinforcing the notion of shared linguistic features between Egyptian and MSA. Conversely, the Moroccan dialect, which was identified as the most challenging for models trained on MSA, also proves to be the least effective training data for models tested on MSA. This consistent underperformance across both scenarios could point to a greater linguistic divergence between Moroccan and MSA, which may involve lexical, syntactic, or phonological differences not easily bridged by the models in question.

Optimal Language Model for MSA Training When training with an MSA dataset, AraBERTv2 emerges as the top-performing language model, with an average score of 65.12 across various Arabic dialects. The strength of this model can be attributed to its well-balanced training regimen, which combines both MSA and dialectal data, resulting in a harmonious blend of specialization and generalization. Models explicitly trained on MSA, namely ARBERTv2 and CAMeLBERT-MSA, closely follow in terms of performance, underscoring the effectiveness of MSA-focused training. In contrast, dialect-specific models like MARBERTv2 and CAMeLBERT-DA still deliver respectable results, although falling behind their MSA-centric counterparts. Interestingly, multilingual models like mDeBERTav3 and mBERT rank lower in performance, possibly due to language interference issues. Overall, our data suggests that a balanced training approach, as exemplified by AraBERTv2, offers the most effective strategy for tasks involving MSA and its various dialects.

Optimal Language Models for Each Dialect Our investigation underscores the significant impact of the choice of language model on the performance of dialectal NER tasks. We find that for the Egyptian and Moroccan dialects, MARBERTv2 excels as the most effective model. This can be attributed to its specialized training on dialectal data, allowing it to capture the nuances specific to these dialects and deliver superior results. In the case of the Syrian dialect, CAMeLBERT-MSA takes the lead. Interestingly, this model is primarily trained on MSA but appears to generalize well to the Syrian dialect, perhaps due to linguistic similarities between the two. This emphasizes the importance of model-dialect congruence, where using a model trained on the same or similar dialect as the dataset can yield better performance.

Dialect	**Mixture**	**Mono** (Best)
ARBERTv2	64.56	58.57 (Syr.)
AraBERTv2	58.61	55.92 (Syr.)
CAMeLBERT-DA	54.84	50.20 (Syr.)
CAMeLBERT-Mix	61.49	61.60 (Syr.)
CAMeLBERT-MSA	63.30	63.24(Syr.)
mBERT	58.60	56.05 (Syr.)
MARBERTv2	66.10	61.38 (Mor.)
mDeBERTav3	60.27	55.92 (Syr.)

Table 5: Performance for MSA when training on a mixture of dialects. We compare the result with the best obtained result when training on a single dialect.

Training on Mixture of Dialects In the context of training on a mixture of Arabic dialects and evaluating on the Modern Standard Arabic (MSA) dataset, our analysis reveals intriguing insights into the impact of dialectal diversity on MSA performance. Remarkably, the performance metrics suggest that training on a mixture of dialects consistently yields competitive accuracy on the MSA dataset. This shows that exposure to a diverse range of dialects during training can enhance a model's adaptability and robustness, enabling it to perform well on MSA.

Effect of Increased MSA Training Data While training on a diverse range of dialects typically enhances performance for Modern Standard Arabic (MSA), it is important to note that training on additional MSA data may not necessarily lead to improved performance in dialects, as demonstrated in Table 6.

6 Related Work

Named Entity Recognition for Modern Standard Arabic The development of Named Entity Recognition techniques in Modern Standard Arabic has been a central focus within the Arabic NLP community. Initially, rule-based NER systems like those described in Shaalan and Raza (2008); Abdallah et al. (2012) relied on manually crafted grammatical rules and gazetteers. While

Model	ARBERTv2	MARBERTv2	AraBERTv2	CAMeLBERT-DA	CAMeLBERT-Mix	CAMeLBERT-MSA	mBERT	mDeBERTav3
Egyptian	55.42	58.29	60.38	53.65	55.19	60.28	53.92	56.78
Moroccan	53.03	54.35	54.52	44.43	50.43	53.31	47.57	51.30
MSA	84.96	84.02	86.61	80.49	84.10	85.51	81.90	84.71
Syrian	65.51	64.45	66.87	57.68	62.81	66.47	59.82	63.36

Table 6: Effect of Increased MSA Data on Performance.

effective, these systems demanded extensive maintenance and lacked scalability. Subsequently, machine learning-based NER methods, as demonstrated by Benajiba and Rosso (2007); Al-Qurishi and Souissi (2021), treated NER as a classification task, leveraging large annotated datasets. This era also witnessed the fusion of rule-based and machine learning-based approaches through hybrid systems (Oudah and Shaalan, 2012; Meselhi et al., 2014), followed by the adoption of deep learning techniques, which allowed for the automatic extraction of intricate features. Deep learning, characterized by neural networks processing word and character embeddings, marked a departure from manual feature engineering, resulting in significantly improved accuracy and a more streamlined approach to Arabic NER. In recent years, pretrained language models (PLMs) such as BERT (Devlin et al., 2019) have opened up a new era in Arabic NER. Arabic-specific PLMs, such as AraBERT (Antoun et al., 2020) and AraELECTRA (Antoun et al., 2021), have been meticulously developed and fine-tuned for NER tasks, offering the advantage of context-rich information. This evolution has given rise to a multitude of high-performance systems (Helwe et al., 2020; El Khbir et al., 2022).

Additionally, extensive annotation efforts have led to the creation of high-quality MSA NER datasets. ACE 2005 (Walker and Consortium, 2005) comprises a diverse text collection with annotations for seven entity types (PER, ORG, GPE, LOC, FAC, VEH, WEA), three mention types (NAM, NOM, PRO), and coreference information. ANERcorp (Benajiba et al., 2007) comprises articles from diverse sources. It includes traditional entity types (ORG, LOC, PER) and introduces a MISC (miscellaneous) type. AQMAR (Mohit et al., 2012) comprises hand-annotated text extracted from Arabic Wikipedia articles. It includes 28 articles categorized by domain, each tagged with named entities and custom entity classes. Wojood (Jarrar et al., 2022) comprises text sourced from different domains and manually annotated with 21 entity types, including both flat and nested entities.

Datasets and Named Entity Recognition for Arabic Dialects Few works addressed NER for Arabic dialects. Zirikly and Diab (2014) introduced an annotated dataset and a named entity recognition system tailored to the Egyptian dialect. However, their evaluation focused solely on two entity types: PER and LOC. In a subsequent work, Zirikly and Diab (2015) presented a gazetteer-free NER system tailored to the Egyptian dialect, evaluated on three entity types: PER, LOC, and ORG. Additionally, Moussa and Mourhir (2023) introduced a manually annotated NER dataset for the Moroccan dialect, which comprises 4 entity types: PER, LOC, ORG and MISC.

7 Conclusion and Future Work

In this work, we explore transfer learning for named entity extraction, specifically from Modern Standard Arabic (MSA) to various Arabic dialects, employing a range of pretrained language models. For this purpose, we annotated a dataset including Moroccan, Syrian, and Egyptian dialects. Our results showed that for both MSA-to-dialects and dialects-to-MSA scenarios, Syrian data demonstrated superior performance, which suggests a robust linguistic affinity between the Syrian dialect and MSA. Similarly, Egyptian models exhibited strong results. In contrast, models trained on the Moroccan dialect consistently face challenges, indicating substantial linguistic divergence between Moroccan Arabic and MSA.

In future work, we plan to include a wider range of Arabic dialects to better understand the nuances and generalization of our results across different dialectal variants. In addition, we plan to explore the nested entity task.

Limitations

While our study provides valuable insights into the transfer learning of named entity extraction between Modern Standard Arabic and Arabic dialects, it is important to acknowledge certain limitations:

- We focus on three Arabic dialects: Moroccan, Syrian and Egyptian. While they offer a rep-

representative sample of the diversity of Arabic, extending our dataset to other dialect variants would enable us to generalize our findings more effectively.

- The annotation of our dataset relies on a single annotator, which may be a potential source of bias. Future work should consider the involvement of multiple annotators to assess inter-annotator agreement and ensure labeling robustness.

Acknowledgements

This work is partially supported by a public grantoverseen by the French National Research Agency (ANR) as part of the program Investissements d'Avenir (ANR-10-LABX-0083). This work was granted access to the HPC/AI resources of [CINES/IDRIS/TGCC] under the allocation 2023AD011013096R1 made by GENCI.

References

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

Muhammad Abdul-Mageed, AbdelRahim Elmadany, and El Moatez Billah Nagoudi. 2021. ARBERT & MARBERT: Deep bidirectional transformers for Arabic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7088–7105, Online. Association for Computational Linguistics.

Muhammad Saleh Al-Qurishi and Riad Souissi. 2021. Arabic named entity recognition using transformer-based-CRF model. In *Proceedings of the 4th International Conference on Natural Language and Speech Processing (ICNLSP 2021)*, pages 262–271, Trento, Italy. Association for Computational Linguistics.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15, Marseille, France. European Language Resource Association.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2021. Araelectra: Pre-training text discriminators for arabic language understanding.

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. pages 1814–1823.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Niama El Khbir, Nadi Tomeh, and Thierry Charnois. 2022. ArabIE: Joint entity, relation and event extraction for Arabic. In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 331–345, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *ArXiv*, abs/2006.03654.

Chadi Helwe, Ghassan Dib, Mohsen Shamas, and Shady Elbassuoni. 2020. A semi-supervised BERT approach for Arabic named entity recognition. In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 49–57, Barcelona, Spain (Online). Association for Computational Linguistics.

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*, Kyiv, Ukraine (Online). Association for Computational Linguistics.

Mustafa Jarrar, Mohammed Khalilia, and Sana Ghanem. 2022. Wojood: Nested Arabic named entity corpus and recognition using BERT. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3626–3636, Marseille, France. European Language Resources Association.

Mohamed A. Meselhi, Hitham M. Abo Bakr, Ibrahim Ziedan, and Khaled Shaalan. 2014. A novel hybrid approach to arabic named entity recognition. In *Machine Translation*, pages 93–103, Berlin, Heidelberg. Springer Berlin Heidelberg.

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, Avignon, France. Association for Computational Linguistics.

Hanane Nour Moussa and Asmaa Mourhir. 2023. Darnercorp: An annotated named entity recognition dataset in the moroccan dialect. *Data in Brief*, 48:109234.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, et al. 2022. Crosslingual generalization through multitask finetuning. *arXiv preprint arXiv:2211.01786*.

Mai Oudah and Khaled Shaalan. 2012. A pipeline Arabic named entity recognition using a hybrid approach. In *Proceedings of COLING 2012*, pages 2159–2176, Mumbai, India. The COLING 2012 Organizing Committee.

Khaled Shaalan and Hafsa Raza. 2008. Arabic named entity recognition from diverse text types. In *Advances in Natural Language Processing*, pages 440–451, Berlin, Heidelberg. Springer Berlin Heidelberg.

C. Walker and Linguistic Data Consortium. 2005. *ACE 2005 Multilingual Training Corpus*. LDC corpora. Linguistic Data Consortium.

Urchade Zaratiana, Niama Elkhbir, Pierre Holat, Nadi Tomeh, and Thierry Charnois. 2022a. Global span selection for named entity recognition. In *Proceedings of the Workshop on Unimodal and Multimodal Induction of Linguistic Structures (UM-IoS)*, pages 11–17, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Urchade Zaratiana, Nadi Tomeh, Pierre Holat, and Thierry Charnois. 2022b. Named entity recognition as structured span prediction. In *Proceedings of the Workshop on Unimodal and Multimodal Induction of Linguistic Structures (UM-IoS)*, pages 1–10, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Ayah Zirikly and Mona Diab. 2014. Named entity recognition system for dialectal Arabic. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 78–86, Doha, Qatar. Association for Computational Linguistics.

Ayah Zirikly and Mona Diab. 2015. Named entity recognition for Arabic social media. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 176–185, Denver, Colorado. Association for Computational Linguistics.