

Adversarial Multitask Learning for Joint Multi-Feature and Multi-Dialect Morphological Modeling

Nasser Zalmout and Nizar Habash

Computational Approaches to Modeling Language Lab

New York University Abu Dhabi

United Arab Emirates

{nasser.zalmout, nizar.habash}@nyu.edu

Abstract

Morphological tagging is challenging for morphologically rich languages due to the large target space and the need for more training data to minimize model sparsity. Dialectal variants of morphologically rich languages suffer more as they tend to be more noisy and have less resources. In this paper we explore the use of multitask learning and adversarial training to address morphological richness and dialectal variations in the context of full morphological tagging. We use multitask learning for joint morphological modeling for the features within two dialects, and as a knowledge-transfer scheme for cross-dialectal modeling. We use adversarial training to learn dialect invariant features that can help the knowledge-transfer scheme from the high to low-resource variants. We work with two dialectal variants: Modern Standard Arabic (high-resource “dialect”¹) and Egyptian Arabic (low-resource dialect) as a case study. Our models achieve state-of-the-art results for both. Furthermore, adversarial training provides more significant improvement when using smaller training datasets in particular.

1 Introduction

Morphological tagging for morphologically rich languages (MRL) involves modeling interdependent features, with a large combined target space. Joint modeling of the different features, through feature concatenation, results in a large target space with increased sparsity. Whereas total separation of the different feature models eliminates access to the other features, which constrains the model. These issues are further exacerbated for dialectal content, with many morphosyntactic variations that further complicate the modeling.

¹We view Arabic as a collective of dialectal variants in which MSA is the main high-resource dialect, and EGY is a low-resource dialect. We therefore use “variant” and “dialect” interchangeably.

In this paper we work with Modern Standard Arabic (MSA) and Egyptian Arabic (EGY), both MRLs, and dialectal variants. Written Arabic text is also highly ambiguous, due to its diacritic-optional orthography, resulting in several interpretations of the same surface forms, and further increasing sparsity. Joint modeling is particularly promising for such ambiguous nature as it supports identifying more complex patterns involving multiple features. In EGY, for example, the suffix *نا* *nA* ‘we, us, our’ in the word *درسنا* *drsna* can be the subject of the perfective 1st person plural verb (‘we studied’), the 1st person plural object clitic of a perfective 3rd person masculine singular verb (‘he taught us’), or the 1st person plural possessive pronoun for the nominal (‘our lesson’), among other possible interpretations.

Morphological tagging models rely heavily on the availability of large annotated training datasets. Unlike MSA, Arabic Dialects are generally low on resources. In this paper we also experiment with knowledge-transfer models from high to low-resource variants. The similarities between the Arabic variants, both for MSA and Dialectal Arabic (DA), like EGY, should facilitate knowledge-transfer, making use of the resources of the high-resource variants. We use multitask learning architectures in several configurations for cross-dialectal modeling. We further investigate the best approaches and configurations to use word and character embeddings in the cross-dialectal multitask learning model, and whether mapping the various pretrained word embedding spaces is beneficial. Despite having several contributions in the literature, the role of mapped embedding spaces has not been studied in the context of joint morphological modeling of different dialects.

Finally, we use adversarial training to learn dialect-invariant features for MSA and EGY. The intuition is to make the modeling spaces for both

variants closer to each other, which should facilitate the knowledge-transfer scheme from the high-resource (MSA) to the low-resource (EGY) sides.

Our models achieve state-of-the-art morphological disambiguation results for both MSA and EGY, with up to 10% relative error reduction. Adversarial training proved more useful when using a smaller EGY training datasets in particular, simulating lower-resource settings. The contributions of the paper include (1) a joint multi-feature and cross-dialectal morphological disambiguation model for several MRL variants, (2) adversarial training for cross-dialectal morphological knowledge-transfer.

2 Linguistic Motivation

MRLs, like Arabic, have many morphemes that represent several morphological features. The target space for the combined morphological features in MRLs therefore tends to be very large. MRLs also tend to have more inflected words than other languages. MRLs also usually have a higher degree of ambiguity, with different interpretations of the same surface form. In Arabic, this ambiguity is exacerbated by the diacritization-optional orthography, which results in having about 12 analyses per word on average (Habash, 2010).

One approach to model morphological richness and ambiguity is to use *morphological analyzers*, which are used to encode all potential word inflections in the language. The ideal morphological analyzer should return all the possible analyses of a surface word (modeling ambiguity), and cover all the inflected forms of a word lemma (modeling morphological richness). The best analysis is then chosen through *morphological disambiguation*, which is essentially part-of-speech tagging for all the features in addition to lemma and diacritized form choices.

MSA is the written Arabic that is mainly used in formal settings. DA, like EGY, on the other hand, is the primarily spoken language used by native Arabic speakers in daily exchanges. DA has recently seen an increase in written content, due to the growing social media use in the region. DA, similar to MSA, is also morphologically rich, with a high degree of ambiguity. DA spans many Arabic dialects that are used across the Arab World, and they vary by the regions and cities they are used in (Bouamor et al., 2018). The large number of DA variants, along with it being mainly spoken,

result in DA being usually low on resources.

MSA and DA have many morphological, lexical and syntactic similarities that a cross-dialectal model can leverage (Habash et al., 2012). DA has many MSA cognates, both MSA and DA use the same script, and DA content in general includes a lot of code-switching with MSA.² These similarities can be useful in a joint learning model, enabling a knowledge-transfer scheme, especially from the high-resource to low-resource variants.

In this paper we focus on EGY as an example of DA. The set of morphological features that we model for both MSA and EGY can be:

- Open-Set Features: Lemmas (lex) and diacritized forms (diac), henceforth "lexicalized features". These features are unrestricted and have large and open vocabularies.
- Closed-Set Features: A set of 14 features, including inflectional features and clitics, each with a corresponding set of values/tags that are predicted using taggers. The inflectional features include: part-of-speech (POS), aspect (asp), case (cas), gender (gen), person (per), number (num), mood (mod), state (stt), voice (vox). The clitics include: enclitics, like pronominal and negative particle enclitics; proclitics, like article proclitic, preposition proclitics, conjunction proclitics, question proclitics.

Morphological disambiguation involves predicting the values for each of these features, then using these predictions to rank the different analyses from the morphological analyzer.

3 Background and Related Work

Joint Modeling in NLP Joint NLP modeling in general has been an active area of research throughout the past several years, supported by recent updates in deep learning architectures. Multitask learning models have been proven very useful for several NLP tasks and applications, (Collobert et al., 2011; Søgaard and Goldberg, 2016; Alonso and Plank, 2017; Bingel and Søgaard, 2017; Hashimoto et al., 2017). Inoue et al. (2017) used multitask learning for fine-grained POS tagging in MSA. We extend their work by doing cross-dialectal modeling and various contributions for low-resource dialects.

²Although EGY, like DA in general, does not have a standardized orthography like MSA (Habash et al., 2018).

Cross-Lingual Transfer Cross-lingual morphology and syntax modeling has also been a very active NLP research area, with contributions in morphological reinflection and paradigm completion (Aharoni et al., 2016; Faruqui et al., 2016; Kann et al., 2017), morphological tagging (Buys and Botha, 2016; Cotterell and Heigold, 2017), parsing (Guo et al., 2015; Ammar et al., 2016), among others. Cotterell and Heigold (2017) used multitask learning for multi-lingual POS tagging, similar in spirit to our approach. Their architecture, however, models the morphological features in each language in a single task, where each target value represents all morphological features combined. This architecture is not suitable for MRLs, with large target spaces.

Adversarial Domain Adaptation Inspired by the work of Goodfellow et al. (2014), adversarial networks have been used to learn domain invariant features in models involving multiple domains, through domain adversarial training (Ganin and Lempitsky, 2015; Ganin et al., 2016). Adversarial training facilitates domain-adaptation schemes, especially in high-resource to low-resource adaptation scenarios. The approach is based on an adversarial discriminator, which tries to identify the domain of the data, and backpropagates the negative gradients in the backward direction. This enables the model to learn shared domain features. Adversarial domain adaptation has been used in several NLP applications, including sentiment analysis (Chen et al., 2016), POS tagging for Twitter (Gui et al., 2017), relation extraction (Fu et al., 2017; Wang et al., 2018), among other applications. As far as we know, we are the first to apply adversarial domain adaptation in the context of dialectal morphological modeling.

Arabic Morphological Modeling Morphological modeling for Arabic has many contributions in both MSA (Diab et al., 2004; Habash and Rambow, 2005; Pasha et al., 2014; Abdelali et al., 2016; Khalifa et al., 2016), and Dialectal Arabic (Duh and Kirchoff, 2005; Al-Sabbagh and Girju, 2012; Habash et al., 2013). There were also several neural extensions that show impressive results (Zalmout and Habash, 2017; Zalmout et al., 2018). These contributions use separate models for each morphological feature, then apply a disambiguation step, similar to several previous models for Arabic (Habash and Rambow, 2005;

Pasha et al., 2014). Shen et al. (2016) use LSTMs with word/character embeddings for Arabic tagging. Darwish et al. (2018) use a CRF model for a multi-dialect POS tagging, using a small annotated Twitter corpus. Alharbi et al. (2018) also use neural models for Gulf Arabic, with good results.

4 Baseline Tagging and Disambiguation Architecture

In this section we present our baseline tagging and disambiguation architectures. We extend this architecture for joint modeling in the section that follows.

4.1 Morphological Feature Tagging

We use a similar tagging architecture to Zalmout et al. (2018), based on a Bi-LSTM tagging model, for the closed-set morphological features. Given a sentence of length L $\{w_1, w_2, \dots, w_L\}$, every word w_j is represented by vector \mathbf{v}_j . We use two LSTM layers to model the relevant context for each direction of the target word, using:

$$\vec{\mathbf{h}}_j = g(\mathbf{v}_j, \vec{\mathbf{h}}_{j-1})$$

$$\overleftarrow{\mathbf{h}}_j = g(\mathbf{v}_j, \overleftarrow{\mathbf{h}}_{j+1})$$

where \mathbf{h}_j is the context vector from the LSTM for each direction. We join both sides, apply a non-linearity function, output layer, and softmax for a probability distribution. The input vector \mathbf{v}_j is comprised of:

$$\mathbf{v}_j = [\mathbf{w}_j; \mathbf{s}_j; \mathbf{a}_j^f]$$

Where \mathbf{w}_j is the word embedding vector, \mathbf{s}_j is a vector representation of the characters within the word, and \mathbf{a}_j^f is a vector representing all the candidate morphological tags (from an analyzer), for feature f .

We pre-train the word embeddings with Word2Vec (Mikolov et al., 2013), using a large external dataset. For the character embeddings vector \mathbf{s}_j we use an LSTM-based architecture, applied to the character sequence in each word separately. We use the last state vector as the embedding representation of the word's characters.

The morphological feature vector \mathbf{a}_j^f embeds the candidate tags for each feature. We use a morphological analyzer to obtain all possible feature values of the word to be analyzed, embed the

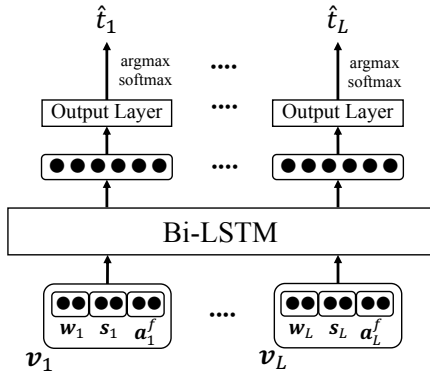


Figure 1: The overall tagging architecture, with the input vector as the concatenation of the word, characters, and candidate tag embeddings.

values using a feature-specific embedding tensor, then sum all the resulting vectors for each feature:

$$\mathbf{a}_j^f = \sum_{n=1}^{N_f} \mathbf{a}_{j,n}^f$$

Where N_f is the maximum number of possible candidate tags for the word j (from the analyzer), for feature f . We sum the vectors because the tags are alternatives, and do not constitute a sequence.

The \mathbf{a}_j^f vector does not constitute a hard constraint and can be discarded if a morphological analyzer is not used. Figure 1 shows the overall tagging architecture.

4.2 Lemmatization and Diacritization

The morphological features that are non-lexical, like POS, gender, number, among others, are handled by the model presented so far, using the multitask learning architecture. Lexical features, like lemmas and diacritized forms, on the other hand, are handled with neural language models, as presented by Zalmout and Habash (2017) and Zalmout et al. (2018). The lexical features are more difficult to model jointly with the non-lexical features, as they have large target spaces, and modeling them as classification tasks is not feasible.

4.3 Full Morphological Disambiguation

The predicted feature values for each word, whether from the tagger or the language models, can be returned directly if we do not use a morphological analyzer, without an explicit ranking step. If a morphological analyzer is used, the disambiguation system selects the optimal analysis for the word from the set of analyses re-

turned by the morphological analyzer. We use the predicted feature values from the taggers and language models to rank the analyses, and select the analysis with highest number of matched feature values. We also use weighted matching; where instead of assigning ones and zeros for the matched/mismatched features, we use a feature-specific matching weight. We replicate the morphological disambiguation pipeline presented in earlier contributions (Zalmout and Habash, 2017; Zalmout et al., 2018), and use the same parameter values and feature weights.

5 Multitask Learning Architecture

Most of the previous approaches for morphological tagging in Arabic learn a separate model for each morphological feature, and combine the predicted tags for disambiguation (Pasha et al., 2014; Zalmout and Habash, 2017; Zalmout et al., 2018). This hard separation eliminates any knowledge sharing among the different features when training and tagging. Joint learning, through parameter sharing in multitask learning, helps prune the space of target values for some morphological features, and reduce sparsity. The separation of the morphological models is also inefficient in terms of execution complexity. Training 14 different models, and running them all during runtime, is very wasteful in terms of execution time, memory footprint, and disk space.

Multitask learning is particularly useful in tasks with relatively complementary models, and usually involves primary and auxiliary tasks. We use multitask learning for joint training of the various morphological features. We extend the morphological tagging architecture presented at the previous section into a multitask learning model. We learn the different morphological features jointly through sharing the parameters of the hidden layers in the Bi-LSTM network. The input is also shared, through the word and character embeddings. We also use a unified feature-tags vector representation for all features, through concatenating the \mathbf{a}_j^f vectors for each feature of each word:

$$\mathbf{a}_j = [\mathbf{a}_j^{pos}; \dots; \mathbf{a}_j^{num}; \dots; \mathbf{a}_j^{vox}]$$

The output layer is separate for each morphological feature, with separate softmax and argmax operations. The loss function is the average of the individual feature losses, which are based on min-

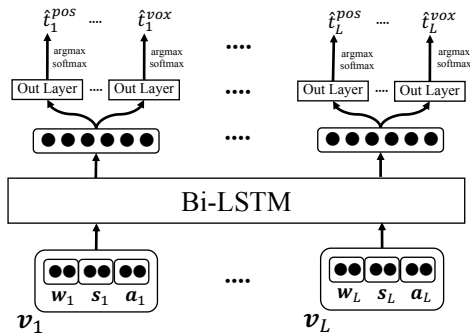


Figure 2: The multitask learning architecture, having separate output layers for each feature.

minimizing cross entropy H for each feature f :

$$H(\hat{T}, T) = \frac{1}{|F|} \sum_{f \in F} H(\hat{t}_f, t_f)$$

Where T represents the combined morphological tags for each word, and F is the set of features $\{pos, asp, \dots, vox\}$. Figure 2 shows the overall architecture for tagging using multitask learning.

6 Cross-Dialectal Model

Joint morphological modeling of high-resource and low-resource languages can be very beneficial as a knowledge-transfer scheme. Knowledge-transfer is more viable for languages that share linguistic similarities. In the context of DA, the linguistic similarities between MSA and the dialects, along with the MSA cognates common in DA, should allow for an efficient transfer model.

We train the model through dividing the datasets of each variant into batches, and running one variant-specific batch at a time. We introduce various extensions to the multitask learning architecture for cross-dialectal modeling. These include sharing the embeddings for the pretrained word embeddings and character embeddings, sharing the output layers for the different features, and adversarial training as a form of dialect adaptation. The decisions of shared vs joint modeling throughout the various architecture choices will also affect the size of the model and number of parameters.

6.1 Shared Embeddings

Pretrained embeddings have been shown to be very beneficial for several NLP tasks in Arabic (Zalmout and Habash, 2017; Erdmann et al., 2018; Watson et al., 2018). In the context of joint modeling of different variants, pretrained embeddings

can either be learnt separately or jointly, with several different configurations that include:

- Separate embedding spaces, through separate models for the different dialects, trained on separate datasets.
- Merged embedding datasets, by merging the datasets for the different dialects and train a single embedding model. This approach is viable because the different Arabic variants use the same script, and DA usually involves a lot of code-switching with MSA.
- Mapped embedding spaces, by training separate models for each dialect, then mapping the embedding spaces together.

We use VECMAP (Artetxe et al., 2016, 2017) to map the embedding spaces of the different variants (MSA and DA). VECMAP uses a seed dictionary to learn a mapping function that minimizes the distances between seed dictionary unigram pairs.

In addition to shared word embeddings, the character-level embeddings can also be learned separately or jointly. We do not use pretrained embeddings for the characters, and the embeddings are learnt as part of the end-to-end system.

6.2 Shared Output Layers

In the multitask learning architecture, each of the different morphological features needs a separate output layer. In our experiments with Arabic, we are modeling 14 morphological features, which requires 14 output layers. For cross-dialectal modeling, we can have separate output layers for each dialect, which results in 28 output layers for MSA and EGY. Another design choice in this case is to share the output layers between the different dialects, regardless of how many dialects are modeled jointly, with 14 shared output layers only.

Despite the morphological features being similar across the dialects, the target space for each feature might vary slightly for each dialect (as in proclitics and enclitics). In the case of shared output layers, we have to merge the target space values for the features of the different dialects, and use this combined set as the target vocabulary.

6.3 Adversarial Dialect Adaptation

Similar to adversarial domain adaptation, the goal of the *adversarial dialect adaptation* approach is to learn common features for the different dialects through an adversarial discriminator. Learning dialect-invariant features would facilitate a

richer knowledge-transfer scheme from the high-resource to the low-resource variants, since they are both modeled in the same invariant space. Adversarial adaptation can make use of a large annotated dataset from the high-resource dialect, unlabeled low-resource dialect data, and a small annotated low-resource dialect dataset. Adversarial adaptation learns dialect invariant features through backpropagating the negative gradients in the backward direction for the discriminator. The backward/forward propagation is managed by the Gradient Reversal Layer. Figure 3 shows the architecture with the discriminator task.

Gradient Reversal Layer Presented by Ganin and Lempitsky (2015), the gradient reversal layer (GRL) passes the identity function in the forward propagation, but negates the gradients it receives in backward propagation, i.e. $g(F(x)) = F(x)$ in forward propagation, but $\Delta g(F(x)) = -\lambda \Delta F(x)$ in backward propagation. λ is a weight parameter for the negative gradient, which can have an update schedule. λ is used to control the dissimilarity of features at the various stages of training. It can be small at the beginning of training to facilitate better morphological modeling, then increased to learn domain invariant features later on.

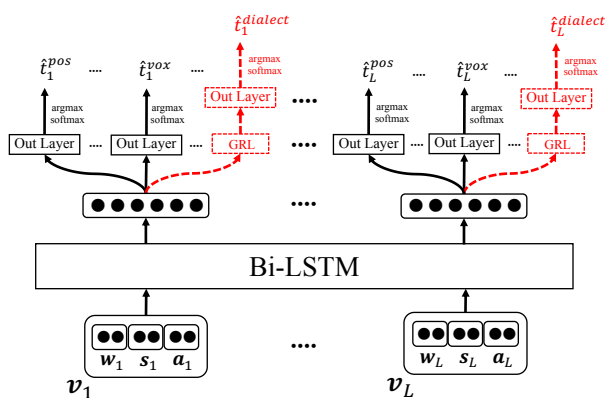


Figure 3: The adversarial adaptation architecture, with a discriminator task that backpropagates negative gradients using the Gradient Reversal Layer (GRL).

Training Process For each of the training batches, we populate half of the batch with samples from the morphologically labeled data, and the other half with the unlabeled data. The model calculates the morphological tagging loss for the first half, and the discriminator loss with the other, and optimizes for both jointly.

7 Experiments and Results

In this section we first discuss the datasets that we use, along with the experimental setup for the various experiments. We then discuss the results of the different models, using the full training datasets, and a learning curve over the EGY dataset, to simulate low-resource settings.

7.1 Data

Labeled Data For MSA we use the Penn Arabic Treebank (PATB parts 1, 2, and 3) (Maamouri et al., 2004). For EGY, we use the ARZ Treebank (ARZTB) annotated corpus from the Linguistic Data Consortium (LDC), parts 1, 2, 3, 4, and 5 (Maamouri et al., 2012). The annotation process and features are similar to those of MSA. We follow the data splits recommended by Diab et al. (2013) for training, development, and testing, for both MSA and EGY. Table 1 shows the data sizes. Throughout the different experiments in this paper, the DEV TEST dataset is used during the system development to assess design choices. The BLIND TEST dataset is used after finalizing the architecture, to evaluate the system and present the overall results. We use Alif/Ya and Hamza normalization, and we remove all diacritics (besides for lemmas and diacritized forms) for all variants.

	TRAIN	DEV TEST	BLIND TEST
MSA	503K	63K	63K
EGY	134K	21K	20K

Table 1: Word count statistics for MSA and EGY.

The morphological analyzers that we use include SAMA (Graff et al., 2009) for MSA, and a combination of SAMA, CALIMA (Habash et al., 2012), and ADAM (Salloum and Habash, 2014) for EGY, as used in the MADAMIRA (Pasha et al., 2014) system.

Unlabeled Data The pretrained word embeddings for MSA are trained using the LDC’s Gigaword corpus (Parker et al., 2011). For EGY we use about 410 million words of the Broad Operational Language Translation (BOLT) Arabic Forum Discussions (Tracey et al., 2018). We use the MADAR corpus (Bouamor et al., 2018) as the seed dictionary for embedding space mapping. We use the EGY data from the work by Zbib et al. (2012) as the unlabeled corpus for EGY.

7.2 Experimental Setup

Tagging Architecture We use two hidden layers of size 800 for the Bi-LSTM network (two for each direction), and a dropout wrapper with keep probability of 0.7, and peephole connections. We use Adam optimizer (Kingma and Ba, 2014) with a learning rate of 0.0005, and cross-entropy cost function. We run the various models for 70 epochs (fixed number of epoch since we use dropout). The LSTM character embedding architecture uses two LSTM layers of size 100, and embedding size 50. We use Word2Vec (Mikolov et al., 2013) to train the word embeddings. The embedding size is 250, and the embedding window is of size two.

Adversarial Adaptation For the adversarial adaptation experiments we first observed that the average sentence length in the unlabeled EGY dataset is very short compared to the MSA dataset (5 words per sentence for the unlabeled dataset, and 31 words per sentence for MSA). The difference in sentence length results in the unlabeled EGY dataset being four times the number of batches compared to MSA, for the same number of tokens, and the model was not converging. We therefore use a minimum sentence length of 14 words for the unlabeled dataset, which results in about 9K sentences (~185K tokens). We also found that a constant λ value of one performed better than scheduling the value starting from zero.

Metrics The evaluation metrics we use include:

- POS accuracy (POS): The accuracy of the POS tags, of a tagset comprised of 36 tags (Habash et al., 2013).
- The non-lexicalized morphological features accuracy (FEATS): The accuracy of the combined 14 closed morphological features.
- Lemmatization accuracy (LEMMA): The accuracy of the fully diacritized lemma.
- Diacritized forms accuracy (DIAC): The accuracy of the diacritized form of the words.
- Full Analysis Accuracy (FULL): The overall accuracy over the full analysis; FEATS (including POS)+LEMMA+DIAC, which is the strictest evaluation approach.

Baselines The baselines are based on separate models for the different features. The first baseline is MADAMIRA (Pasha et al., 2014), which is a popular morphological disambiguation tool

for Arabic. MADAMIRA uses SVM taggers for the different non-lexical features, and n-gram language models for the lemmas and diacritized forms. We also use the neural extensions of MADAMIRA (Zalmout and Habash, 2017; Zalmout et al., 2018), which are based on a similar architecture, but use LSTM taggers instead of the SVM models, and LSTM-based language models instead of the n-gram models.

7.3 Results

To evaluate the performance of the knowledge-transfer scheme, we present the results in two parts. The first presents the results for the full MSA and EGY datasets, evaluating the accuracy of the various architecture configurations. We then present the results of a learning curve over the size of the EGY training dataset, modeling various degrees of low-resource performance. The goal is to assess the multitask learning and adversarial training models in particular, and the degree of knowledge-transfer, which should be more helpful when the size of the EGY training data is lower.

7.3.1 Joint Morphological Modeling

Table 2 shows the results of the joint modeling of MSA and EGY. Based on the results, we make the following observations:

Multi-Feature Modeling The results for the multi-feature models show consistent and significant improvement compared to the separate models for each feature, especially for MSA. This supports the assumption that multi-feature modeling can identify more complex patterns involving multiple features, that separate models cannot.

Cross-Dialectal Modeling: Merged Training Data vs Multitask Learning For the cross-dialectal MSA and EGY models, we first experiment with merging the training datasets for both, and train a single model over the merged datasets. This model is a simple baseline for the cross-dialectal models, but imposes hard joint modeling that might lead to some knowledge loss.

The results indicate that the multitask learning architecture performs much better, especially for MSA. The accuracy for POS tagging for EGY in particular was higher or similar though. This is probably because POS behaves very similarly in both MSA and EGY, unlike other morphological features that might converge slightly. So the added MSA training samples were generally helpful.

MODEL	DEV TEST					BLIND TEST				
	FULL	FEATS	DIAC	LEX	POS	FULL	FEATS	DIAC	LEX	POS
MADAMIRA _{MSA} (Pasha et al., 2014)	85.6	87.1	87.7	96.3	97.1	85.6	87.3	87.6	96.3	97.0
MSA _{separate features} (Zalmout and Habash, 2017)	90.4	92.3	92.4	96.9	97.9	90.1	92.3	92.1	96.6	97.8
MSA _{MTL:MSA}	90.8	92.7	92.7	96.9	97.9	90.8	93.0	92.5	96.7	97.9
MSA _{MSA+EGY} merged training datasets	90.1	91.9	91.8	96.9	97.8	89.8	92.0	91.4	96.5	97.7
MSA _{MTL:MSA+EGY} mapped embedding spaces	90.6	92.5	92.4	96.8	97.8	90.3	92.5	91.9	96.5	97.7
MSA _{MTL:MSA+EGY} merged embedding corpora	91.1	93.0	92.9	96.9	97.9	91.0	93.2	92.6	96.7	98.0
MSA _{MTL:MSA+EGY} separate embedding spaces	91.2	93.1	92.9	97.0	98.0	91.1	93.3	92.7	96.7	98.0
+ shared output layers per feature	91.4	93.3	93.1	97.0	98.0	91.2	93.4	92.8	96.8	98.0
+ shared character embeddings	91.2	93.1	93.0	97.0	98.0	91.1	93.3	92.7	96.7	97.9
MSA _{MTL:MSA+EGY} Adversarial Dialect Adaptation*	91.3	93.2	93.0	97.0	98.0	91.2	93.3	92.8	96.7	97.9
MADAMIRA _{EGY} (Pasha et al., 2014)	76.2	86.7	82.4	86.4	91.7	77.3	86.9	83.3	87.3	91.8
EGY _{separate features} (Zalmout et al., 2018)	77.0	88.8	82.9	87.6	92.9	78.0	88.8	83.6	87.8	93.3
EGY _{MTL:EGY}	77.2	88.8	82.9	87.6	93.1	78.1	88.8	83.5	88.0	93.4
EGY _{MSA+EGY} merged training datasets	77.1	88.9	82.7	87.6	93.5	78.2	89.0	83.5	88.0	93.8
EGY _{MTL:MSA+EGY} mapped embedding spaces	76.7	88.3	82.6	87.3	92.7	78.0	88.6	83.3	87.8	93.3
EGY _{MTL:MSA+EGY} merged embedding corpora	77.2	89.0	82.9	87.7	93.1	78.1	88.9	83.5	88.0	93.5
EGY _{MTL:MSA+EGY} separate embedding spaces	77.3	89.0	83.0	87.7	93.1	78.4	89.2	83.7	88.0	93.6
+ shared output layers per feature	77.4	89.1	83.0	87.7	93.2	78.5	89.3	83.8	88.0	93.7
+ shared character embeddings	77.3	89.0	82.9	87.7	93.2	78.2	89.1	83.6	88.1	93.7
EGY _{MTL:MSA+EGY} Adversarial Dialect Adaptation*	77.5	89.3	83.1	87.7	93.3	78.6	89.4	83.8	88.1	93.8

Table 2: Disambiguation results for joint MSA and EGY modeling. MTL is Multitask Learning. *Best adversarial result was with merged embedding spaces.

Embedding Models Joint embedding spaces between the dialects, whether through embedding space mapping or through learning the embeddings on the combined corpus, did not perform well. Using separate embedding models (whether for word or character embeddings) for each dialect shows better accuracy. Embedding models learn properties and morphosyntactic structures that are specific to the training data. Mapping the embedding spaces likely results in some knowledge loss. Unlike the adversarial training model though, at which the merged embedding datasets model performed better. This is expected since the goal of adversarial training is to bring the overall feature spaces closer to learn dialect-invariant features.

Shared Output Layers The results indicate that using shared output layers for the different dialects improves the overall accuracy. Shared output layers are more likely to learn shared morphosyntactic structures from the other dialect, thus helping both. Having separate layers wastes another joint learning potential. The shared output layers further reduce the size of the overall model.

Adversarial Dialect Adaptation The adversarial adaptation experiments show slightly higher results for EGY, but very close results to the multitask learning model for MSA. Since MSA is resource-rich it is expected that adversarial training would not be beneficial (or even hurtful), as the

dialect-invariant features would hinder the full utilization of the rich MSA resources. For EGY, we expect that the knowledge-transfer model would be more beneficial in lower-resource scenarios, we therefore experiment with a learning curve for the training dataset size in the next section.

7.3.2 Modeling Training Data Scarcity

EGY TRAIN SIZE	EGY	MSA-EGY	
		MTL	ADV
2K (1.5%)	29.7	61.9	71.1
8K (6%)	62.5	73.5	78.3
16K (12%)	74.7	78.1	81.5
33K (25%)	80.7	81.6	83.5
67K (50%)	83.3	82.0	84.0
134K (100%)	84.5	85.4	85.6

Table 3: The results (FEATS) of the learning curve over the EGY training dataset, for the EGY dataset alone, multitask learning (MTL), and the adversarial training (ADV). We do not use morphological analyzers here, so the results are not comparable to Table 2.

Knowledge-transfer schemes are more valuable in low-resource settings for the target language. To simulate the behavior of the multitask and adversarial learning architectures in such setting, we train the model using fractions of the EGY training data. We reduce the training dataset size by a factor of two each time. We then simulate extreme scarcity, having only 2K EGY annotated tokens.

Low-resource dialects will have very limited

or no morphological analyzers, so we also simulate the lack of morphological analyzers for EGY. Since we are not using an EGY morphological analyzer, we evaluate the models on the set of non-lexicalized and clitics features only, without the diacritized forms and lemmas. We also do not perform an explicit disambiguation step through analysis ranking, and we evaluate on the combined morphological tags directly for each word.

Table 3 shows the results. Multitask learning with MSA consistently outperforms the models that use EGY data only. The accuracy almost doubles in the 2K model. We also notice that the accuracy gap increases as the EGY training dataset size decreases, highlighting the importance of joint modeling with MSA in low-resource DA settings. The adversarial adaptation results in the learning curve further show a significant increase in accuracy with decreasing training data size, compared to the multitask learning results. The model seems to be facilitating more efficient knowledge-transfer, especially for the lower-resource EGY experiments. We can also observe that for the extreme low-resource setting, we can double the accuracy through adversarial multitask learning, achieving about 58% relative error reduction.

The results also indicate that with only 2K EGY annotated tokens, and with adversarial multitask learning with MSA, we can achieve almost the same accuracy as 16K tokens using EGY only. This is a significant result, especially when commissioning new annotation tasks for other dialects.

Error Analysis We investigated the results in the learning curve to understand the specific areas of improvement with multitask learning and adversarial training. We calculated the accuracies of each of the features, for both models, and across all the dataset sizes. We observed that the POS and Gender features benefited the most of the joint modeling techniques. Whereas features like Mood and Voice benefited the least. This is probably due to the relatively similar linguistic behavior for POS and Gender in both MSA and EGY, unlike Mood or Voice, which are less relevant to DA, and can be somewhat inconsistent with MSA. The improvement was consistent for both approaches, and across the training data sizes, with POS having almost 61% relative error reduction in the 2K dataset with adversarial training, and Mood (the least improving feature) of about 8%. And 8% for POS, and 0% for Mood, in the full size dataset.

8 Conclusions and Future Work

In this paper we presented a model for joint morphological modeling of the features in morphologically rich dialectal variants. We also presented several extensions for cross-dialectal modeling. We showed that having separate embedding models, but shared output layers, performs the best. Joint modeling for the features within each dialect performs consistently better than having separate models, and joint cross-dialectal modeling performs better than dialect-specific models. We also used adversarial training to facilitate a knowledge-transfer scheme, providing the best result for EGY, especially in lower-resource cases. Our models result in state-of-the-art results for both MSA, and EGY. Future work includes joint and cross-dialectal lemmatization models, in addition to further extension to other dialects.

Acknowledgment The first author was supported by the New York University Abu Dhabi Global PhD Student Fellowship program. The support and resources from the High Performance Computing Center at New York University Abu Dhabi are also gratefully acknowledged.

References

- 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, San Diego, California.
- Roei Aharoni, Yoav Goldberg, and Yonatan Belinkov. 2016. Improving sequence to sequence learning for morphological inflection generation: The biunit systems for the sigmorphon 2016 shared task for morphological reinflection. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 41–48.
- Rania Al-Sabbagh and Roxana Girju. 2012. A supervised POS tagger for written Arabic social networking corpora. In *Proceedings of KONVENS 2012*, pages 39–52. OGAI. Main track: oral presentations.
- Randah Alharbi, Walid Magdy, Kareem Darwish, Ahmed Abdelali, and Hamdy Mubarak. 2018. Part-of-Speech Tagging for Arabic Gulf Dialect Using Bi-LSTM. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France. European Language Resources Association (ELRA).
- Héctor Martínez Alonso and Barbara Plank. 2017. When is multitask learning effective? semantic sequence prediction under varying data conditions. In

- 15th Conference of the European Chapter of the Association for Computational Linguistics, pages 44–53.
- Waleed Ammar, George Mulcaire, Miguel Ballesteros, Chris Dyer, and Noah A Smith. 2016. Many languages, one parser. *Transactions of the Association for Computational Linguistics*, 4:431–444.
- Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2016. Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *EMNLP*, pages 2289–2294.
- Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 451–462.
- Joachim Bingel and Anders Søgaard. 2017. [Identifying beneficial task relations for multi-task learning in deep neural networks](#). In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 164–169, Valencia, Spain. Association for Computational Linguistics.
- Houda Bouamor, Nizar Habash, Mohammad Salameh, Wajdi Zaghouni, Owen Rambow, Dana Abdulrahim, Ossama Obeid, Salam Khalifa, Fadhl Eryani, Alexander Erdmann, and Kemal Oflazer. 2018. The MADAR Arabic dialect corpus and lexicon. In *The International Conference on Language Resources and Evaluation*, Miyazaki, Japan.
- Jan Buys and Jan A Botha. 2016. Cross-lingual morphological tagging for low-resource languages. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1954–1964.
- Xilun Chen, Yu Sun, Ben Athiwaratkun, Claire Cardie, and Kilian Weinberger. 2016. [Adversarial deep averaging networks for cross-lingual sentiment classification](#). *ArXiv e-prints*.
- Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuska. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.
- Ryan Cotterell and Georg Heigold. 2017. [Cross-lingual character-level neural morphological tagging](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 748–759, Copenhagen, Denmark. Association for Computational Linguistics.
- Kareem Darwish, Hamdy Mubarak, Ahmed Abdellali, Mohamed Eldesouki, Younes Samih, Randah Alharbi, Mohammed Attia, Walid Magdy, and Laura Kallmeyer. 2018. Multi-Dialect Arabic POS Tagging: A CRF Approach. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France. European Language Resources Association (ELRA).
- Mona Diab, Nizar Habash, Owen Rambow, and Ryan Roth. 2013. LDC Arabic treebanks and associated corpora: Data divisions manual. *arXiv preprint arXiv:1309.5652*.
- Mona Diab, Kadri Hacioglu, and Daniel Jurafsky. 2004. Automatic Tagging of Arabic Text: From Raw Text to Base Phrase Chunks. In *Proceedings of the 5th Meeting of the North American Chapter of the Association for Computational Linguistics/Human Language Technologies Conference (HLT-NAACL04)*, pages 149–152, Boston, MA.
- Kevin Duh and Katrin Kirchhoff. 2005. [POS tagging of dialectal Arabic: a minimally supervised approach](#). In *Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages*, Semitic ’05, pages 55–62, Ann Arbor, Michigan.
- Alexander Erdmann, Nasser Zalmout, and Nizar Habash. 2018. [Addressing noise in multidialectal word embeddings](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 558–565, Melbourne, Australia. Association for Computational Linguistics.
- Manaaf Faruqui, Yulia Tsvetkov, Graham Neubig, and Chris Dyer. 2016. Morphological inflection generation using character sequence to sequence learning. In *Proceedings of NAACL-HLT*, pages 634–643.
- Lisheng Fu, Thien Huu Nguyen, Bonan Min, and Ralph Grishman. 2017. [Domain adaptation for relation extraction with domain adversarial neural network](#). In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 425–429. Asian Federation of Natural Language Processing.
- Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning-Volume 37*, pages 1180–1189. JMLR. org.
- Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. [Domain-adversarial training of neural networks](#). *J. Mach. Learn. Res.*, 17(1):2096–2030.
- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. [Generative adversarial nets](#). In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc.
- David Graff, Mohamed Maamouri, Basma Bouziri, Sondos Krouna, Seth Kulick, and Tim Buckwalter. 2009. Standard Arabic Morphological Analyzer (SAMA) Version 3.1. Linguistic Data Consortium LDC2009E73.
- Tao Gui, Qi Zhang, Haoran Huang, Minlong Peng, and Xuanjing Huang. 2017. [Part-of-speech tagging for](#)

- twitter with adversarial neural networks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2411–2420. Association for Computational Linguistics.
- Jiang Guo, Wanxiang Che, David Yarowsky, Haifeng Wang, and Ting Liu. 2015. Cross-lingual dependency parsing based on distributed representations. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 1234–1244.
- Nizar Habash, Fadhil Eryani, Salam Khalifa, Owen Rambow, Dana Abdulrahim, Alexander Erdmann, Reem Faraj, Wajdi Zaghouni, Houda Bouamor, Nasser Zalmout, Sara Hassan, Faisal Al-Shargi, Sakhar Alkhereyf, Basma Abdulkareem, Ramy Eskander, Mohammad Salameh, and Hind Saddiki. 2018. [Unified guidelines and resources for Arabic dialect orthography](#). In *Proceedings of the 11th Language Resources and Evaluation Conference*, Miyazaki, Japan. European Language Resource Association.
- Nizar Habash, Ramy Eskander, and Abdelati Hawwari. 2012. A Morphological Analyzer for Egyptian Arabic. In *Proceedings of the Twelfth Meeting of the Special Interest Group on Computational Morphology and Phonology*, pages 1–9, Montréal, Canada.
- Nizar Habash and Owen Rambow. 2005. Arabic Tokenization, Part-of-Speech Tagging and Morphological Disambiguation in One Fell Swoop. In *Proceedings of the 43rd Annual Meeting of the ACL*, pages 573–580, Ann Arbor, Michigan.
- Nizar Habash, Ryan Roth, Owen Rambow, Ramy Eskander, and Nadi Tomeh. 2013. [Morphological Analysis and Disambiguation for Dialectal Arabic](#). In *Proceedings of NAACL-HLT*, pages 426–432, Atlanta, Georgia.
- Nizar Y Habash. 2010. *Introduction to Arabic natural language processing*, volume 3. Morgan & Claypool Publishers.
- Kazuma Hashimoto, Yoshimasa Tsuruoka, Richard Socher, et al. 2017. A joint many-task model: Growing a neural network for multiple nlp tasks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1923–1933.
- Go Inoue, Hiroyuki Shindo, and Yuji Matsumoto. 2017. Joint prediction of morphosyntactic categories for fine-grained Arabic part-of-speech tagging exploiting tag dictionary information. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 421–431.
- Katharina Kann, Ryan Cotterell, and Hinrich Schütze. 2017. One-shot neural cross-lingual transfer for paradigm completion. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1993–2003.
- Salam Khalifa, Nasser Zalmout, and Nizar Habash. 2016. YAMAMA: Yet Another Multi-Dialect Arabic Morphological Analyzer. In *Proceedings of the International Conference on Computational Linguistics (COLING): System Demonstrations*, pages 223–227.
- Diederik P. Kingma and Jimmy Ba. 2014. [Adam: A method for stochastic optimization](#). *CoRR*, abs/1412.6980.
- Mohamed Maamouri, Ann Bies, Tim Buckwalter, and Wigdan Mekki. 2004. The Penn Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus. In *NEMLAR Conference on Arabic Language Resources and Tools*, pages 102–109, Cairo, Egypt.
- Mohamed Maamouri, Sondos Krouna, Dalila Tabessi, Nadia Hamrouni, and Nizar Habash. 2012. Egyptian Arabic Morphological Annotation Guidelines.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.
- Robert Parker, David Graff, Ke Chen, Junbo Kong, and Kazuaki Maeda. 2011. Arabic Gigaword Fifth Edition. LDC catalog number No. LDC2011T11, ISBN 1-58563-595-2.
- Arfath Pasha, Mohamed Al-Badrashiny, Ahmed El Kholy, Ramy Eskander, Mona Diab, Nizar Habash, Manoj Pooleery, Owen Rambow, and Ryan Roth. 2014. MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. In *In Proceedings of LREC*, Reykjavik, Iceland.
- Wael Salloum and Nizar Habash. 2014. ADAM: Analyzer for Dialectal Arabic Morphology. *Journal of King Saud University-Computer and Information Sciences*, 26(4):372–378.
- Qinlan Shen, Daniel Clothiaux, Emily Tagtow, Patrick Littell, and Chris Dyer. 2016. [The role of context in neural morphological disambiguation](#). In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 181–191, Osaka, Japan. The COLING 2016 Organizing Committee.
- Anders Søgaard and Yoav Goldberg. 2016. [Deep multi-task learning with low level tasks supervised at lower layers](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 231–235, Berlin, Germany. Association for Computational Linguistics.
- Jennifer Tracey, Haejoong Lee, Stephanie Strassel, and Safa Ismael. 2018. BOLT Arabic Discussion Forum Source Data. LDC catalog number LDC2018T10.
- Xiaozhi Wang, Xu Han, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2018. [Adversarial multi-lingual neural relation extraction](#). In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1156–1166. Association for Computational Linguistics.

- Daniel Watson, Nasser Zalmout, and Nizar Habash. 2018. Utilizing character and word embeddings for text normalization with sequence-to-sequence models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 837–843.
- Nasser Zalmout, Alexander Erdmann, and Nizar Habash. 2018. Noise-Robust Morphological Disambiguation for Dialectal Arabic. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.
- Nasser Zalmout and Nizar Habash. 2017. [Don't Throw Those Morphological Analyzers Away Just Yet: Neural Morphological Disambiguation for Arabic](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 704–713, Copenhagen, Denmark. Association for Computational Linguistics.
- Rabih Zbib, Erika Malchiodi, Jacob Devlin, David Stallard, Spyros Matsoukas, Richard Schwartz, John Makhoul, Omar F Zaidan, and Chris Callison-Burch. 2012. Machine translation of Arabic dialects. In *Proceedings of the 2012 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 49–59. Association for Computational Linguistics.