Out-of-Domain Dependency Parsing for Dialects of Arabic: A Case Study

Noor Abo Mokh*, Daniel Dakota, Sandra Kübler
Indiana University
{noorabom,ddakota,skuebler}@iu.edu

Abstract

We study dependency parsing for four Arabic dialects (Gulf, Levantine, Egyptian, and Maghrebi). Since no syntactically annotated data exist for Arabic dialects, we train the parser on a Modern Standard Arabic (MSA) corpus, which creates an out-of-domain setting. We investigate methods to close the gap between the source (MSA) and target data (dialects), e.g., by training on syntactically similar sentences to the test data. For testing, we manually annotate a small data set from a dialectal corpus. We focus on parsing two linguistic phenomena, which are difficult to parse: Idafa and coordination. We find that we can improve results by adding in-domain MSA data while adding dialectal embeddings only results in minor improvements.

1 Introduction

Syntactic parsing for Arabic is challenging, mostly because of its morphological complexity. Parsing dialectal Arabic is more complex due in part to the lack of syntactically annotated corpora. Consequently, this task has not been as extensively studied. The available treebanks for Arabic, such as CATiB (Habash and Roth, 2009), CAMEL Treebank (Habash et al., 2022) and PATB (Maamouri et al., 2004), are all in MSA. MSA is used for formal written communications and in news, while dialectal Arabic (DA) is used for all other communication forms. Because of the lack of dialectal treebanks, the most common scenario is training on MSA and testing on dialectal data, without quantitative means of assessing performance. Training on MSA treebanks does not only mean differences in dialects, but also in text domains, further complicating the issue. In such scenario, the domain shifts result in a drop in performance due to differences in properties of the data (e.g., Gildea, 2001). We group both types of differences under domain differences, since the current data situation does not allow us to separate the two.

Given that syntactically annotating a treebank for dialectal Arabic requires an extensive amount of manual work, we decided to focus on two syntactic constructions, which are known to be difficult to parse and that show significant differences across the dialects: Idafa and coordination relations (Green and Manning, 2010).[1] We assume that if we can improve parsing quality for these difficult constructions, parsing quality overall should also increase, especially since most of our modifications to handle the out-of-domain scenario are not specifically geared towards handling these phenomena but are rather general.

We investigate methods to improve dependency parsing of dialectal Arabic in an out-of-domain setting, when there is no available training data that matches the test data in both domain and dialect. We experiment with a selection of similarity criteria to adapt our source to the target data to improve performance. For evaluation, we annotate test sentences for four dialects of Arabic, covering the phenomena listed above.

1.1 Linguistic Phenomena

We focus on two linguistic phenomena that are challenging to parse: Idafa and (phrasal) coordination. For our test data, we use sentences that have phrases with Idafa or coordination (regardless of other syntactic phenomena). For more detailed information on the test data, see section 3.1.

Idafa is a type of possessive in Arabic where two or more nouns are combined (as an Idafa chain). بيت العَائِلِة (Eng.: the family house) is an Idafa example with two nouns in a sequence. In such a construction, the first noun (the head of the construction) is possessed by the following noun (Habash,

[*] The work was done prior to joining Amazon.

[1] Green and Manning (2010) focus on constituency parsing, but we are unaware of any dependency parsing work that examines specifically Idafa and coordination for Arabic.

Proceedings of The Second Arabic Natural Language Processing Conference, pages 170–182
August 16, 2024 ©2024 Association for Computational Linguistics

2010). While Idafa in MSA has a standard form, in dialectal Arabic, it can either be realized similar to MSA or using a preposition to indicate the possessive relation, depending on the dialect. See examples (1)–(5) for the different realizations of Idafa (the examples are taken from MADAR (Bouamor et al., 2019)).

(1) MSA: مفتَاح البلَاد
det-city-pl key-sg
Eng: the key of the city

(2) Egyptian:
a. كود البلد
det-city-sg key-sg
b. كود بتَاع البلد
det-city-sg *prep* key-sg

(3) Maghrebi:
a. مفتَاح البلَاد
det-city-pl key-sg
b. مفتَاح تع البلَاد
det-city-pl *prep* key-sg

(4) Gulf:
a. الكود البلد
det-city-sg key-sg
b. كود مَال البلد
det-city-sg *prep* key-sg

(5) Levantine
a. مفتَاح البلد
det-city-sg key-sg
b. مفتَاح تبعي البلد
det-city-sg *prep* key-sg

The different realizations of Idafa in dialects are not well-documented in the literature. In this study, we make our decisions based on the parallelism of sentences across dialects. In the studied dialects, Egyptian, Maghrebi, Gulf and Levantine, prepositions such as بتَاع, تع, حق, مَال, تبعي, تبع are normally used for possessives. For instance, example (2) shows the two Idafa representations for Egyptian. In the prepositional Idafa in example (2-b), the preposition بتَاع connects the two nouns.

The second phenomenon we chose is (phrasal) coordination. Coordination is marked in all Arabic dialects with ‘wa’ and its phonological variants. We are aware that there are other variants used for coordination, however, in this study, we use ‘wa’ plus its dialectal variants to choose coordination sentences, since it is the most frequent conjunction in the MSA part of MADAR (Bouamor et al., 2018) (we use dialectal MADAR sentences for our test data).

1.2 Research Questions

We address the following research questions:

1. Can we improve parsing accuracy for dialectal data by using training sentences of similar length?

2. How does sentential coordination, which is frequent in MSA but does not occur in dialectal data, impact automatic syntactic analysis? Does removing such sentences from training data help disambiguate coordination phrases?

3. Can we improve parsing accuracy for dialectal data by using training sentences that are syntactically similar between the training and test set? Can we determine similarity by using perplexity?

4. Is the use of dialectal embeddings beneficial when parsing out-of-domain data?

5. Does adding in-domain data, i.e., the MSA portion of the parallel corpus, to the training data improve parsing accuracy for dialectal data?

The rest of the paper is organized as follows: section 2 describes related work. In section 3, we provide an overview of the experiments. We present our results in section 4, followed by an error analysis in section 6. In the last section, we provide a summary and conclusion.

2 Related Work

2.1 Arabic Dependency Parsing

The inclusion of Arabic in the CoNLL2007 Shared Task (Nivre et al., 2007a) helped facilitate research in Arabic dependency parsing. Early work by Marton et al. (2010, 2013) examined the importance of morphological features in both gold and predicted conditions on CATiB using MaltParser (Nivre et al., 2007b). They note that due to prediction difficulty, certain features, such as person and number, are more helpful under predicted conditions, while other features, such as case, are more beneficial when having access to gold information.

This is in line with work by Mohamed (2011), who shows how minor errors in Arabic upstream predictions contribute to substantial degradation of parsing performance. To help optimize for features, CamelParser1.0 (Shahrour et al., 2016) modifies MaltParser into a two-stage optimization process, first creating a base parsing model before selecting for morphological features, which improves results over the base parsing model.

Taji et al. (2017) use MaltParser to compare parsing results between Universal Dependency (de Marneffe et al., 2021) and CATiB annotations on the same treebank. They find the latter to achieve higher performance, potentially due to the substantially lower number of dependency labels. A multi-task parsing setup is explored by Kankanampati et al. (2020) using a Multidimensional Easy First parser (Constant et al., 2016) to leverage both the UD and CATiB treebank representations against each other, finding that error reduction was driven by improvements on partial dependencies. The recent release of CamelParser2.0 (Elshabrawy et al., 2023) uses a version of the Dozat and Manning (2017) parser and incorporates various BERT embeddings, achieving improved performance on available UD treebanks and CATiB over other available Arabic parsing models.

2.2 Domain Adaptation for Dependency Parsing

Domain adaptation for dependency parsing has traditionally focused on selecting optimal source data for the target domain (Plank and van Noord, 2011; McDonald et al., 2011; Mukherjee and Kübler, 2018) using both lexicalized (Falenska and Çetinoğlu, 2017) and delexicalized methods (Rosa and Žabokrtský, 2015).

More recent approaches have focused on sharing of information via embeddings (Li et al., 2019, 2020; Stymne, 2020), yielding improvements across domains, as has the addition of loss weighting in multi-task architectures to compensate for data imbalances in source and target (Dakota et al., 2021). While domain specific embedding models can certainly benefit a target domain (Liu et al., 2020), diversifying the data used to generate the underlying embeddings enables more generalizability across tasks and domains (Martin et al., 2020; Virtanen et al., 2019; Inoue et al., 2021).

2.3 NLP for Arabic Dialects

There has been an increased need to develop more resources for dialectal Arabic (Darwish et al., 2021). Prior work looks at building sources for multi-dialect studies (Al-Sabbagh and Girju, 2012; Abdul-Mageed et al., 2014), specific Arabic dialects (Seddah et al., 2020; Gugliotta et al., 2023), and Classical Arabic (Al-Ghamdi et al., 2021).

Research has often focused on building or leveraging cross-dialect models. Difficulties in Arabic dialect identification stem from overlap and code-switching (Abdelali et al., 2021) given a lack of standardization within dialects. Early constituency parsing work (Chiang et al., 2006) highlights the difficulties of leveraging MSA on transcriptions of spoken Levantine. Attia and Elkahky (2019) show how normalization across dialects helps reduce data sparsity and improves POS tagging. Both dialect familiarity (Abu Farha and Magdy, 2022) and annotation inconsistency between dialects (Abo Mokh et al., 2022) have shown to be impediments for effective cross-dialect modeling.

3 Experimental Setup

3.1 Data

The main challenge of this study is the lack of syntactically annotated data for dialectal Arabic. The only available Arabic resources to train a parser are in MSA, e.g. CATiB (Habash and Roth, 2009) and Penn Treebank (Maamouri et al., 2004), with a small set of Egyptian Arabic available in CATiB (Habash and Roth, 2009). Several dialectal corpora have been published in the last decade: Habibi corpus (El-Haj, 2020), MADAR (Bouamor et al., 2018), PADIC (Meftouh et al., 2015), and Curras+baladi (Al-Haff et al., 2022). While there are a few dialectal datasets with POS annotations (e.g. the one by Darwish et al. (2018)), to the best of our knowledge, no dialectal corpus with syntactic annotations exists. This means that we are limited to using MSA data for training, thus requiring us to parse out-of-domain.

CATiB For training, we use the Columbia Arabic Treebank (CATiB; Habash and Roth, 2009), which uses dependency grammatical representations, different from Universal Dependencies (UD) (de Marneffe et al., 2021). While UD uses a schema that unifies 100 languages, CATiB uses an Arabic traditional syntax-inspired schema.

CATiB uses a small POS tagset[2], plus eight dependency relation labels: SBJ, OBJ, TPC (topic in complex nominal sentences), PRD (predicate), IDF (Idafa), TMZ (Tamyeez), MOD (a modifier of verbs or nouns).

CATiB is based on MSA texts from different news feeds collected from multiple news agencies and newspapers. The training portion includes 15 747 sentences, and the test portion consists of 1 959 sentences.

MADAR For testing, we utilize the available dialectal corpus MADAR (Bouamor et al., 2018). The corpus consists of translations of the (English) Basic Traveling Expression Corpus (BTEC) by Takezawa et al. (2007). MADAR is a collection of parallel sentences from different dialects that represent the Arabic varieties of 25 cities[3] in addition to MSA. We follow the traditional grouping of dialects as described by Darwish et al. (2017): Egyptian, Gulf, Levantine, and Maghrebi according to geographical location.

From MADAR, we also use the MSA part (2 000 sentences) that is syntactically annotated (Habash et al., 2022) as an (additional) in-domain training data. The syntactic annotations are consistent with those in CATiB. These sentences are from the same textual domain as the dialectal test sentences, but repesent a different dialect.

3.2 Test Data: Selection and Annotation

As described above, we focus on two phenomena: Idafa and coordination. We randomly extract Idafa and coordination sentences from the MADAR corpus, 100 sentences per phenomenon per dialect. We verify that the sentences per dialect are variants of the same sentence in the set, and are renditions of similar sentences in multiple dialects. This results in a total of 800 parallel sentences.

For Idafa, we choose sentences with sequences of two or more nouns. For coordination, we limit our corpus to sentences using 'wa' as the coordinating conjunction.

We manually annotate the selected sentences with dependency annotations following CATiB guidelines (see section 3.1 for information on the labels). Of the eight dependency relations in CATiB, TPC and TMZ were not used, since these are relations specific to MSA. As for PRD, which may occur in dialects, in MADAR, it occurred very infrequently.

Treebank	Train	Test
CATiB	15762	1959
CATiB Max 15	2663	
CATiB No initial 'wa'	7180	
CATiB Gulf Perplexity	1999	
CATiB Levantine Perplexity	1999	
CATiB Egyptian Perplexity	1998	
CATiB Maghrebi Perplexity	1998	
MADAR MSA	1600	200

Table 1: Treebank statistics.

We had to make several decisions wrt. using the CATiB annotation scheme for dialectal data. First, all dialects use standard Idafa and prepositional phrases to express a possessive relation. We decided to annotate such prepositional phrases as Idafa (similar to examples (1)–(5)) since the parallel structure of the corpus allows us to make this distinction. This construction (using prepositions) is not represented in CATiB. Another decision concerns the treatment of multiple roots in Arabic. CATiB annotates multiple roots as '—' when there is more than one full clause in a sentence. We follow this convention to ensure consistency between training and test data.

Table 1 gives an overview of training and test set sizes, including the training data variants based on our adaptation strategies (see section 4).

3.3 Parser

For parsing, we use an implementation of the Dozat and Manning (2017) parser[4]. The parser is a neural graph-based dependency parser, which uses biaffine attention and a biaffine classifier in combination with dimension reducing MLP layers to reduce non-relevant information. We slightly modify the standard architecture to allow a setting that additionally permits using only BERT-based embeddings as input[5]. We experiment with two different Arabic BERT-based embeddings: `asafaya/bert-base-arabic` (araBERT) (Safaya et al., 2020) and `CAMeL-Lab/bert-base-`

[2] NOM (nouns, pronouns, adjectives, and adverbs), PROP (proper nouns), VRB (active voice verbs), VRBPASS (passive voice verbs), PRT (particles) and PNX (punctuation)

[3] The following cities are covered: Aleppo, Alexandria, Algiers, Amman, Aswan, Baghdad, Basra, Beirut, Benghazi, Cairo, Damascus, Doha, Fes, Jeddah, Jerusalem, Khartoum, Mosul, Moscut, Rabat, Riyadh, Sanaa, Salt, Sfax, Tripoli, Tunis.

[4] `https://github.com/yzhangcs/parser`

[5] The default behavior or the parser is to concatenate word embeddings to additional feature embeddings.

`arabic-camelbert-mix` (Inoue et al., 2021). The former are trained on mostly MSA sources, while the latter used a mixture of MSA, Classical Arabic, and dialectal Arabic sources. We also experiment with concatenating additional character (`+char`) embeddings with the BERT embeddings[6]. All experiments, unless noted otherwise, use araBERT, since araBERT corresponds to the training data. Additionally, these embeddings were generated using the average embedding representations over all subtokens which has been indicated to be the most effective in domain adaptation settings (Dakota and Kübler, 2024).

3.3.1 POS Tagging

For the experiments using similarity to choose training sentences, we need to POS tag the data. We use CAMelTools POS tagger (Obeid et al., 2020), which is part of the morphological analyzer included in CAMeLTools. CAMeLTools uses databases for MSA, Egyptian, and Gulf dialects. It uses 29 POS tags[7]. Since the CAMelTools POS tagger uses a different tagset from CATiB, we map the CAMelTools tags to CATiB tags. This was a strict many-to-one mapping, so we performed a rule-based mapping of the tagger output.

3.4 Calculating Perplexity

We use perplexity based on POS bigrams to select a subset of the original training corpus that is the most similar to the test set. To create a reliable language model, we need a large training set similar to the test set. Since the original test set is very small, we use the complete MADAR data (50 000 sentences) for training the language model. We use the NLTK implementation of the language model module (LM), to train the Maximum Likelihood Estimator (MLE).

3.5 Evaluation

For evaluation, we use the CoNLL2018 (Zeman et al., 2018) scorer. We report different types of scores: (1) Unlabeled (UAS) and Labeled Attachment Score (LAS) for each dialect over *all dependencies* in the test sentences (All). (2) Since we focus on two different constructions, we also provide an evaluation of only those dependencies that are part of those constructions. For coordinations, the coordinating conjunction ‘wa’ depends on the first conjunct, and the second conjunct depends on ‘wa’. We evaluate these two dependencies when evaluating coordination specifically. For Idafa, we specifically evaluate all dependencies linking sequences of two or more nouns.

Treebank	UAS	LAS
CATiB	90.3	88.7
MADAR MSA	97.9	84.9

Table 2: In-domain baseline results.

4 Results

4.1 In-Domain Baselines

We first provide in-domain baseline results. These provide an upper bound of how well the parser works in an optimal situation. We provide two baselines, one where we train and test on CATiB, and one where we train and test on MADAR MSA (the syntactically annotated MSA part of MADAR). These results are shown in Table 2. Overall, the parser is very reliable. UAS is higher for MADAR MSA (97.9% as compared to 90.3% for CATiB), which is probably a result of the shorter sentences in MADAR MSA. The fact that LAS is lower for MADAR MSA (84.9% vs. 88.7% for CATiB) may point to differences in labeling decisions between the two treebanks.

4.2 Out-of-Domain Baseline

We then establish our baseline for the out-of-domain setting. For this experiment, we train the model on CATiB and test on the dialectal test set selected from MADAR. The results are shown in Table 3 (Full CATiB). As expected, the results show a degradation in performance when used out-of-domain, i.e., tested on the dialect sentences. UAS scores for all dependencies range from 55.1 (Gulf) to 57.1 (Egyptian), and LAS scores from 23.2 (Maghrebi) to 27.5 (Levantine). Since MSA does not have prepositional Idafa phrases, there is a possibility that these Idafa constructions will be parsed poorly. Since Gulf and Maghrebi have the most occurrences of prepositional Idafa, we would expect these two dialects to have the lowest LAS scores, which is the case.

For coordination dependencies, the results tend to be lower than the results on the complete sentences (only Gulf coordinations reach higher scores), and LAS and UAS are either identical or

[6] We decided against using POS tag embeddings to avoid additional discrepancies between the training and test data.

[7] For the full CAMelTools tagset, see `https://camel-tools.readthedocs.io/en/stable/reference/camel_morphology_features.html`.

		Gulf		Levantine		Egyptian		Maghrebi	
Train	Test	UAS	LAS	UAS	LAS	UAS	LAS	UAS	LAS
Full CATiB	All	55.1	25.9	57.1	27.5	57.5	26.8	55.5	23.2
	Coordination	60.0	60.0	45.2	45.2	45.4	44.4	49.0	49.0
	Idafa	84.6	43.0	88.0	54.2	91.6	49.2	91.6	38.6
Short CATiB	All	57.4	31.8	57.8	31.3	57.9	30.3	57.8	29.5
	Coordination	59.0	59.0	48.4	48.4	44.4	44.4	52.8	51.9
	Idafa	90.5	67.1	91.5	66.9	95.4	65.9	95.4	**59.0**
No sent. coord.	All	**60.9**	**34.9**	60.6	33.4	60.4	32.4	60.0	30.5
	Coordination	**73.3**	**72.3**	54.7	54.7	50.5	50.5	62.2	56.7
	Idafa	91.9	67.1	**95.0**	**71.8**	92.4	**72.7**	**96.2**	55.3
Perplexity	All	60.4	**34.9**	60.6	32.8	59.2	31.0	59.6	30.0
	Coordination	68.5	67.6	**55.7**	**55.7**	42.4	42.4	**62.5**	**61.5**
	Idafa	**92.7**	**70.8**	94.3	65.4	92.4	61.3	95.4	49.2
Add MADAR MSA	All	60.0	34.3	**62.0**	**34.2**	**60.6**	**33.5**	**61.9**	**33.3**
	Coordination	49.5	48.5	52.6	52.6	**51.5**	**51.5**	54.8	52.8
	Idafa	91.2	67.8	93.6	64.7	**96.5**	71.9	96.2	51.5

Table 3: Results for each set of experiments, for all dependencies, for coordination dependencies, and for Idafa dependencies. (Best results per dialect in bold.)

	All	Coord.	Idafa
CATiB all	37.4		
CATiB short	9.1		
MADAR	5.8	12.3	9.9
Gulf		12.1	10.0
Levantine		12.7	9.9
Egyptian		12.1	10.0
Maghrebi		12.1	10.5

Table 4: Average sentences length across datasets.

very similar. For Idafa dependencies, in contrast, UAS is considerably higher (between 84.6 for Gulf and 91.6 for Maghrebi) than LAS (between 38.6 for Maghrebi and 54.2 for Levantine). This shows that determining the structure is relatively easy, but determining the type of dependency is much more difficult.

4.3 Adjusting Sentence Length in the Training Set

One possible explanation for the low results out-of-domain can be found in the differences in average sentence length between CATiB and MADAR (see Table 4). It is well known that sentence length affects parser performance (Ababou et al., 2023), since it correlates with syntactic complexity. The sentences in CATiB are significantly longer, with an average length of 37.4. In the MADAR corpus, the average setnence length is 5.8; in our test set, the average sentence length is between 9.9 and 12.7 words across dialects. This means that the parser may learn a complex model of coordination and Idafa, which is too complex for the sentences in MADAR overall.

We address this issue by modifying the training set to closely resemble the test set. I.e., we select only sentences from CATiB that are on average of similar length to test sentences for training. When extracting sentences from CATiB, we found that the number of sentences with a similar length to the test data is too low to create a large enough training set. For this reason, we decided to extract sentences of at most 15 words, to increase the number of sentences in training data. This resulted in a total of 2 662 sentences for training. Note that this is modeling the upper range of sentence length in MADAR.

Table 3 shows the results of training on shorter sentences (Short CATiB). The overall results show an improvement of 2-3 points over training on the full CATiB in both UAS and LAS. For coordination dependencies, the shorter sentences had less of an impact, for Gulf, the results even decreased by 1 point. For Idafa, in contrast, especially LAS shows a significant improvement across UAS and LAS. The highest increase of 24.1 points occurs in LAS for Levantine.

4.4 Sentential Coordination

When looking at the parses created in the experiments above, we noticed another difference between the training and test data: MSA frequently uses connectives such as ‘wa’ (the equivalent of ’and’ in English) instead of punctuation, creating run-on sentences. We note that this construction is infrequent in the test dataset. For this reason, we examine whether removing sentences with such sentential coordination, as opposed to phrasal coordination, from the training data will help the parser learn a better model wrt. the test data.

We extract only sentences without sentential coordination, (operationalized as having a ‘wa’ in initial position) from CATiB and train on this subset; the training data for this experiment contains a total of 2 000 sentences).

The results of this experiment (no sent. coord.) in Table 3 improved overall and on the coordination dependencies for all dialects in both LAS and UAS scores in comparison to the short CATiB experiments. Surprisingly, we also see an improvement for Idafa dependencies, for all dialects and most metrics.

Gulf shows the largest results in comparison to the other dialects. We assume that this is the case because there is a syntactic similarity between Gulf syntactic structures and MSA structures.

4.5 Training on Structurally Similar Sentences

In this experiment, we examine another method for creating a training set more similar to the test data: We investigate whether increasing the structural similarity of the training to the test data is beneficial for improving the parsing results. We define structural similarity as a low perplexity score based on a pre-neural language model (using bigrams) trained on the automatically POS tagged test data. We use perplexity to select training sentences that are similar to test sentences, where lower perplexity means higher similarity. This is a standard method for domain adaptation in parsing (see e.g. Hwa, 2001; Khan et al., 2013), and does not use the syntactic annotations of the test set.

The results of the perplexity experiments are shown in Table 3 (perplexity). The overall results are slightly lower or identical to the results without sentential coordination. When we look at the coordination dependencies, we see improvements in the results for Levantine and Egyptian as well as for Idafa dependencies in Gulf.

4.6 Adding In-Domain Training Data

In the last set of experiments to adapt the training data to the test data, we include the syntactically annotated MSA subset of MADAR (a total of 2 000 sentences; see section 3.1) to the training data.

The results of these experiments are shown in Table 3 (add MADAR MSA). This setting shows the overall highest results for Levantine, Egyptian, and Maghrebi. Only for Gulf, the experiments without sentential coordination resulted in slightly higher scores. For the coordination dependencies, this setting results in the highest scores for Egyptian; for the other dialects the results are somewhat or considerably lower than the the best results. For Idafa dependencies, the results for UAS in Egyptian are the highest, the others are only slightly lower than for the experiment without sentential coordination.

4.7 Using Dialectal Embedddings

Since the experiments in the previous section resulted in only moderate improvements, we decided to investigate whether adding different embeddings would further boost our results. On the one hand, we use character embeddings, which are trained during parser training on the training data. On the other hand, we use the CAMEL BERT embeddings instead of araBERT, since the training set for those embeddings contains dialectal data. If the differences in lexicon are the major issue with parsing the MADAR data, using either type of embedding should help. However, if using those embeddings only results in minor gains, we can conclude that the difficulty is less due to dialectal differences but rather text genre difference.

Since training with the CATiB+MADAR MSA data takes a long time, we decided to use the training data without sentential coordination as basis for this set of experiments.

The results of the experiments using embeddings are shown in Table 5. For comparison purposes, we repeat the results of the experiments (no sent. coord.) that serve as the basis.

The results provide a rather inconsistent picture. The clearest trends is that coordination dependencies do not profit from dialectal embeddings, the highest results are reached across all dialects in the no sent. coord. setting. When looking at all dependencies, the same is true for Gulf and Maghrebi. For Levantine, adding character embeddings to araBERT results in a small gain (from 60.6 to 62.3

		Gulf		Levantine		Egyptian		Maghrebi	
Train	Test	UAS	LAS	UAS	LAS	UAS	LAS	UAS	LAS
No sent. coord. (araBERT)	All	**60.9**	**34.9**	60.6	33.4	**60.4**	32.4	**60.0**	**30.5**
	Coord.	**73.3**	**72.3**	**54.7**	**54.7**	**50.5**	**50.5**	**62.2**	**56.7**
	Idafa	91.9	67.1	95.0	71.8	92.4	72.7	96.2	**55.3**
araBERT + char	All	59.1	33.4	**62.3**	**34.0**	**60.4**	32.3	59.5	29.7
	Coord.	65.7	66.6	52.6	52.6	46.4	46.4	49.0	50.0
	Idafa	92.7	74.4	**95.7**	**75.3**	**96.2**	71.9	94.6	53.7
CAMEL BERT	All	59.3	33.6	60.0	32.6	59.8	31.8	58.0	29.3
	Coord.	59.0	59.0	47.3	47.3	45.4	45.4	52.8	52.8
	Idafa	**94.1**	68.6	94.3	64.7	95.4	68.9	**96.9**	50.7
CAMEL BERT + char	All	53.2	34.2	60.9	33.9	**60.4**	**33.8**	57.6	28.9
	Coord.	61.9	61.9	49.4	49.4	47.4	47.4	48.8	47.1
	Idafa	93.4	**75.1**	94.3	73.2	94.6	**75.0**	96.2	53.7

Table 5: Results using embeddings, for all dependencies, for coordination dependencies, and for Idafa dependencies. (Best results per dialect in bold.)

	Short Idafa			Complex Idafa			Prepositional Idafa		
dialect	#	% correct	(# c.)	#	% correct	(# c.)	#	% correct	(# c.)
Gulf	63	88.73	(71)	17	58.82	(10)	27	74.01	(20)
Levantine	69	75.00	(92)	18	66.67	(12)	2	0.00	(0)
Egyptian	67	69.79	(96)	19	42.11	(8)	1	100	(1)
Maghrebi	83	84.69	(98)	15	26.67	(4)	33	6.06	(2)

Table 6: Idafa performance in no sent. coord. experiments.

for UAS and from 33.4 to 34.0 for LAS). For Egyptian, all but CAMEL BERT reach the same UAS, while the highest LAS is reached by those embeddings combined with character embeddings. For Idafa, in contrast, Levantine and Egyptian profit from the addition of character embeddings, while Maghrebi does not, and Gulf profits in terms of UAS but not LAS.

Overall, the results using the dialectal embeddings, either pretrained or character embedding, are disappointing, they are very similar to the results using only araBERT. In other words, better lexical coverage does not provide major gains. A look at out-of-vocabulary (OOV) rates on the subword level shows that araBERT reaches an 86-88% overlap while the overlap for CAMEL BERT is 90-93%[8]. Since OOV rates are not a significant issue, we assume that the difficulty mostly stems from the differences in text genre.

5 Error Analysis

We had a closer look at the Idafa results based on the training set without sentential coordination. We separated the 100 Idafa sentences per dialect into those that had short Idafa constructions (involving 2 nouns but no preposition), complex Idafa (more than 2 nouns), and prepositional Idafa. The results are shown in Table 6. They show that short Idafa is recognized well by the parser across all dialects, with Egyptian having the lowest accuracy of close to 70%. For complex Idafa, the numbers are lower, as expected. However, we also see that they are easier to recognize in Gulf and Levantine while they seem extremely difficult in Maghrebi. In many of those cases, the subsequent nouns are not labeled as belonging to the Idafa, but rather as modifiers, etc. Another observation is that sometimes the first noun is also assigned the Idafa label. For instance, for the Maghrebi sentence نلغي خدمة الغسيل (Eng.: we cancel laundry service), both nouns are assigned Idafa as dependency, while only the second one is supposed to be labeled as Idafa.

Maghrebi shows the same pattern for prepositional Idafa. We assume that this is due to the more significant differences between this dialect and MSA as documented previously (Kwaik et al., 2018; Harrat et al., 2015; Abunasser, 2015). We

[8] For the exact percentages, see Table 8 in the Appendix.

also see that Levantine and Egyptian only have a small number of prepositional Idafa cases, which mirrors the distribution in the whole MADAR data.

6 Conclusion and Future Work

In this paper we investigated the question of dependency parsing for dialects of Arabic when using out-of-domain data for training. Since no syntactically annotated data exist for dialectal Arabic, we have annotated a small test set, focusing on two difficult to parse phenomena, Idafa and coordination. We investigated methods for making the training data more similar to the test data and the effect of using dialectal embeddings in the parser. Results show that we reach the highest results by adding MSA data from the same domain. Thus, we conclude that the text genre differences have more of a negative effect on the parser than dialectal differences. For future work, we plan to extend this study to more syntactic phenomena, and we will investigate more traditional domain adaptation methods.

Acknowledgements

This research was supported in part by Lilly Endowment, Inc., through its support for the Indiana University Pervasive Technology Institute.

References

Nabil Ababou, Azzeddine Mazroui, and Rachid Belehbib. 2023. From extended chunking to dependency parsing using traditional Arabic grammar. *Language Resources and Evaluation*, 57(3):1–33.

Ahmed Abdelali, Hamdy Mubarak, Younes Samih, Sabit Hassan, and Kareem Darwish. 2021. QADI: Arabic dialect identification in the wild. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 1–10, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Muhammad Abdul-Mageed, Mona Diab, and Sandra Kübler. 2014. SAMAR: Subjectivity and sentiment analysis for Arabic social media. *Computer Speech & Language*, 28(1):20–37.

Noor Abo Mokh, Daniel Dakota, and Sandra Kübler. 2022. Improving POS tagging for Arabic dialects on out-of-domain texts. In *Proceedings of the Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 238–248, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Ibrahim Abu Farha and Walid Magdy. 2022. The effect of Arabic dialect familiarity on data annotation. In *Proceedings of the Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 399–408, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Mahmoud Abedel Kader Abunasser. 2015. *Computational measures of linguistic variation: A study of Arabic varieties*. University of Illinois at Urbana-Champaign.

Sharefah Al-Ghamdi, Hend Al-Khalifa, and Abdulmalik Al-Salman. 2021. A dependency treebank for classical Arabic poetry. In *Proceedings of the Sixth International Conference on Dependency Linguistics (Depling, SyntaxFest 2021)*, pages 1–9, Sofia, Bulgaria. Association for Computational Linguistics.

Karim Al-Haff, Mustafa Jarrar, Tymaa Hammouda, and Fadi Zaraket. 2022. Curras + baladi: Towards a Levantine corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference (LREC)*, pages 769–778, Marseille, France.

Rania Al-Sabbagh and Roxana Girju. 2012. YADAC: Yet another dialectal Arabic corpus. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2882–2889, Istanbul, Turkey. European Language Resources Association (ELRA).

Mohammed Attia and Ali Elkahky. 2019. Segmentation for domain adaptation in Arabic. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 119–129, Florence, Italy. Association for Computational Linguistics.

Houda Bouamor, Nizar Habash, Mohammad Salameh, Wajdi Zaghouani, Owen Rambow, Dana Abdulrahim, Ossama Obeid, Salam Khalifa, Fadhl Eryani, Alexander Erdmann, and Kemal Oflazer. 2018. The MADAR Arabic dialect corpus and lexicon. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC)*, Miyazaki, Japan.

Houda Bouamor, Sabit Hassan, and Nizar Habash. 2019. The madar shared task on arabic fine-grained dialect identification. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 199–207.

David Chiang, Mona Diab, Nizar Habash, Owen Rambow, and Safiullah Shareef. 2006. Parsing Arabic dialects. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 369–376.

Matthieu Constant, Joseph Le Roux, and Nadi Tomeh. 2016. Deep lexical segmentation and syntactic parsing in the easy-first dependency framework. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1095–1101, San Diego, California.

Daniel Dakota and Sandra Kübler. 2024. Bits and pieces: Investigating the effects of subwords in multi-task parsing across languages and domains. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2397–2409, Torino, Italia. ELRA and ICCL.

Daniel Dakota, Zeeshan Ali Sayyed, and Sandra Kübler. 2021. Bidirectional domain adaptation using weighted multi-task learning. In *Proceedings of the 17th International Conference on Parsing Technologies and the IWPT 2021 Shared Task on Parsing into Enhanced Universal Dependencies (IWPT 2021)*, pages 93–105, Online. Association for Computational Linguistics.

Kareem Darwish, Nizar Habash, Mourad Abbas, Hend Al-Khalifa, Huseein T. Al-Natsheh, Houda Bouamor, Karim Bouzoubaa, Violetta Cavalli-Sforza, Samhaa R. El-Beltagy, Wassim El-Hajj, Mustafa Jarrar, and Hamdy Mubarak. 2021. A panoramic survey of natural language processing in the Arab world. *Communications of the ACM*, 64(4):72–81.

Kareem Darwish, Hamdy Mubarak, Ahmed Abdelali, and Mohamed Eldesouki. 2017. Arabic POS tagging: Don't abandon feature engineering just yet. In *Proceedings of the Third Arabic Natural Language Processing Workshop*, pages 130–137, Valencia, Spain.

Kareem Darwish, Hamdy Mubarak, Ahmed Abdelali, 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))*, Miyazaki, Japan.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47(2):255–308.

Timothy Dozat and Christopher Manning. 2017. Deep biaffine attention for neural dependency parsing. In *5h International Conference on Learning Representations (ICLR 2017)*, Toulon, France.

Mahmoud El-Haj. 2020. Habibi - a multi dialect multi national Arabic song lyrics corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference (LREC)*, pages 1318–1326, Marseille, France.

Ahmed Elshabrawy, Muhammed AbuOdeh, Go Inoue, and Nizar Habash. 2023. CamelParser2.0: A state-of-the-art dependency parser for Arabic. In *Proceedings of ArabicNLP 2023*, pages 170–180, Singapore (Hybrid).

Agnieszka Falenska and Özlem Çetinoğlu. 2017. Lexicalized vs. delexicalized parsing in low-resource scenarios. In *Proceedings of the 15th International Conference on Parsing Technologies*, pages 18–24, Pisa, Italy.

Daniel Gildea. 2001. Corpus variation and parser performance. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 167–202, Pittsburgh, PA.

Spence Green and Christopher D Manning. 2010. Better Arabic parsing: Baselines, evaluations, and analysis. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING)*, pages 394–402.

Elisa Gugliotta, Michele Mallia, and Livia Panascì. 2023. Towards a unified digital resource for Tunisian Arabic. In *Proceedings of the 4th Conference on Language, Data and Knowledge*, pages 579–590, Vienna, Austria. NOVA CLUNL, Portugal.

Nizar Habash. 2010. *Introduction to Arabic Natural Language Processing*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Nizar Habash, Muhammed AbuOdeh, Dima Taji, Reem Faraj, Jamila El Gizuli, and Omar Kallas. 2022. Camel Treebank: An open multi-genre Arabic dependency treebank. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference (LREC)*, pages 2672–2681, Marseille, France.

Nizar Habash and Ryan Roth. 2009. CATiB: The Columbia Arabic Treebank. In *Proceedings of the ACL-IJCNLP Conference*, pages 221–224, Suntec, Singapore.

Salima Harrat, Karima Meftouh, Mourad Abbas, Salma Jamoussi, Motaz Saad, and Kamel Smaili. 2015. Cross-dialectal Arabic processing. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 620–632, Cairo, Egypt. Springer.

Rebecca Hwa. 2001. On minimizing training corpus for parser acquisition. In *Proceedings of the Workshop on Computational Natural Language Learning (CoNLL)*, Toulouse, France.

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, Kyiv, Ukraine (Virtual).

Yash Kankanampati, Joseph Le Roux, Nadi Tomeh, Dima Taji, and Nizar Habash. 2020. Multitask easy-first dependency parsing: Exploiting complementarities of different dependency representations. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2497–2508, Barcelona, Spain (Online).

Mohammad Khan, Markus Dickinson, and Sandra Kübler. 2013. Towards domain adaptation for parsing web data. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP)*, Hissar, Bulgaria.

Kathrein Abu Kwaik, Motaz Saad, Stergios Chatzikyriakidis, and Simon Dobnika. 2018. A lexical distance study of Arabic dialects. *Procedia Computer Science*, 142:2–13.

Ying Li, Zhenghua Li, and Min Zhang. 2020. Semi-supervised domain adaptation for dependency parsing via improved contextualized word representations. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3806–3817, Barcelona, Spain (Online).

Zhenghua Li, Xue Peng, Min Zhang, Rui Wang, and Luo Si. 2019. Semi-supervised domain adaptation for dependency parsing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2386–2395, Florence, Italy.

Zhuang Liu, Degen Huang, Kaiyu Huang, Zhuang Li, and Jun Zhao. 2020. Finbert: A pre-trained financial language representation model for financial text mining. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 4513–4519. International Joint Conferences on Artificial Intelligence Organization. Special Track on AI in FinTech.

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*, volume 27, pages 466–467. Cairo, Egypt.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: a tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, Online. Association for Computational Linguistics.

Yuval Marton, Nizar Habash, and Owen Rambow. 2010. Improving Arabic dependency parsing with lexical and inflectional morphological features. In *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 13–21.

Yuval Marton, Nizar Habash, and Owen Rambow. 2013. Dependency parsing of Modern Standard Arabic with lexical and inflectional features. *Computational Linguistics*, 39(1):161–194.

Ryan McDonald, Slav Petrov, and Keith Hall. 2011. Multi-source transfer of delexicalized dependency parsers. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 62–72, Edinburgh, Scotland, UK.

Karima Meftouh, Salima Harrat, Salma Jamoussi, Mourad Abbas, and Kamel Smaili. 2015. Machine translation experiments on PADIC: A parallel Arabic dialect corpus. In *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation*, pages 26–34.

Emad Mohamed. 2011. The effect of automatic tokenization, vocalization, stemming, and POS tagging on Arabic dependency parsing. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, pages 10–18, Portland, Oregon, USA. Association for Computational Linguistics.

Atreyee Mukherjee and Sandra Kübler. 2018. Domain adaptation in dependency parsing via transformation based error driven learning. In *Proceedings of the 17th International Workshop on Treebanks and Linguistic Theories (TLT 2018), December 13–14, 2018, Oslo University, Norway*, pages 179–192.

Joakim Nivre, Johan Hall, Sandra Kübler, Ryan McDonald, Jens Nilsson, Sebastian Riedel, and Deniz Yuret. 2007a. The CoNLL 2007 shared task on dependency parsing. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 915–932, Prague, Czech Republic. Association for Computational Linguistics.

Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gülşen Eryiğit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. 2007b. Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.

Ossama Obeid, Nasser Zalmout, Salam Khalifa, Dima Taji, Mai Oudah, Bashar Alhafni, Go Inoue, Fadhl Eryani, Alexander Erdmann, and Nizar Habash. 2020. CAMeL tools: An open source python toolkit for Arabic natural language processing. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 7022–7032, Marseille, France.

Barbara Plank and Gertjan van Noord. 2011. Effective measures of domain similarity for parsing. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1566–1576, Portland, Oregon, USA.

Rudolf Rosa and Zdeněk Žabokrtský. 2015. KLcpos3 - a language similarity measure for delexicalized parser transfer. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 243–249, Beijing, China.

Ali Safaya, Moutasem Abdullatif, and Deniz Yuret. 2020. KUISAIL at SemEval-2020 task 12: BERT-CNN for offensive speech identification in social media. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 2054–2059, Barcelona (online).

Djamé Seddah, Farah Essaidi, Amal Fethi, Matthieu Futeral, Benjamin Muller, Pedro Javier Ortiz Suárez, Benoît Sagot, and Abhishek Srivastava. 2020. Building a user-generated content North-African Arabizi treebank: Tackling hell. In *Proceedings of the 58th Annual Meeting of the Association for Computational*

Linguistics, pages 1139–1150, Online. Association for Computational Linguistics.

Anas Shahrour, Salam Khalifa, Dima Taji, and Nizar Habash. 2016. CamelParser: A system for Arabic syntactic analysis and morphological disambiguation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations*, pages 228–232, Osaka, Japan.

Sara Stymne. 2020. Cross-lingual domain adaptation for dependency parsing. In *Proceedings of the 19th International Workshop on Treebanks and Linguistic Theories*, pages 62–69, Düsseldorf, Germany. Association for Computational Linguistics.

Dima Taji, Nizar Habash, and Daniel Zeman. 2017. Universal Dependencies for Arabic. In *Proceedings of the Third Arabic Natural Language Processing Workshop*, pages 166–176, Valencia, Spain. Association for Computational Linguistics.

Toshiyuki Takezawa, Genichiro Kikui, Masahide Mizushima, and Eiichiro Sumita. 2007. Multilingual spoken language corpus development for communication research. *International Journal of Computational Linguistics & Chinese Language Processing: Special Issue on Invited Papers from ISCSLP 2006*, 12(3):303–324.

Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. 2019. Multilingual is not enough: BERT for Finnish. *Preprint*, arXiv:1912.07076.

Daniel Zeman, Jan Hajič, Martin Popel, Martin Potthast, Milan Straka, Filip Ginter, Joakim Nivre, and Slav Petrov. 2018. CoNLL 2018 shared task: Multilingual parsing from raw text to Universal Dependencies. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–21, Brussels, Belgium.

A Appendix

Hyperparameters	Value
Word Embedding Dimension	100
Character Embedding Dimension	100
Number of BERT Layers Used	4
Bert Mapping Dimension	100
Optimizer	Adam
Patience	100
Batch Size	20000 tokens
Learning Rate	2e-3

Table 7: Parser hyperparameter settings

		araBERT	camel-bert
Gulf	Coord	87.46	91.33
Gulf	Idafa	87.80	91.18
Levantine	Coord	87.78	91.64
Levantine	Idafa	88.11	93.12
Egyptian	Coord	87.75	91.57
Egyptian	Idafa	87.49	91.93
Maghrebi	Coord	87.52	90.29
Maghrebi	Idafa	86.35	90.72

Table 8: Percentage of subword overlap rates when generating averages embedding representations over subtokens, using the no sent. coord. training condition.