Identifying Broken Plurals, Irregular Gender, and Rationality in Arabic Text

Sarah Alkuhlani and **Nizar Habash**
Center for Computational Learning Systems
Columbia University
{sma2149,nh2142}@columbia.edu

Abstract

Arabic morphology is complex, partly because of its richness, and partly because of common irregular word forms, such as broken plurals (which resemble singular nouns), and nouns with irregular gender (feminine nouns that look masculine and vice versa). In addition, Arabic morpho-syntactic agreement interacts with the lexical semantic feature of rationality, which has no morphological realization. In this paper, we present a series of experiments on the automatic prediction of the latent linguistic features of functional gender and number, and rationality in Arabic. We compare two techniques, using simple maximum likelihood (MLE) with back-off and a support vector machine based sequence tagger (Yamcha). We study a number of orthographic, morphological and syntactic learning features. Our results show that the MLE technique is preferred for words seen in the training data, while the Yamcha technique is optimal for unseen words, which are our real target. Furthermore, we show that for unseen words, morphological features help beyond orthographic features and that syntactic features help even more. A combination of the two techniques improves overall performance even further.

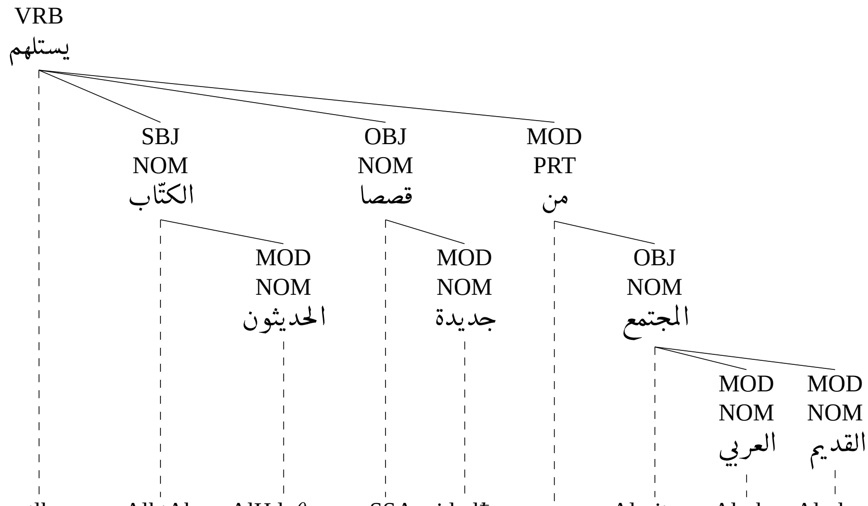
1 Introduction

Arabic morphology is complex, partly because of its richness, and partly because of its complex morpho-syntactic agreement rules which depend on functional features not necessarily expressed in word forms. Particularly challenging are broken plurals (which resemble singular nouns), nouns with irregular gender (masculine nouns that look feminine and feminine nouns that look masculine), and the semantic feature of rationality, which has no morphological realization (Smrž, 2007b; Alkuhlani and Habash, 2011). These features heavily participate in Arabic morpho-syntactic agreement. Alkuhlani and Habash (2011) show that without proper modeling, Arabic agreement cannot be accounted for in about a third of all noun-adjective pairs and a quarter of verb-subject pairs. They also report that over half of all plurals in Arabic are irregular, 8% of nominals have irregular gender and almost half of all proper nouns and 5% of all nouns are rational.

In this paper, we present results on the task of automatic identification of functional gender, number and rationality of Arabic words in context. We consider two supervised learning techniques: a simple maximum-likelihood model with back-off (MLE) and a support-vector-machine-based sequence tagger, Yamcha (Kudo and Matsumoto, 2003). We consider a large number of orthographic, morphological and syntactic learning features. Our results show that the MLE technique is preferred for words seen in the training data, while the Yamcha technique is optimal for unseen words, which are our real target. Furthermore, we show that for unseen words, morphological features help beyond orthographic features and that syntactic features help even more. A combination of the two techniques improves overall performance even further.

This paper is structured as follows: Sections 2 and 3 present relevant linguistic facts and related work, respectively. Section 4 presents the data collection we use and the metrics we target. Section 5 discusses our approach. And Section 6 presents our results.

Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, pages 675–685,
Avignon, France, April 23 - 27 2012.



Word	ystlhm	AlktAb	AlHdyθwn	qSSA	jdyd$\hbar$	mn	Almjtmς	Alςrby	Alqdym
Form	MS	MS	MP	MS	FS	NaNa	MS	MS	MS
Func	MSN	MPR	MPN	FPI	FSN	NaNaNa	MSI	MSN	MSN
Gloss	be-inspired	the-writers	the-modern	stories	new	from	culture	Arab	ancient
English	'Modern writers are inspired by ancient Arab culture to write new stories .'								

Figure 1: An example Arabic sentence showing its dependency representation together with the form-based and functional gender and number features and rationality. The dependency tree is in the CATiB treebank representation (Habash and Roth, 2009). The shown POS tags are VRB "verb", NOM "nominal (noun/adjective)", and PRT "particle". The relations are SBJ "subject", OBJ "object" and MOD "modifier". The form-based features are only for gender and number.

2 Linguistic Facts

Arabic has a rich and complex morphology. In addition to being both templatic (root/pattern) and concatenative (stems/affixes/clitics), Arabic's optional diacritics add to the degree of word ambiguity. We focus on two problems of Arabic morphology: the discrepancy between morphological form and function; and the complexity of morpho-syntactic agreement rules.

2.1 Form and Function

Arabic nominals (*i.e. nouns, proper nouns and adjectives*) and verbs inflect for gender: masculine (M) and feminine (F), and for number: singular (S), dual (D) and plural (P). These features are regularly expressed using a set of suffixes that uniquely convey gender and number combinations: +ϕ (MS), ة+ +$\hbar$[1] (FS), ون+ +*wn* (MP), and ات+ +*At* (FP). For example, the adjective ماهر *mAhr* 'clever' has the following forms among others: ماهر *mAhr* (MS), ماهرة *mAhr$\hbar$* (FS), ماهرون *mAhrwn* (MP), and ماهرات *mAhrAt* (FP). For a sizable minority of words, these features are expressed templatically, i.e., through pattern change, coupled with some singular suffix. A typical example of this phenomenon is the class of *broken plurals*, which accounts for over half of all plurals (Alkuhlani and Habash, 2011). In such cases, the form of the morphology (singular suffix) is inconsistent with the word's functional number (plural). For example, the word كاتب *kAtb* (MS) 'writer' has the broken plural: كتّاب *ktAb* ($\frac{MS}{MP}$).[2] See the second word in the example in Figure 1, which is the word كتّاب *ktAb* 'writers' prefixed with the definite article *Al+*. In addition to broken plurals, Arabic has words with irregular gender, e.g., the feminine singular adjective 'red' حمراء *HmrA'* ($\frac{MS}{FS}$), and the nouns خليفة *xlyf$\hbar$* ($\frac{FS}{MS}$) 'caliph' and حامل *HAml* ($\frac{MS}{FS}$) 'pregnant'. Verbs and nominal duals do not display this discrepancy.

2.2 Morpho-syntactic Agreement

Arabic gender and number features participate in morpho-syntactic agreement within specific con-

[1]Arabic transliteration is presented in the Habash-Soudi-Buckwalter (HSB) scheme (Habash et al., 2007): (in alphabetical order) *AbtθjHxd$\eth$rzs$\check{s}$SDT$\check{D}\varsigma\gamma$fqklmnhwy* and the additional symbols: ' ء, Â أ, Ǎ إ, Ā آ, ŵ ؤ, ŷ ئ, $\hbar$ ة, ý ى.

[2]This nomenclature denotes ($\frac{Form}{Function}$).

structions such as nouns with their adjectives and verbs with their subjects. Arabic agreement rules are more complex than the simple matching rules found in languages such as Spanish (Holes, 2004; Habash, 2010). For instance, Arabic adjectives agree with the nouns they modify in gender and number except for plural irrational (non-human) nouns, which always take feminine singular adjectives. Rationality ('humanness' 'عاقل/غير عاقل') is a morpho-lexical feature that is narrower than animacy. English expresses it mainly in pronouns (*he/she* vs. *it*) and relativizers (*men who...* vs. *cars/cows which...*). We follow the convention by Alkuhlani and Habash (2011) who specify rationality as part of the functional features of the word. The values of this feature are: rational (R), irrational (I), and not-specified (N). N is assigned to verbs, adjectives, numbers and quantifiers.[3] For example, in Figure 1, the plural rational noun الكتّاب *AlktAb* ($\frac{MS}{MPR}$) 'writers' takes the plural adjective الحديثون *AlHdyθwn* ($\frac{MP}{MPN}$) 'modern'; while the plural irrational word قصصا *qSSA* 'stories' ($\frac{MS}{FPI}$) takes the feminine singular adjective جديدة *jdydħ* ($\frac{FS}{FSN}$).

3 Related Work

Much work has been done on Arabic morphological analysis, morphological disambiguation and part-of-speech (POS) tagging (Al-Sughaiyer and Al-Kharashi, 2004; Soudi et al., 2007; Habash, 2010). The bulk of this work does not address form-function discrepancy or morpho-syntactic agreement issues. This includes the most commonly used resources and tools for Arabic NLP: the Buckwalter Arabic Morphological Analyzer (BAMA) (Buckwalter, 2004) which is used in the Penn Arabic Tree Bank (PATB) (Maamouri et al., 2004), and the various POS tagging and morphological disambiguation tools trained using them (Diab et al., 2004; Habash and Rambow, 2005). There are some important exceptions (Goweder et al., 2004; Habash, 2004; Smrž, 2007b; Elghamry et al., 2008; Abbès et al., 2004; Attia, 2008; Altantawy et al., 2010; Alkuhlani and Habash, 2011).

In terms of resources, Smrž (2007b)'s work contrasting illusory (form) features and functional features inspired our distinction of morphological form and function. However, unlike him, we do not distinguish between sub-functional (logical and formal) features. His ElixirFM analyzer (Smrž, 2007a) extends BAMA by including functional number and *some* functional gender information, but not rationality. This analyzer was used as part of the annotation of the Prague Arabic Dependency Treebank (PADT) (Smrž and Hajič, 2006). More recently, Alkuhlani and Habash (2011) built on the work of Smrž (2007b) and extended beyond it to fully annotate functional gender, number and rationality in the PATB part 3. We use their resource to train and evaluate our system.

In terms of techniques, Goweder et al. (2004) investigated several approaches using root and pattern morphology for identifying broken plurals in undiacritized Arabic text. Their effort resulted in an improved stemming system for Arabic information retrieval that collapses singulars and plurals. They report results on identifying broken plurals out of context. Similar to them, we undertake the task of identifying broken plurals; however, we also target the templatic gender and rationality features, and we do this in-context. Elghamry et al. (2008) presented an automatic cue-based algorithm that uses bilingual and monolingual cues to build a web-extracted lexicon enriched with gender, number and rationality features. Their automatic technique achieves an F-score of 89.7% against a gold standard set. Unlike them, we use a manually annotated corpus to train and test the prediction of gender, number and rationality features.

Our approach to identifying these features explores a large set of orthographic, morphological and syntactic learning features. This is very much following several previous efforts in Arabic NLP in which different tagsets and morphological features have been studied for a variety of purposes, e.g., base phrase chunking (Diab, 2007) and dependency parsing (Marton et al., 2010). In this paper we use the parser of Marton et al. (2010) as our source of syntactic learning features. We follow their splits for training, development and testing.

[3]We previously defined the rationality value N as *not-applicable* when we only considered nominals (Alkuhlani and Habash, 2011). In this work, we rename the rationality value N as *not-specified* without changing its meaning. We use the value Na (*not-applicable*) for parts-of-speech that do not have a meaningful value for any feature, e.g., prepositions have gender, number and rationality values of Na.

4 Problem Definition

Our goal is to predict the functional gender, number and rationality features for all words.

4.1 Corpus and Experimental Settings

We use the corpus of Alkuhlani and Habash (2011), which is based on the PATB. The corpus contains around 16.6K sentences and over 400K tokens. We use the train/development/test splits of Marton et al. (2010). We train on a quarter of the training set and classify words in sequence. We only use a portion of the training data to increase the percentage of words unseen in training. We also compare to using all of the training data in Section 6.7.

Our data is gold tokenized; however, all of the features we use are predicted using MADA (Habash and Rambow, 2005) following the work of Marton et al. (2010). Words whose tags are unknown in the training set are excluded from the evaluation, but not training. In terms of ambiguity, the percentage of word types with ambiguous gender, number and rationality in the train set is 1.35%, 0.79%, and 4.8% respectively. These percentages are consistent with how we perform on these features, with number being the easiest and rationality the hardest.

4.2 Metrics

We report all results in terms of token accuracy. Evaluation is done for the following sets: all words, seen words, and unseen words. A word is considered seen if it is in the training data regardless of whether it appears with the same lemma and POS tag or not. Defining seen words this way makes the decision on whether a word is seen or unseen unaffected by lemma and/or POS prediction errors in the development and test sets. Using our definition of seen words, 34.3% of words types (and 10.2% of word tokens) in the development set have not been seen in quarter of the training set.

We train single classifiers for G (gender), N (number), R (rationality), GN and GNR, and evaluate them. We also combine the tags of the single classifiers into larger tags (G+N, GN+R and G+N+R).

5 Approach

Our approach involves using two techniques: MLE with back-off and Yamcha. For each technique, we explore the effects of different learning features and try to come up with the best technique and feature set for each target feature.

5.1 Learning Features

We investigate the contribution of different learning features in predicting functional gender, number and rationality features. The learning features are explored in the following order:

Orthographic Features These features are organized in two sets: W1 is the unnormalized form of the word, and W2 includes W1 plus letter n-grams. The n-grams used are the first letter, first two letters, last letter, and last two letters of the word form. We tried using the Alif/Ya normalized forms of the words (Habash, 2010), but these behaved consistently worse than the unnormalized forms.

Morphological Features We explore the following morphological features inspired by the work of Marton et al. (2010):

- POS tags. We experiment with different POS tag sets: CATiB-6 (6 tags) (Habash et al., 2009), CATiB-EX (44 tags), Kulick (34 tags) (Kulick et al., 2006), Buckwalter (BW) (Buckwalter, 2004), which is the tag used in the PATB (430 tags), and a reduced form of BW tag that ignores case and mood (BW-) (217 tags). These tags differ in their granularity and range from very specific tags (Buckwalter) to more general tags (CATiB).
- Lemma. We use the diacritized lemma (Lemma), and the normalized and undiacritized form of the lemma, the LMM (LMM).
- Form-based features. Form-based features (F) are extracted from the word form and do not necessarily reflect functional features. These features are form-based gender, form-based number, person and the definite article.

Syntactic Features We use the following syntactic features (SYN) derived from the CATiB dependency version of the PATB (Habash and Roth, 2009): parent, dependency relation, order of appearance (the word comes before or after its parent), the distance between the word and its parent, and the parent's orthographic and morphological features.

For all of these features, we train on gold values, but only experiment with predicted values in the development and test sets. For predicting morphological features, we use the MADA system (Habash and Rambow, 2005). The MADA system corrects for suboptimal orthographic choices and effectively produces a consistent and unnormalized orthography. For the syntactic features, we use Marton et al. (2010)'s system.

5.2 Techniques

We describe below the two techniques we explored.

MLE with Back-off We implemented an MLE system with multiple back-off modes using our set of linguistic features. The order of the back-off is from specific to general. We start with an MLE system that uses only the word form, and backs off to the most common feature value across all words (excluding unknown and Na values). This simple MLE system is used as a baseline.

As we add more features to the MLE system, it tries to match all these features to predict the value for a given word. If such a combination of features is not seen in the training set, the system backs off to a more general combination of features. For example, if an MLE system is using the features W2+LMM+BW, the system tries to match this combination. If it is not seen in training, the system backs off to the following set: LMM+BW, and tries to return the most common value for this POS tag and lemma combination. If again it fails to find a match, it backs off to BW, and returns the most common value for that particular POS tag. If no word is seen with this POS tag, the system returns the most common value across all words.

Yamcha Sequence Tagger We use Yamcha (Kudo and Matsumoto, 2003), a support-vector-machine-based sequence tagger. We perform different experiments with the different sets of features presented above. After that, we apply a consistency filter that ensures that every word-lemma-pos combination always gets the same value for gender, number and rationality features. Yamcha in its default settings tags words using a window of two words before and two words after the word being tagged. This gives Yamcha an advantage over the MLE system which tags each word independently.

Single vs Joint Classification In this paper, we only discuss systems trained for a single classifier (for gender, for number and for rationality). In experiments we have done, we found that training single classifiers and combining their outcomes almost always outperforms a single joint classifier for the three target features. In other words, combining the results of G and N (G+N) outperforms the results of the single classifier GN. The same is also true for G+N+R, which outperforms GNR and GN+R. Therefore, we only present the results for the single classifiers G, N, R and their combination G+N+R.

6 Results

We perform a series of experiments increasing in feature complexity. We greedily select which features to pass on to the next level of experiments. In cases of ties, we pass the top two performers to the next step. We discuss each of these experiments next for both the MLE and Yamcha techniques. Statistical significance is measured using the McNemar test of statistical significance (McNemar, 1947).

6.1 Experiment Set I: Orthographic Features

The first set of experiments uses the orthographic features. See Table 1. The MLE system with the word only feature (W1) is effectively our baseline. It does surprisingly well for seen cases. In fact it is the highest performer across all experiments in this paper for seen cases. For unseen cases, it produces a miserable and expected low score of 21.0% accuracy. The addition of the n-gram features (W2) improves statistically significantly over W1 for unseen cases, but it is indistinguishable for seen cases. The Yamcha system shows the same difference in results between W1 and W2.

Across the two sets of features, the MLE system consistently outperforms Yamcha in the case of seen words, while Yamcha does better for unseen words. This can be explained by the fact that the MLE system matches only on the word form and if the word is unseen, it backs off to the most common value across all words. Moreover, Yamcha uses some limited context information that allows it to generalize for unseen words.

Among the target features, number is the easiest to predict, while rationality is the hardest.

	MLE								Yamcha							
	G		N		R		G+N+R		G		N		R		G+N+R	
Features	seen	unseen	seen	unseen	seen	unseen	seen	unseen	seen	unseen	seen	unseen	seen	unseen	seen	unseen
W1	**99.2**	61.6	**99.3**	69.2	**97.4**	44.7	**97.0**	21.0	95.9	67.8	96.7	72.0	94.5	67.4	90.2	35.2
W2	**99.2**	**81.7**	**99.3**	**81.6**	**97.4**	**63.4**	**97.0**	**49.1**	**97.1**	**86.6**	**97.7**	**87.1**	**95.6**	**82.0**	**92.8**	**65.5**

Table 1: Experiment Set I: Baselines and simple orthographic features. W1 is the word only. W2 is the word with additional 1-gram and 2-gram prefix and suffix features. All numbers are accuracy percentages.

	MLE								Yamcha							
	G		N		R		G+N+R		G		N		R		G+N+R	
Features	seen	unseen	seen	unseen	seen	unseen	seen	unseen	seen	unseen	seen	unseen	seen	unseen	seen	unseen
W2+F	99.2	86.9	99.3	88.9	97.4	63.4	96.9	51.9	97.7	89.8	98.1	91.7	96.0	83.5	93.8	72.0
W2+Lemma	97.4	68.3	97.6	71.5	95.6	**70.3**	95.2	**33.8**	97.4	**86.8**	97.7	86.4	96.1	82.2	93.3	65.4
W2+LMM	**99.1**	**68.8**	**99.3**	**71.7**	**97.2**	67.6	**96.8**	33.2	**97.5**	86.7	**97.9**	**86.6**	96.1	**82.6**	**93.5**	**65.7**
W2+CATIB	**99.1**	85.0	**99.3**	83.8	**97.4**	70.0	**97.1**	56.2	**97.5**	87.9	**98.0**	88.6	96.0	83.5	93.6	69.7
W2+CATIB-EX	**99.1**	85.7	**99.3**	84.3	**97.4**	70.4	**97.1**	56.7	**97.5**	88.0	97.9	88.1	96.0	83.6	93.6	69.9
W2+Kulick	99.0	86.7	99.1	85.6	97.1	78.7	96.7	65.5	97.3	88.8	97.9	89.4	95.8	83.5	93.3	70.9
W2+BW-	99.0	**88.8**	99.0	**88.8**	97.0	**80.7**	96.6	**68.5**	**97.5**	**89.7**	**98.0**	**91.2**	96.0	85.2	**93.7**	**73.2**
W2+BW	98.6	87.9	98.5	**88.8**	96.8	80.3	95.9	67.8	**97.5**	89.5	97.9	89.5	**96.1**	**85.7**	**93.7**	72.8

Table 2: Experiment Set II.a: Morphological features: (i) form-based gender and number, (ii) lemma and LMM (undiacritized lemma) and (iii) a variety of POS tag sets. For each subset, the best performers are bolded.

6.2 Experiment Set II: Morphological Features

Individual Morphological Features In this set of experiments, we use our best system from the previous set, W2, and add individual morphological features to it. We organize these features in three sub-groups: (i) form-based features (F), (ii) lemma and LMM, and (iii) the five POS tag sets. See Table 2.

The F, Lemma and LMM improve over the baseline in terms of unseen words for both MLE and Yamcha techniques. However, for seen words, these systems do worse than or equal to the baseline when the MLE technique is used. The MLE system in these cases tries to match the word and its morphological features as a single unit and if such a combination is not seen, it backs off to the morphological feature which is more general. Since we are using predicted data, prediction errors could be the reason behind this decrease in accuracy for seen words. Among these systems, W2+F is the best for both Yamcha and MLE except for rationality which is expected since there are no form-based features for rationality. In this set of experiments, Yamcha consistently outperforms MLE when it comes to unseen words, but for seen words, MLE does better almost always. LMM overall does better than Lemma. This is reasonable given that LMM is easier to predict; although LMM is more ambiguous.

As for the POS tag sets, looking at the MLE results, CATIB-EX is the best performer for seen words, and BW- is the best for unseen. CATIB-6 is a general POS tag set and since the MLE technique is very strict in its matching process (an exact match or no match), using a general key to match on adds a lot of ambiguity. With Yamcha, BW and BW- are the best among all POS. Yamcha is still doing consistently better in terms of unseen words. The best two systems from both Yamcha and MLE are used as the basic systems for the next subset of experiments where we combine the morphological features.

Combined Morphological Features Until this point, all experiments using the two techniques are similar. In this subset, MLE explores the effect of using the CATIB-EX and BW- with other morphological features. And Yamcha explores the effect of using BW- and BW with other morphological features. See Table 3. Again, Yamcha is still doing consistently better in terms of unseen words, but when it comes to seen words, MLE performs better. For seen words, our best results come from MLE using CATIB-EX and LMM. For unseen words, our best results come from Yamcha with the BW- tag and the form-based features

MLE								
Features:	G		N		R		G+N+R	
W2	seen	unseen	seen	unseen	seen	unseen	seen	unseen
+CATIB-EX	**99.1**	85.7	**99.3**	84.3	**97.4**	70.4	**97.0**	56.7
+F	98.7	88.6	99.1	89.4	94.9	70.4	94.3	59.7
+LMM	**99.1**	78.9	**99.3**	80.4	97.3	69.6	96.9	44.7
+LMM+F	98.7	89.9	99.0	89.7	94.8	69.6	94.2	58.1
+BW-	99.0	88.8	99.0	88.8	97.0	80.7	96.6	68.5
+F	99.0	88.8	99.1	**89.9**	97.0	80.7	96.6	69.6
+LMM	98.9	**90.0**	99.0	88.0	97.0	**83.6**	96.6	69.8
+LMM+F	98.9	**90.0**	99.0	89.1	97.0	**83.6**	96.6	**70.8**

Yamcha								
Features:	G		N		R		G+N+R	
W2	seen	unseen	seen	unseen	seen	unseen	seen	unseen
+BW	97.5	89.5	97.9	89.5	96.1	85.7	93.7	72.8
+F	97.8	90.6	98.2	92.4	96.3	85.3	94.2	75.4
+LMM	97.6	88.9	98.1	88.9	96.5	85.7	94.1	72.3
+LMM+F	**98.1**	90.4	**98.4**	**92.5**	**96.7**	**85.8**	**94.8**	**75.9**
+BW-	97.5	89.7	98.0	91.2	96.0	85.2	93.7	73.2
+F	97.7	**90.7**	98.2	**92.5**	96.1	85.6	94.0	75.3
+LMM	97.7	89.6	98.1	90.4	96.2	85.1	94.0	72.5
+LMM+F	98.0	90.3	98.2	92.4	96.5	85.7	94.5	75.1

Table 3: Experiment Set II.b: Combining different morphological features.

	Yamcha							
	G		N		R		G+N+R	
Features:	seen	unseen	seen	unseen	seen	unseen	seen	unseen
W2 +BW +F+SYN	97.3	90.6	97.8	92.5	96.1	86.1	93.5	76.0
W2 +BW +LMM+SYN	97.4	89.1	97.5	88.3	96.2	86.0	93.4	71.7
W2 +BW +LMM+F+SYN	**97.5**	**90.8**	**98.0**	92.5	**96.4**	**86.2**	**93.8**	**76.2**
W2 +BW- +F+SYN	97.4	90.7	97.9	**92.7**	96.1	85.2	93.5	75.0
W2 +BW- +LMM+SYN	97.4	89.5	97.7	89.8	96.1	85.7	93.4	72.1
W2 +BW- +LMM+F+SYN	97.4	**90.8**	97.9	**92.7**	96.2	85.3	93.6	75.2

Table 4: Experiment Set III: Syntactic features.

for both gender and number. For rationality, the best features to use with Yamcha are BW, LMM and form-based features. The lemma seems to actually hurt when predicting gender and number. This can be explained by the fact that gender and number features are often properties of the word form and not of the lemma. This is different for rationality, which is a property of the lemma and therefore, we expect the lemma to help.

The fact that the predicted BW set helps is not consistent with previous work by Marton et al. (2010). In that effort, BW helps parsing only in the gold condition. BW prediction accuracy is low because it includes case endings. We postulate that perhaps in our task, which is far more limited than general parsing, errors in case prediction may not matter too much. The more complex tag set may actually help establish good local agreement sequences (even if incorrect casewise), which is relevant to the target features.

6.3 Experiment Set III: Syntactic Features

This set of experiments adds syntactic features to the experiments in set II. We add syntax to the systems that uses Yamcha only since it is not obvious how to add syntactic information to the MLE system. Syntax improves the prediction accuracy for unseen words but not for seen words. In Yamcha, we can argue that the +/-2 word window allows some form of shallow syntax modeling, which is why Yamcha is doing better from the start. But the longer distance features are helping even more, perhaps because they capture agreement relations. The overall best system for unseen words is W2+BW+LMM+F+SYN, except for number, where W2+BW-+F+SYN is slightly better. In terms of G+N+R scores, W2+BW+LMM+F+SYN is statistically significantly better than all other systems in this set for seen and unseen words, except for unseen words with W2+BW+F+SYN. W2+BW+LMM+F+SYN is also statistically significantly better than its non-syntactic variant for both seen and unseen words. The prediction accuracy for seen words is still not as good as the MLE systems.

6.4 System Combination

The simple MLE W1 system, which happens to be the baseline, is the best predictor for seen words, and the more advanced Yamcha system using syntactic features is the best predictor for unseen words. Next, we create a new system that takes advantage of the two systems. We use the simple MLE W1 system for seen words, and Yamcha with syntax for unseen words. For unseen words, since each target feature has its own set of best learning features, we also build a combination system that uses the best systems for gender, number and rationality and combine their output into a single system for unseen words. For gender and rationality, we use W2+BW+LMM+F+SYN, and for number, we use W2+BW-+F+SYN. As expected the combination system outperforms the basic systems. For comparison: The MLE W1 system gets an (all, seen, unseen) scores of (89.3, 97.0, 21.0) for G+N+R, while the best single Yamcha syntactic system gets (92.0, 93.8, 76.2); the combination on the other hand gets (**94.9**, **97.0**, **76.2**). The overall (all) improvement over the MLE baseline or the best Yamcha translates into 52% error reduction or 36% error reduction, respectively.

6.5 Error Analysis

We conducted an analysis of the errors in the output of the combination system as well as the two systems that contributed to it.

In the combination system, out of the total error in G+N+R (5.1%), 53% of the cases are for seen words (3.0% of all seen) and 47% for unseen words (23.8% of all unseen). Overall, rationality errors are the biggest contributor to G+N+R error at 73% relative, followed by gender (33% relative) and number (26% relative). Among error cases of seen words, rationality errors soar to 87% relative, almost four times the corresponding gender and number errors (27% and 22%, respectively). However, among error cases of unseen words, rationality errors are 57% relative, while gender and number corresponding errors are (39% and 31%, respectively). As expected, rationality is much harder to tag than gender and number due to its higher word-form ambiguity and dependence on context.

We classified the type of errors in the MLE system for seen words, which we use in the combination system. We found that 86% of the G+N+R errors involve an ambiguity in the training data where the correct answer was present but not chosen. This is an expected limitation of the MLE approach. In the rest of the cases, the correct answer was not actually present in the training data. The proportion of ambiguity errors is almost identical for gender, number and rationality. However rationality overall is the biggest cause of error, simply due to its higher degree of ambiguity.

	All	seen	unseen
MLE W1	88.5	96.8	21.2
Yamcha BW+LMM+F	91.4	94.1	70.4
Yamcha BW+LMM+F+SYN	91.0	93.3	72.2
Combination	**94.1**	**96.8**	**72.4**

Table 5: Results on blind test. Scores for All/Seen/Unseen are shown for the G+N+R condition. We compare the MLE word baseline, with the best Yamcha system with and without syntactic features and the combined system.

Since the Yamcha system uses MADA features, we investigated the effect of the correctness of MADA features on the system prediction accuracy. The overall MADA accuracy in identifying the lemma and the Buckwalter tag *together* – a very harsh measure – is 77.0% (79.3% for seen and 56.8% for unseen). Our error analysis shows that when MADA is correct, the prediction accuracy for G+N+R is 95.6%, 96.5% and 84.4% for all, seen and unseen, respectively. However, this accuracy goes down to 79.2%, 82.5% and 65.5% for all, seen and unseen, respectively, when MADA is wrong. This suggests that the Yamcha system suffers when MADA makes wrong choices and improving MADA would lead to improvement in the system's performance.

6.6 Blind Test

Finally, we apply our baseline, best combination model and best single Yamcha syntactic model (with and without syntax) to the blind test set. The results are in Table 5. The results in the blind test are consistent with the development set. The MLE baseline is best on seen words, Yamcha is best on unseen words, syntactic features help in handling unseen words, and overall combination improves over all specific systems.

6.7 Additional Training Data

After experimenting on quarter of the train set to optimize for various settings, we train our combination system on the full train set and achieve (96.0, 96.8, 74.9) for G+N+R (all, seen, unseen) on the development set and (96.5, 96.8, 65.6) on the blind test set. As expected, the overall (all) scores are higher simply due to the additional training data. The results on seen and unseen words, which are redefined against the larger training set, are not higher than results for the quarter training data. Of course, these numbers should not be compared directly. The number of unseen word tokens in the full train set is 3.7% compared to 10.2% in quarter of the train set.

6.8 Comparison with MADA

We compare our results with the form-based features from the state-of-the-art morphological analyzer MADA (Habash and Rambow, 2005). We use the form-based gender and number features produced by MADA after we filter MADA choices by tokenization. Since MADA does not give a rationality value, we assign the value I (irrational) to nouns and proper nouns and the value N (not-specified) to verbs and adjectives. Everything else receives Na (not-applicable). The POS tags are determined by MADA.

On the development set, MADA achieves (72.6, 73.1, 58.6) for G+N+R (all, seen, unseen), where the seen/unseen distinction is based on the full training set in the previous section and is provided for comparison reasons only. The results for the test set are (71.4, 72.2, 53.7). These results are consistent with our expectation that MADA will do badly on this task since it is not designed for it (Alkuhlani and Habash, 2011). We should remind the reader that MADA-derived features are used as machine learning features in this paper, where they actually help. In the future, we plan to integrate this task inside of MADA.

6.9 Extrinsic Evaluation

We use the predicted gender, number and rationality features that we get from training on the full train set in a dependency syntactic parsing experiment. The parsing feature set we use is the best performing feature set described in (Marton et al., 2011), which used an earlier unpublished version of our MLE model. The parser we use is the Easy-First Parser (Goldberg and Elhadad, 2010). More details on this parsing experiment is in Marton et al. (2012).

The functional gender and number features increase the labeled attachment score by 0.4% absolute over a comparable model that uses the form-based gender and number features. Rationality on the other hand does not help much. One possible reason for this is the lower quality of the predicted rationality feature compared to the other features. Another possible reason is that the rationality feature is not utilized optimally in the parser.

7 Conclusions and Future Work

We presented a series of experiments for automatic prediction of the latent features of functional gender and number, and rationality in Arabic. We compared two techniques, a simple MLE with back-off and an SVM-based sequence tagger, Yamcha, using a number of orthographic, morphological and syntactic features. Our conclusions are that for words seen in training, the MLE model does best; for unseen word, Yamcha does best; and most interestingly, we found that syntactic features help the prediction for unseen words.

In the future, we plan to explore training on predicted features instead of gold features to minimize the effect of tagger errors. Furthermore, we plan to use our tools to collect vocabulary not covered by commonly used morphological analyzers and try to assign them correct functional features. Finally, we would like to use our predictions for gender, number and rationality as learning features for relevant NLP applications such as sentiment analysis, phrase-based chunking and named entity recognition.

Acknowledgments

We would like to thank Yuval Marton for help with the parsing experiments. The first author was funded by a scholarship from the Saudi Arabian Ministry of Higher Education. The rest of the work was funded under DARPA projects number HR0011-08-C-0004 and HR0011-08-C-0110.

References

Ramzi Abbès, Joseph Dichy, and Mohamed Hassoun. 2004. The Architecture of a Standard Arabic Lexical Database. Some Figures, Ratios and Categories from the DIINAR.1 Source Program. In Ali Farghaly and Karine Megerdoomian, editors, *COLING 2004 Computational Approaches to Arabic Script-based Languages*, pages 15–22, Geneva, Switzerland, August 28th. COLING.

Imad Al-Sughaiyer and Ibrahim Al-Kharashi. 2004. Arabic Morphological Analysis Techniques: A Comprehensive Survey. *Journal of the American Society for Information Science and Technology*, 55(3):189–213.

Sarah Alkuhlani and Nizar Habash. 2011. A Corpus for Modeling Morpho-Syntactic Agreement in Arabic: Gender, Number and Rationality. In *Proceedings of the 49th Annual Meeting of the Association* *for Computational Linguistics (ACL'11)*, Portland, Oregon, USA.

Mohamed Altantawy, Nizar Habash, Owen Rambow, and Ibrahim Saleh. 2010. Morphological Analysis and Generation of Arabic Nouns: A Morphemic Functional Approach. In *Proceedings of the seventh International Conference on Language Resources and Evaluation (LREC)*, Valletta, Malta.

Mohammed Attia. 2008. *Handling Arabic Morphological and Syntactic Ambiguity within the LFG Framework with a View to Machine Translation*. Ph.D. thesis, The University of Manchester, Manchester, UK.

Tim Buckwalter. 2004. Buckwalter arabic morphological analyzer version 2.0. LDC catalog number LDC2004L02, ISBN 1-58563-324-0.

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.

Mona Diab. 2007. Towards an Optimal POS tag set for Modern Standard Arabic Processing. In *Proceedings of Recent Advances in Natural Language Processing (RANLP)*, Borovets, Bulgaria.

Khaled Elghamry, Rania Al-Sabbagh, and Nagwa El-Zeiny. 2008. Cue-based bootstrapping of Arabic semantic features. In *JADT 2008: 9es Journées internationales d'Analyse statistique des Données Textuelles*.

Yoav Goldberg and Michael Elhadad. 2010. An efficient algorithm for easy-first non-directional dependency parsing. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of he Association for Computational Linguistics*, pages 742–750, Los Angeles, California, June. Association for Computational Linguistics.

Abduelbaset Goweder, Massimo Poesio, Anne De Roeck, and Jeff Reynolds. 2004. Identifying Broken Plurals in Unvowelised Arabic Text. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 246–253, Barcelona, Spain, July.

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 Association for Computational Linguistics (ACL'05)*, pages 573–580, Ann Arbor, Michigan.

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

Nizar Habash, Abdelhadi Soudi, and Tim Buckwalter. 2007. On Arabic Transliteration. In A. van den Bosch and A. Soudi, editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer.

Nizar Habash, Reem Faraj, and Ryan Roth. 2009. Syntactic Annotation in the Columbia Arabic Treebank. In *Proceedings of MEDAR International Conference on Arabic Language Resources and Tools*, Cairo, Egypt.

Nizar Habash. 2004. Large Scale Lexeme Based Arabic Morphological Generation. In *Proceedings of Traitement Automatique des Langues Naturelles (TALN-04)*, pages 271–276. Fez, Morocco.

Nizar Habash. 2010. *Introduction to Arabic Natural Language Processing*. Morgan & Claypool Publishers.

Clive Holes. 2004. *Modern Arabic: Structures, Functions, and Varieties*. Georgetown Classics in Arabic Language and Linguistics. Georgetown University Press.

Taku Kudo and Yuji Matsumoto. 2003. Fast Methods for Kernel-Based Text Analysis. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL'03)*, pages 24–31, Sapporo, Japan, July.

Seth Kulick, Ryan Gabbard, and Mitch Marcus. 2006. Parsing the Arabic Treebank: Analysis and Improvements. In *Proceedings of the Treebanks and Linguistic Theories Conference*, pages 31–42, Prague, Czech Republic.

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.

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, Los Angeles, CA, USA, June.

Yuval Marton, Nizar Habash, and Owen Rambow. 2011. Improving Arabic Dependency Parsing with Form-based and Functional Morphological Features. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL'11)*, Portland, Oregon, USA.

Yuval Marton, Nizar Habash, and Owen Rabmow. 2012. Dependency Parsing of Modern Standard Arabic with Lexical and Inflectional Features. Manuscript submitted for publication.

Quinn McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.

Otakar Smrž and Jan Hajič. 2006. The Other Arabic Treebank: Prague Dependencies and Functions. In Ali Farghaly, editor, *Arabic Computational Linguistics: Current Implementations*. CSLI Publications.

Otakar Smrž. 2007a. ElixirFM – implementation of functional arabic morphology. In *ACL 2007 Proceedings of the Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources*, pages 1–8, Prague, Czech Republic. ACL.

Otakar Smrž. 2007b. *Functional Arabic Morphology. Formal System and Implementation*. Ph.D. thesis, Charles University in Prague, Prague, Czech Republic.

Abdelhadi Soudi, Antal van den Bosch, and Günter Neumann, editors. 2007. *Arabic Computational Morphology. Knowledge-based and Empirical Methods*, volume 38 of *Text, Speech and Language Technology*. Springer, August.