Feature-Based Tagger of Approximations of Functional Arabic Morphology

Jan Hajič & Otakar Smrž

Institute of Formal and Applied Linguistics
Charles University in Prague

Tim Buckwalter & Hubert Jin

Linguistic Data Consortium
University of Pennsylvania

The Fourth Workshop on Treebanks and Linguistic Theories

December 10, 2005 Universitat de Barcelona

Context

- Morphological tagging of Arabic recently addressed by Habash and Rambow (2005), Smith et al. (2005), Diab et al. (2004)

Context

- Morphological tagging of Arabic recently addressed by Habash and Rambow (2005), Smith et al. (2005), Diab et al. (2004)
- There exist the Penn Arabic Treebank (Maamouri et al., 2004, PATB) as well as the Prague Arabic Dependency Treebank (Hajič et al., 2004, PADT), both with data sets in numerous development versions

Context

- Morphological tagging of Arabic recently addressed by Habash and Rambow (2005), Smith et al. (2005), Diab et al. (2004)
- There exist the Penn Arabic Treebank (Maamouri et al., 2004, PATB) as well as the Prague Arabic Dependency Treebank (Hajič et al., 2004, PADT), both with data sets in numerous development versions
- Functional Arabic Morphology of PADT is here approximated building on the standard Buckwalter Morphology of PATB

Context

- Morphological tagging of Arabic recently addressed by Habash and Rambow (2005), Smith et al. (2005), Diab et al. (2004)
- There exist the Penn Arabic Treebank (Maamouri et al., 2004, PATB) as well as the Prague Arabic Dependency Treebank (Hajič et al., 2004, PADT), both with data sets in numerous development versions
- Functional Arabic Morphology of PADT is here approximated building on the standard Buckwalter Morphology of PATB
- Feature-based tagger by Hajič and Hladká (1998) successfully applied to various inflectional and agglutinative languages

Outline

Morphology Disambiguation

- Arabic is a language of rich morphology, both derivational and inflectional (Holes, 2004), with highly ambiguous orthography
- Boundaries of syntactic units, **tokens**, are obscure in writing
- Orthographical **strings** consist of up to four syntactic tokens

Morphology Disambiguation

- Arabic is a language of rich morphology, both derivational and inflectional (Holes, 2004), with highly ambiguous orthography
- Boundaries of syntactic units, **tokens**, are obscure in writing
- Orthographical **strings** consist of up to four syntactic tokens

- Disambiguation encompasses subproblems like tokenization, full morphological tagging or its simplified 'part-of-speech' versions, lemmatization, diacritization or restoration of the structural components of words, plus combinations thereof

Functional Approximation

The underlying morphological engine for both the Penn Arabic Treebank and the Prague Arabic Dependency Treebank is the Buckwalter Arabic Morphological Analyzer. While PATB adopts the analyses in their original format (Maamouri et al., 2004), the PADT annotations take place on *quasi-functional* approximations organized into MorphoTrees (Smrž and Pajas, 2004).

Functional Approximation

The underlying morphological engine for both the Penn Arabic Treebank and the Prague Arabic Dependency Treebank is the Buckwalter Arabic Morphological Analyzer. While PATB adopts the analyses in their original format (Maamouri et al., 2004), the PADT annotations take place on *quasi-functional* approximations organized into MorphoTrees (Smrž and Pajas, 2004).

With respect to the linguistic view and the architecture of the tagger that we will develop, we unify the format of the morphological data by converting all the Parts of PATB into the approximation, which is done in two steps: (a) the morphs of the original input strings are re-grouped to form tokens (b) the corresponding sequences of tags are mapped into the fixed-width positional notation of PADT.

He will notify them about that through SMS messages, the Internet, and other means. سَيُخبِرُهُم بِذٰلِكَ عَن طَرِيقِ الرَّسَائِلِ القَصِيرَةِ وَالإِنتِرنِتِ وَغَيرِهَا.

He will notify them about that through SMS messages, the Internet, and other means. سيخبرهم بذلك عن طريق الرسائل القصيرة والإنترنت وغيرها.

String	Token	Token Tag	Buckwalter's M-Tags	Token Form	Token Gloss
سيخبرهم		F---------	FUT	*sa-*	will
		VIIA-3MS--	IV3MS+IV+IVSUFF_MOOD:I	*yu-ḫbir-u*	he-notify
		S----3MP4-	IVSUFF_DO:3MS	*-hum*	them
بذلك		P---------	PREP	*bi-*	about/by
		SD----MS--	DEM_PRON_MS	*ḏālika*	that
عن		P---------	PREP	*ʿan*	by/about
طريق		N-------2R	NOUN+CASE_DEF_GEN	*ṭarīq-i*	way-of
الرسائل		N-------2D	DET+NOUN+CASE_DEF_GEN	*ar-rasāʾil-i*	the-messages
القصيرة		A-----FS2D	DET+ADJ+NSUFF_FEM_SG+ +CASE_DEF_GEN	*al-qaṣīr-at-i*	the-short
والإنترنت		C---------	CONJ	*wa-*	and
		Z-------2D	DET+NOUN_PROP+ +CASE_DEF_GEN	*al-ʾinternet-i*	the-internet
وغيرها		C---------	CONJ	*wa-*	and
		FN------2R	NEG_PART+CASE_DEF_GEN	*ġayr-i*	other/not-of
		S----3FS2-	POSS_PRON_3FS	*-hā*	them

He will notify them about that through SMS messages, the Internet, and other means. سيخبرهم بذلك عن طريق الرسائل القصيرة والإنترنت وغيرها.

String	Token	Token Tag	Buckwalter's M-Tags	Token Form	Token Gloss
سيخبرهم		F---------	FUT	*sa-*	will
		VIIA-3MS--	IV3MS+IV+IVSUFF_MOOD:I	*yu-ḫbir-u*	he-notify
		S----3MP4-	IVSUFF_DO:3MS	*-hum*	them
بذلك		P---------	PREP	*bi-*	about/by
		SD----MS--	DEM_PRON_MS	*ḏālika*	that
عن		P---------	PREP	*ʿan*	by/about
طريق		N-------2R	NOUN+CASE_DEF_GEN	*ṭarīq-i*	way-of
الرسائل		N-------2D	DET+NOUN+CASE_DEF_GEN	*ar-rasāʾil-i*	the-messages
القصيرة		A-----FS2D	DET+ADJ+NSUFF_FEM_SG+ +CASE_DEF_GEN	*al-qaṣīr-at-i*	the-short
والإنترنت		C---------	CONJ	*wa-*	and
		Z-------2D	DET+NOUN_PROP+ +CASE_DEF_GEN	*al-ʾinternet-i*	the-internet
وغيرها		C---------	CONJ	*wa-*	and
		FN------2R	NEG_PART+CASE_DEF_GEN	*ġayr-i*	other/not-of
		S----3FS2-	POSS_PRON_3FS	*-hā*	them

He will notify them about that through SMS messages, the Internet, and other means. سيخبرهم بذلك عن طريق الرسائل القصيرة والإنترنت وغيرها.

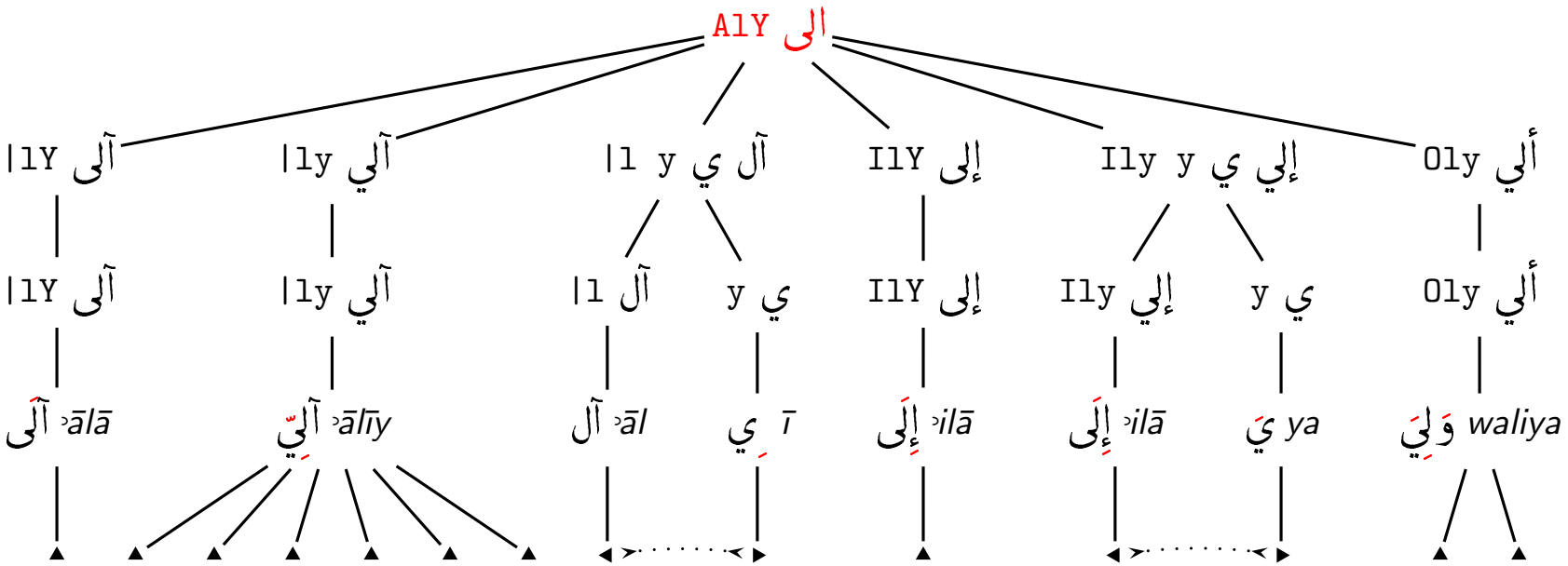
String	Token	Token Tag	Buckwalter's M-Tags	Token Form	Token Gloss
سيخبرهم		F---------	FUT	*sa-*	will
		VIIA-3MS--	IV3MS+IV+IVSUFF_MOOD:I	*yu-ḫbir-u*	he-notify
		S----3MP4-	IVSUFF_DO:3MS	*-hum*	them
بذلك		P---------	PREP	*bi-*	about/by
		SD----MS--	DEM_PRON_MS	*ḏālika*	that
عن		P---------	PREP	*ʿan*	by/about
طريق		N-------2R	NOUN+CASE_DEF_GEN	*ṭarīq-i*	way-of
الرسائل		N-------2D	DET+NOUN+CASE_DEF_GEN	*ar-rasāʾil-i*	the-messages
القصيرة		A-----FS2D	DET+ADJ+NSUFF_FEM_SG+ +CASE_DEF_GEN	*al-qaṣīr-at-i*	the-short
والإنترنت		C---------	CONJ	*wa-*	and
		Z-------2D	DET+NOUN_PROP+ +CASE_DEF_GEN	*al-ʾinternet-i*	the-internet
وغيرها		C---------	CONJ	*wa-*	and
		FN------2R	NEG_PART+CASE_DEF_GEN	*ġayr-i*	other/not-of
		S----3FS2-	POSS_PRON_3FS	*-hā*	them

MorphoTrees vs. Lists

Suppose you have a list of morphological analyses [the next slide] for a given input string . . .

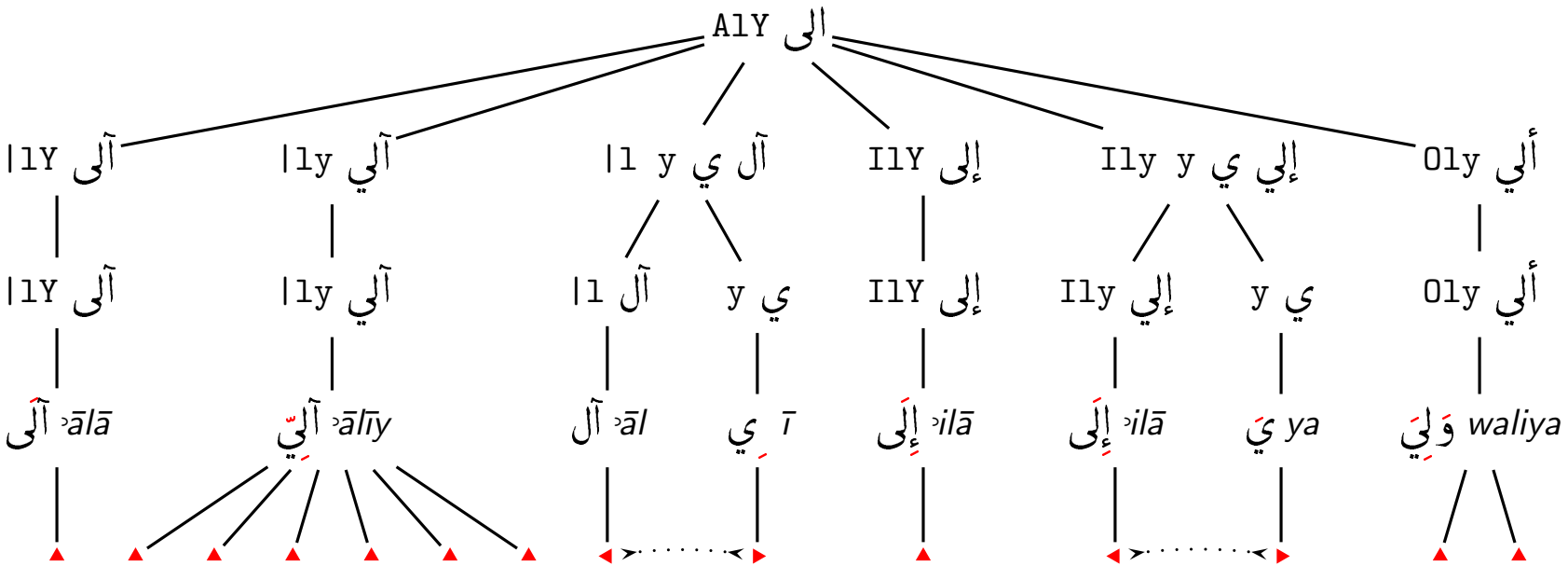
MorphoTrees vs. Lists

. . . organize them into a hierarchy with the string as its root



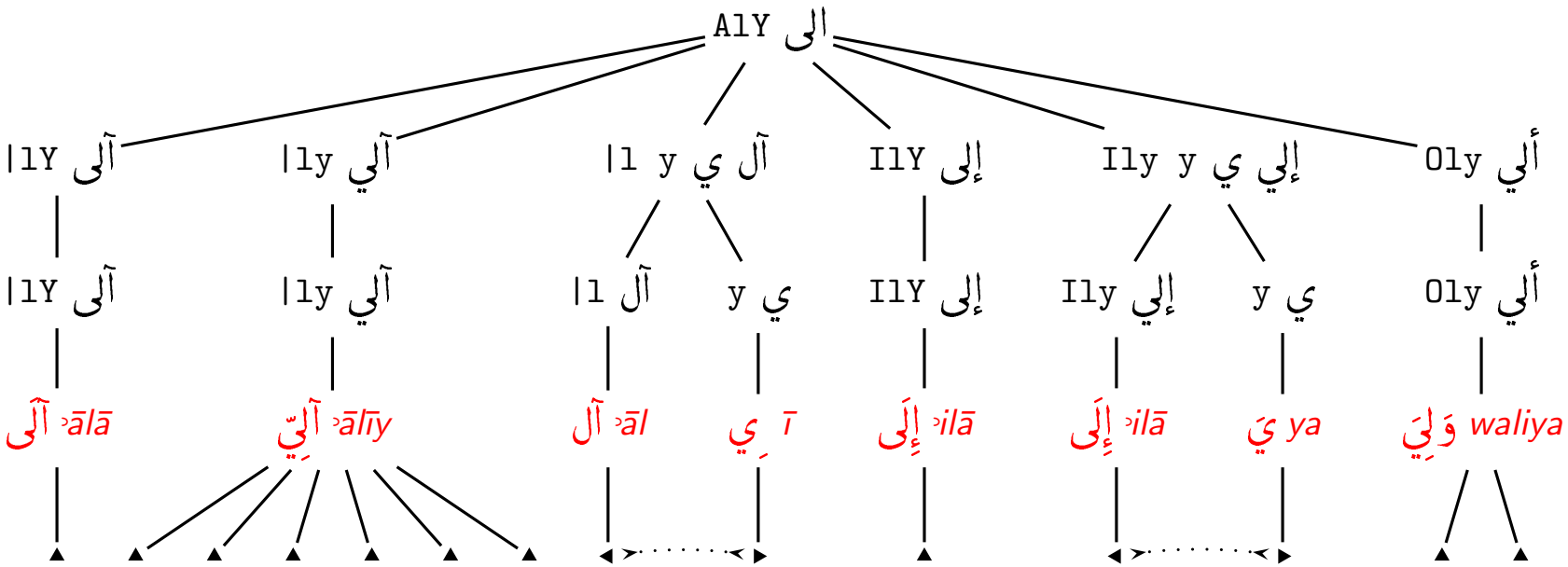
MorphoTrees vs. Lists

. . . organize them into a hierarchy with the string as its root and the full tokens as the leaves



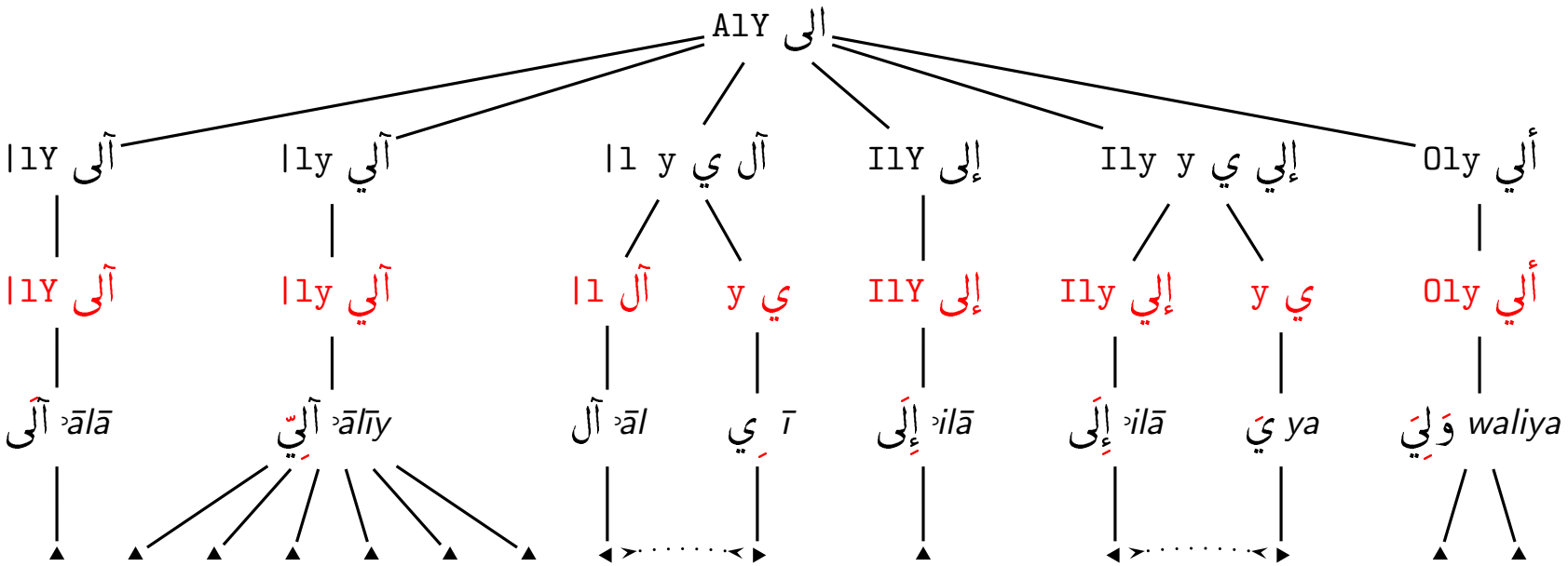
MorphoTrees vs. Lists

. . . organize them into a hierarchy with the string as its root and the full tokens as the leaves, grouped by their lemmas



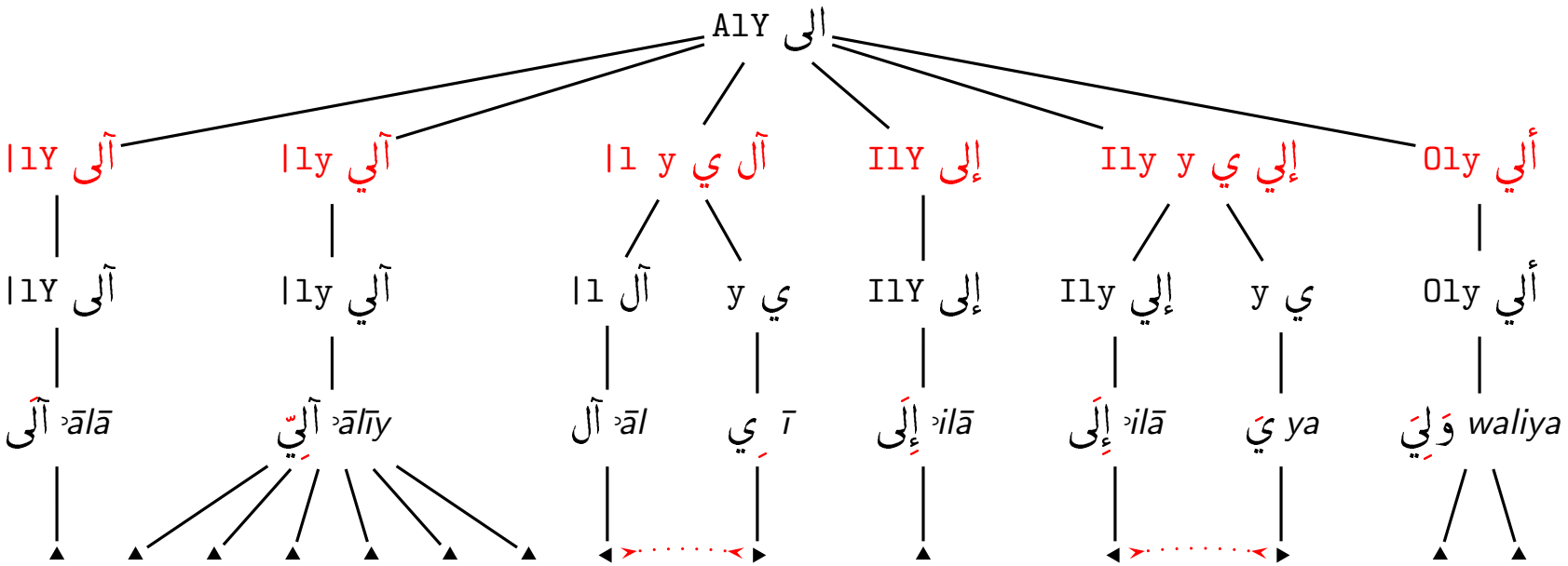
MorphoTrees vs. Lists

. . . organize them into a hierarchy with the string as its root and the full tokens as the leaves, grouped by their lemmas, canonical forms



MorphoTrees vs. Lists

. . . organize them into a hierarchy with the string as its root and the full tokens as the leaves, grouped by their lemmas, canonical forms and partitionings of the string into such forms:



Morphs	Form	Token Tag	Lemma	Glosses per Morph
|laY+(null)	ʾālā	VP-A-3MS--	ʾālā	promise/take an oath + he/it
|liy˜	ʾālīy	A---------	ʾālīy	mechanical/automatic
|liy˜+u	ʾālīy-u	A-------1R	ʾālīy	mechanical . . . + [def.nom.]
|liy˜+i	ʾālīy-i	A-------2R	ʾālīy	mechanical . . . + [def.gen.]
|liy˜+a	ʾālīy-a	A-------4R	ʾālīy	mechanical . . . + [def.acc.]
|liy˜+N	ʾālīy-un	A-------1I	ʾālīy	mechanical . . . + [indef.nom.]
|liy˜+K	ʾālīy-in	A-------2I	ʾālīy	mechanical . . . + [indef.gen.]
|l +	ʾāl	N--------R	ʾāl	family/clan +
+ iy	-ī	S----1-S2-	ī	+ my
IilaY	ʾilā	P---------	ʾilā	to/towards
Iilay +	ʾilay	P---------	ʾilā	to/towards +
+ ya	-ya	S----1-S2-	ya	+ me
Oa+liy+(null)	ʾa-lī	VIIA-1-S--	waliya	I + follow/come after + [ind.]
Oa+liy+a	ʾa-liy-a	VISA-1-S--	waliya	I + follow/come after + [sub.]

List of the morphological analyses of the input string AlY الى with the quasi-functional token tags derived from the original Buckwalter's tags.

Morphs	Form	Token Tag	Lemma	Glosses per Morph
|laY+(null)	ʾālā	VP-A-3MS--	ʾālā	promise/take an oath + he/it
|liy˜	ʾālīy	A---------	ʾālīy	mechanical/automatic
|liy˜+u	ʾālīy-u	A-------1R	ʾālīy	mechanical . . . + [def.nom.]
|liy˜+i	ʾālīy-i	A-------2R	ʾālīy	mechanical . . . + [def.gen.]
|liy˜+a	ʾālīy-a	A-------4R	ʾālīy	mechanical . . . + [def.acc.]
|liy˜+N	ʾālīy-un	A-------1I	ʾālīy	mechanical . . . + [indef.nom.]
|liy˜+K	ʾālīy-in	A-------2I	ʾālīy	mechanical . . . + [indef.gen.]
|l +	ʾāl	N--------R	ʾāl	family/clan +
+ iy	-ī	S----1-S2-	ī	+ my
IilaY	ʾilā	P---------	ʾilā	to/towards
Iilay +	ʾilay	P---------	ʾilā	to/towards +
+ ya	-ya	S----1-S2-	ya	+ me
Oa+liy+(null)	ʾa-lī	VIIA-1-S--	waliya	I + follow/come after + [ind.]
Oa+liy+a	ʾa-liy-a	VISA-1-S--	waliya	I + follow/come after + [sub.]

String Tag VISA-1-S--

Morphs	Form	Token Tag	Lemma	Glosses per Morph
|laY+(null)	ʾālā	VP-A-3MS--	ʾālā	promise/take an oath + he/it
|liy˜	ʾālīy	A---------	ʾālīy	mechanical/automatic
|liy˜+u	ʾālīy-u	A-------1R	ʾālīy	mechanical . . . + [def.nom.]
|liy˜+i	ʾālīy-i	A-------2R	ʾālīy	mechanical . . . + [def.gen.]
|liy˜+a	ʾālīy-a	A-------4R	ʾālīy	mechanical . . . + [def.acc.]
|liy˜+N	ʾālīy-un	A-------1I	ʾālīy	mechanical . . . + [indef.nom.]
|liy˜+K	ʾālīy-in	A-------2I	ʾālīy	mechanical . . . + [indef.gen.]
|l +	ʾāl	N--------R	ʾāl	family/clan +
+ iy	-ī	S----1-S2-	ī	+ my
IilaY	ʾilā	P---------	ʾilā	to/towards
Iilay +	ʾilay	P---------	ʾilā	to/towards +
+ ya	-ya	S----1-S2-	ya	+ me
Oa+liy+(null)	ʾa-lī	VIIA-1-S--	waliya	I + follow/come after + [ind.]
Oa+liy+a	ʾa-liy-a	VISA-1-S--	waliya	I + follow/come after + [sub.]

String Tag N--------RS----1-S2-

Morphs	Form	Token Tag	Lemma	Glosses per Morph
|laY+(null)	ʾālā	VP-A-3MS--	ʾālā	promise/take an oath + he/it
|liy˜	ʾālīy	A---------	ʾālīy	mechanical/automatic
|liy˜+u	ʾālīy-u	A-------1R	ʾālīy	mechanical . . . + [def.nom.]
|liy˜+i	ʾālīy-i	A-------2R	ʾālīy	mechanical . . . + [def.gen.]
|liy˜+a	ʾālīy-a	A-------4R	ʾālīy	mechanical . . . + [def.acc.]
|liy˜+N	ʾālīy-un	A-------1I	ʾālīy	mechanical . . . + [indef.nom.]
|liy˜+K	ʾālīy-in	A-------2I	ʾālīy	mechanical . . . + [indef.gen.]
|l +	ʾāl	N--------R	ʾāl	family/clan +
+ iy	-ī	S----1-S2-	ī	+ my
IilaY	ʾilā	P---------	ʾilā	to/towards
Iilay +	ʾilay	P---------	ʾilā	to/towards +
+ ya	-ya	S----1-S2-	ya	+ me
Oa+liy+(null)	ʾa-lī	VIIA-1-S--	waliya	I + follow/come after + [ind.]
Oa+liy+a	ʾa-liy-a	VISA-1-S--	waliya	I + follow/come after + [sub.]

Ambiguity Classes

```
VP-A-3MS--------------------------------
A-I- ---1RS    1 S2
NIS  1  2I
P       4
```

Morphs	Form	Token Tag	Lemma	Glosses per Morph
|laY+(null)	ʾālā	VP-A-3MS--	ʾālā	promise/take an oath + he/it
|liy˜	ʾālīy	A-----------	ʾālīy	mechanical/automatic
|liy˜+u	ʾālīy-u	A-------1R	ʾālīy	mechanical . . . + [def.nom.]
|liy˜+i	ʾālīy-i	A-------2R	ʾālīy	mechanical . . . + [def.gen.]
|liy˜+a	ʾālīy-a	A-------4R	ʾālīy	mechanical . . . + [def.acc.]
|liy˜+N	ʾālīy-un	A-------1I	ʾālīy	mechanical . . . + [indef.nom.]
|liy˜+K	ʾālīy-in	A-------2I	ʾālīy	mechanical . . . + [indef.gen.]
|l +	ʾāl	N--------R	ʾāl	family/clan +
+ iy	-ī	S----1-S2-	ī	+ my
IilaY	ʾilā	P---------	ʾilā	to/towards
Iilay +	ʾilay	P---------	ʾilā	to/towards +
+ ya	-ya	S----1-S2-	ya	+ me
Oa+liy+(null)	ʾa-lī	VIIA-1-S--	waliya	I + follow/come after + [ind.]
Oa+liy+a	ʾa-liy-a	VISA-1-S--	waliya	I + follow/come after + [sub.]

Ambiguity Classes

```
V         -
AP-   3   1-
N-IA -MS2R-     - --
PIS--1--4IS----1-S2--------------------
```

Feature-Based Tagger

- Adaptation of the feature-based, exponential-model tagger described in (Hajič and Hladká, 1998)
- Predicting individual columns/categories of tags separately, with the help of morphological dictionary

Feature-Based Tagger

- Adaptation of the feature-based, exponential-model tagger described in (Hajič and Hladká, 1998)
- Predicting individual columns/categories of tags separately, with the help of morphological dictionary

- Input strings are considered 4-tuples of tokens, inducing **string tags** as concatenations of the **token tags**

Feature-Based Tagger

- Adaptation of the feature-based, exponential-model tagger described in (Hajič and Hladká, 1998)
- Predicting individual columns/categories of tags separately, with the help of morphological dictionary

- Input strings are considered 4-tuples of tokens, inducing **string tags** as concatenations of the **token tags**

- Tagging decides tokenization as well, but lemmatization is carried out in another process

Probabilistic Model

Conditional exponential model for predicting event y (unique value of a category) in context x (set of attribute–value pairs)

$$p_{AC,e}(y|x) = \frac{\exp(\sum_{i=1}^{n} \lambda_i f_i(y,x))}{Z(x)} \quad (1)$$

where $f_i(y,x)$ is the set of binary-valued *features* of the event and its context

Probabilistic Model

Conditional exponential model for predicting event y (unique value of a category) in context x (set of attribute–value pairs)

$$p_{AC,e}(y|x) = \frac{\exp(\sum_{i=1}^{n} \lambda_i f_i(y,x))}{Z(x)} \quad (1)$$

where $f_i(y,x)$ is the set of binary-valued *features* of the event and its context, λ_i is a "weight" of f_i, and the normalization is

$$Z(x) = \sum_{y \in Y} \exp(\sum_{i=1}^{n} \lambda_i f_i(y,x)) \quad (2)$$

Probabilistic Model

Conditional exponential model for predicting event y (unique value of a category) in context x (set of attribute–value pairs)

$$p_{AC,e}(y|x) = \frac{\exp(\sum_{i=1}^{n} \lambda_i f_i(y,x))}{Z(x)} \quad (1)$$

where $f_i(y,x)$ is the set of binary-valued *features* of the event and its context, λ_i is a "weight" of f_i, and the normalization is

$$Z(x) = \sum_{y \in Y} \exp(\sum_{i=1}^{n} \lambda_i f_i(y,x)) \quad (2)$$

Modeling separately each ambiguity class AC of each of the 4×10 morphological categories, y-unigram smoothing the final distribution

$$p_{AC}(y|x) = \sigma p_{AC,e}(y|x) + (1-\sigma) p_{AC,1}(y) \quad (3)$$

Feature Contexts

The pool of contexts of prospective features is defined as a full cross-product of y and a combination of

> *[A] an ambiguity class of a single category, which may be different from the category being predicted, or [B] a word form (the input string), or [C] a single position value membership in an ambiguity class, or [D] a full tag (to the left of the current position in data only),*

Feature Contexts

The pool of contexts of prospective features is defined as a full cross-product of y and a combination of

> *[A] an ambiguity class of a single category, which may be different from the category being predicted, or [B] a word form (the input string), or [C] a single position value membership in an ambiguity class, or [D] a full tag (to the left of the current position in data only),*

with

> *[E] the current position, or [F] immediately preceding/following position in text, or [G] position $\pm$ 2 strings apart, or [H] closest preceding/following position (up to four positions away) having a certain ambiguity class in the POS category.*

Treebank Experiments

Characteristics	# Train	# Test Data		# String Tags		# Token Tags	
Experiment	Strings	Strings	Tokens	Total	Anno.	Total	Anno.
PATB Part 2 Prototype	122 556	19 683	23 074	2 031	852	317	242
PATB Part 3	320 998	19 283	22 690	2 864	1 251	391	314
PADT MorphoTrees	106 887	19 253	22 547	3 164	927	378	265
PATB Part 1	120 045	19 339	22 131	884	534	165	143
PATB Part 1 Revised	125 392	19 363	22 104	2 226	785	401	271

Performance	Per String	Per Token				
Experiment	Full (40)	Full (10)	**POS**	Lemma	Tknz++	**Tknz**
PATB Part 2 Prototype	87.88	89.31	96.46	92.33	99.31	99.51
PATB Part 3	86.82	88.17	95.25	89.91	97.52	98.60
PADT MorphoTrees	87.73	89.24	96.02	90.64	97.71	99.25
PATB Part 1	96.85	96.99	97.37	92.75	97.47	99.37
PATB Part 1 Revised	88.13	89.16	95.57	90.27	97.13	98.86

Treebank Experiments

Characteristics	# Train	# Test Data		# String Tags		# Token Tags	
Experiment	Strings	Strings	Tokens	Total	Anno.	Total	Anno.
PATB Part 2 Prototype	122 556	19 683	23 074	2 031	852	317	242
PATB Part 3	320 998	19 283	22 690	2 864	1 251	391	314
PADT MorphoTrees	106 887	19 253	22 547	3 164	927	378	265
PATB Part 1	120 045	19 339	22 131	884	534	165	143
PATB Part 1 Revised	125 392	19 363	22 104	2 226	785	401	271

Performance	Per String	Per Token				
Experiment	Full (40)	Full (10)	**POS**	Lemma	Tknz++	**Tknz**
PATB Part 2 Prototype	87.88	89.31	96.46	92.33	99.31	99.51
PATB Part 3	86.82	88.17	95.25	89.91	97.52	98.60
PADT MorphoTrees	87.73	89.24	96.02	90.64	97.71	99.25
PATB Part 1	96.85	96.99	97.37	92.75	97.47	99.37
PATB Part 1 Revised	88.13	89.16	95.57	90.27	97.13	98.86

Treebank Experiments

Characteristics	# Train	# Test Data		# String Tags		# Token Tags	
Experiment	Strings	Strings	Tokens	Total	Anno.	Total	Anno.
PATB Part 2 Prototype	122 556	19 683	23 074	2 031	852	317	242
PATB Part 3	320 998	19 283	22 690	2 864	1 251	391	314
PADT MorphoTrees	106 887	19 253	22 547	3 164	927	378	265
PATB Part 1	120 045	19 339	22 131	884	534	165	143
PATB Part 1 Revised	125 392	19 363	22 104	2 226	785	401	271

Performance	Per String	Per Token				
Experiment	Full (40)	Full (10)	**POS**	Lemma	Tknz++	**Tknz**
PATB Part 2 Prototype	87.88	89.31	96.46	92.33	99.31	99.51
PATB Part 3	86.82	88.17	95.25	89.91	97.52	98.60
PADT MorphoTrees	87.73	89.24	96.02	90.64	97.71	99.25
PATB Part 1	96.85	96.99	97.37	92.75	97.47	99.37
PATB Part 1 Revised	88.13	89.16	95.57	90.27	97.13	98.86

PATB Part 2 Prototype

Performance	Per String	Per Token				
Experiment	Full (40)	Full (10)	**POS**	Lemma	Tknz++	**Tknz**
PATB Part 2 Prototype	87.88	89.31	96.46	92.33	99.31	99.51
PATB Part 3	86.82	88.17	95.25	89.91	97.52	98.60
PADT MorphoTrees	87.73	89.24	96.02	90.64	97.71	99.25
PATB Part 1	96.85	96.99	97.37	92.75	97.47	99.37
PATB Part 1 Revised	88.13	89.16	95.57	90.27	97.13	98.86

- The analyses do not seem to overgenerate for orthographical variation (Buckwalter, 2004) too much yet (note the similar Tknz++ and Tknz)
- Habash and Rambow (2005) report 96.5 % accuracy in POS tagging for the comparable dataset and tagset, counting, unlike us, only the well-tokenized data

PATB Part 3

Performance	Per String	Per Token				
Experiment	Full (40)	Full (10)	**POS**	Lemma	Tknz++	**Tknz**
PATB Part 2 Prototype	87.88	89.31	96.46	92.33	99.31	99.51
PATB Part 3	86.82	88.17	95.25	89.91	97.52	98.60
PADT MorphoTrees	87.73	89.24	96.02	90.64	97.71	99.25
PATB Part 1	96.85	96.99	97.37	92.75	97.47	99.37
PATB Part 1 Revised	88.13	89.16	95.57	90.27	97.13	98.86

- Brings the advanced features of Buckwalter's morphology, including complete vocalization (with case and mood endings), extended lexicon, and finer tags for verbs and particles
- Therefore, the mapping into the approximation also improved, and the complexity of the tagset largely increased compared to that of the prototype

PADT MorphoTrees

Performance	Per String	Per Token				
Experiment	Full (40)	Full (10)	**POS**	Lemma	Tknz++	**Tknz**
PATB Part 2 Prototype	87.88	89.31	96.46	92.33	99.31	99.51
PATB Part 3	86.82	88.17	95.25	89.91	97.52	98.60
PADT MorphoTrees	87.73	89.24	96.02	90.64	97.71	99.25
PATB Part 1	96.85	96.99	97.37	92.75	97.47	99.37
PATB Part 1 Revised	88.13	89.16	95.57	90.27	97.13	98.86

- Due to the nature of MorphoTrees, where long-dependency relations between tokens may be weakened and some values in tags expanded for the sake of more precise annotations, certain token combinations may be listed in the format for the tagger that the analyzer would not produce
- MorphoTrees are the 'purest' available approximation of the Functional Arabic Morphology

PATB Part 1

Performance	Per String	Per Token				
Experiment	Full (40)	Full (10)	**POS**	Lemma	Tknz++	**Tknz**
PATB Part 2 Prototype	87.88	89.31	96.46	92.33	99.31	99.51
PATB Part 3	86.82	88.17	95.25	89.91	97.52	98.60
PADT MorphoTrees	87.73	89.24	96.02	90.64	97.71	99.25
PATB Part 1	96.85	96.99	97.37	92.75	97.47	99.37
PATB Part 1 Revised	88.13	89.16	95.57	90.27	97.13	98.86

- These annotations are morphologically most 'impoverished', but have been used by other researchers (Diab et al., 2004; Habash and Rambow, 2005) to train very successful taggers based on support vector machines
- Habash and Rambow (2005) reach 96.2 % of accuracy in full token tagging and 98.1 % of POS accuracy, which are results well comparable to ours

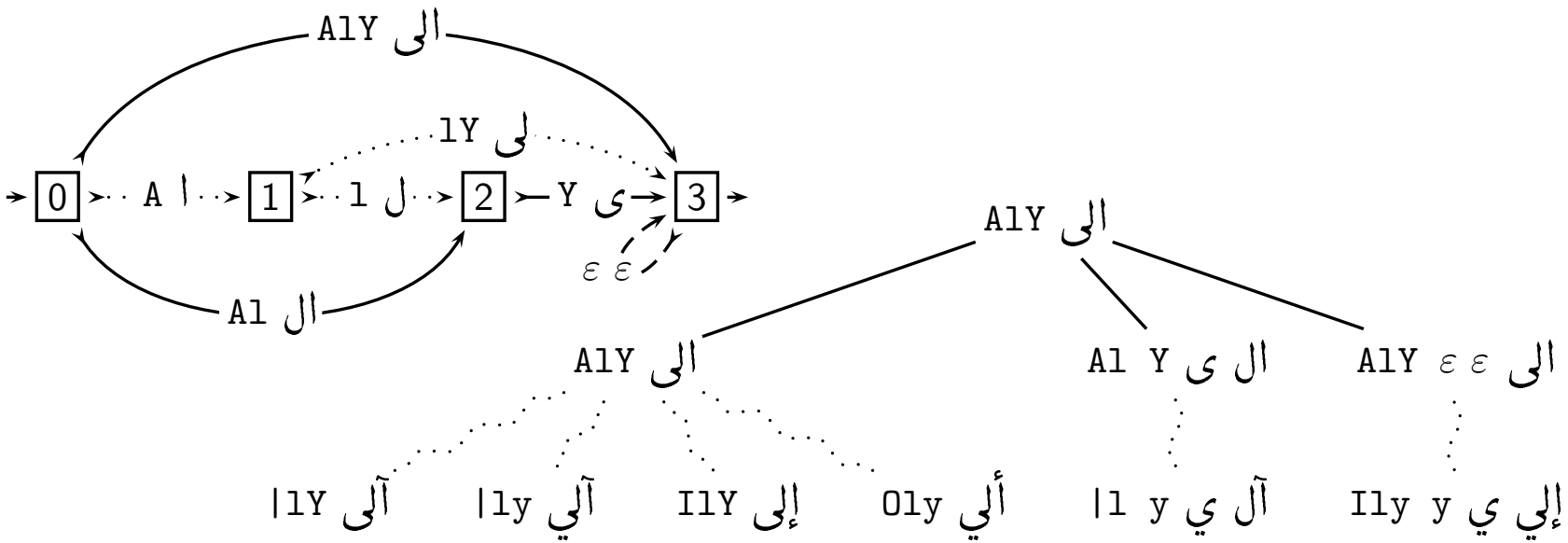
PATB Part 1 Revised

Performance	Per String	Per Token				
Experiment	Full (40)	Full (10)	**POS**	Lemma	Tknz++	**Tknz**
PATB Part 2 Prototype	87.88	89.31	96.46	92.33	99.31	99.51
PATB Part 3	86.82	88.17	95.25	89.91	97.52	98.60
PADT MorphoTrees	87.73	89.24	96.02	90.64	97.71	99.25
PATB Part 1	96.85	96.99	97.37	92.75	97.47	99.37
PATB Part 1 Revised	88.13	89.16	95.57	90.27	97.13	98.86

- Revised with a complete coverage of the lexicon and including the advanced features of the morphology
- Used by Smith et al. (2005) for the development of their log-linear source–channel tagging model achieving overwhelming results (accuracy) — 96.1 % in full string tag disambiguation, 95.4 % in the restoration of morphs, and 94.6 % in assigning one representative lemma per input string

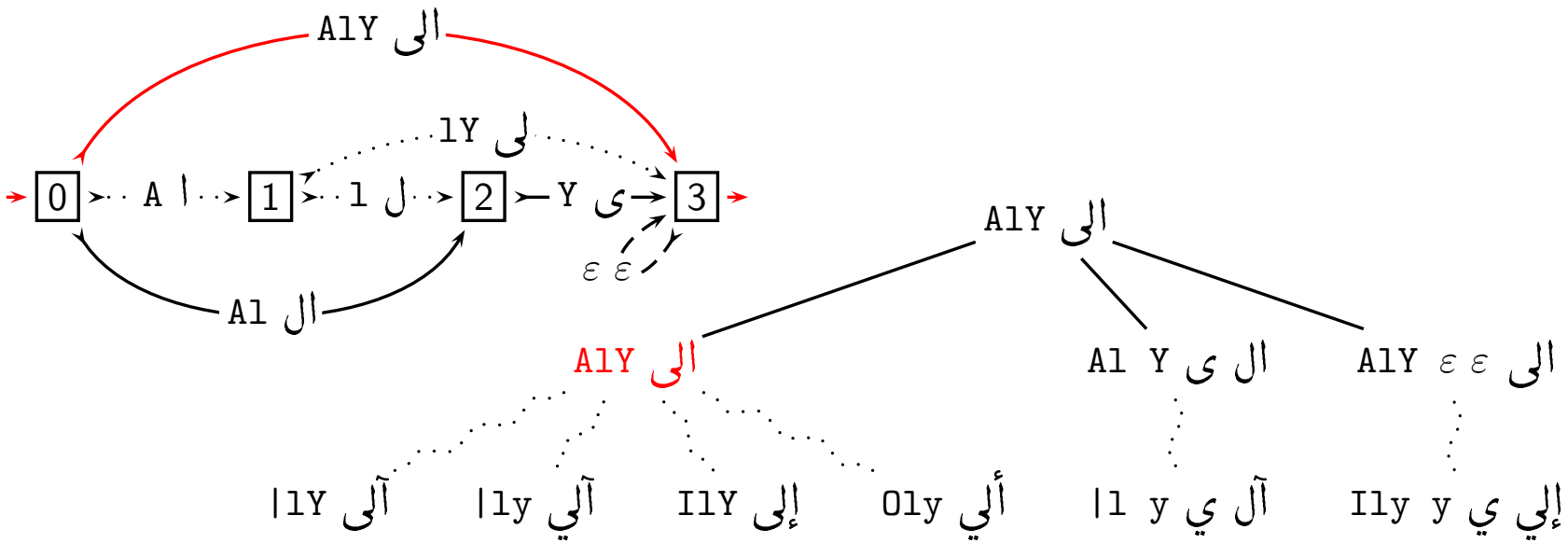
Tknz versus Tknz++

We introduce two measures for tokenization. Tknz is close to the evaluations in (Habash and Rambow, 2005; Diab et al., 2004) which only check the partitioning determined by finding token boundaries between the characters of the original string, and do not, unlike Tknz++, require the tokenization to faithfully reconstruct the canonical non-vocalized forms of tokens, as in MorphoTrees.



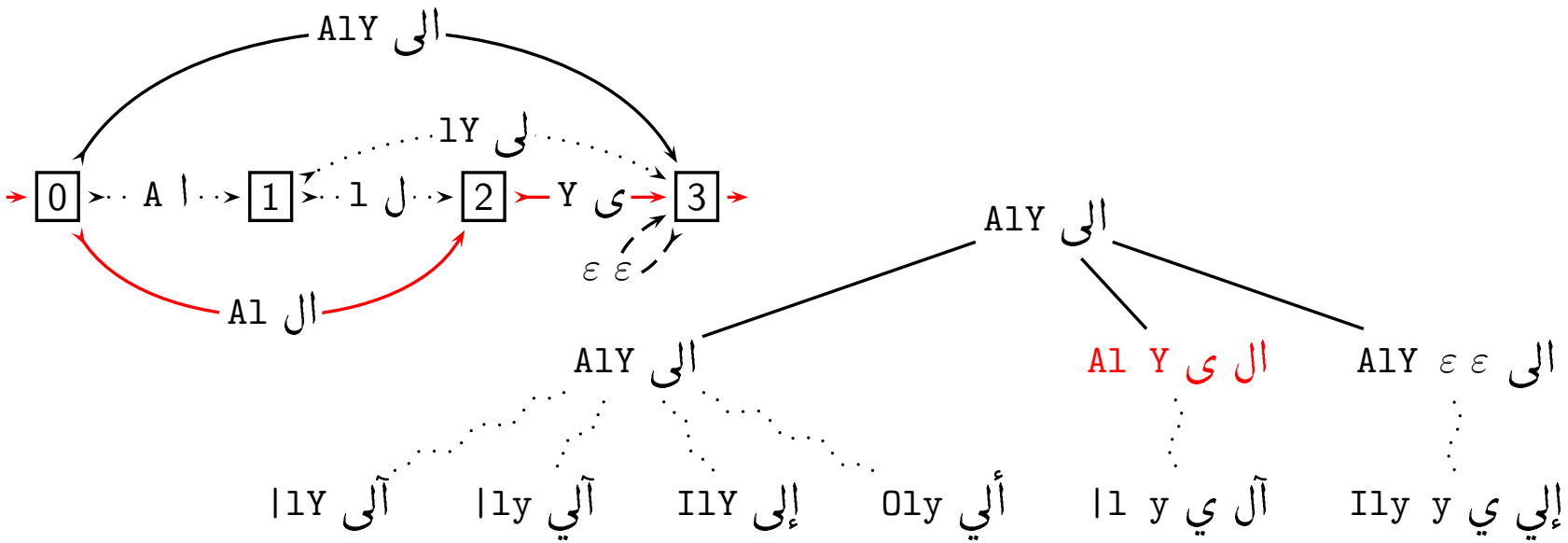
Tknz versus Tknz++

We introduce two measures for tokenization. Tknz is close to the evaluations in (Habash and Rambow, 2005; Diab et al., 2004) which only check the partitioning determined by finding token boundaries between the characters of the original string, and do not, unlike Tknz++, require the tokenization to faithfully reconstruct the canonical non-vocalized forms of tokens, as in MorphoTrees.



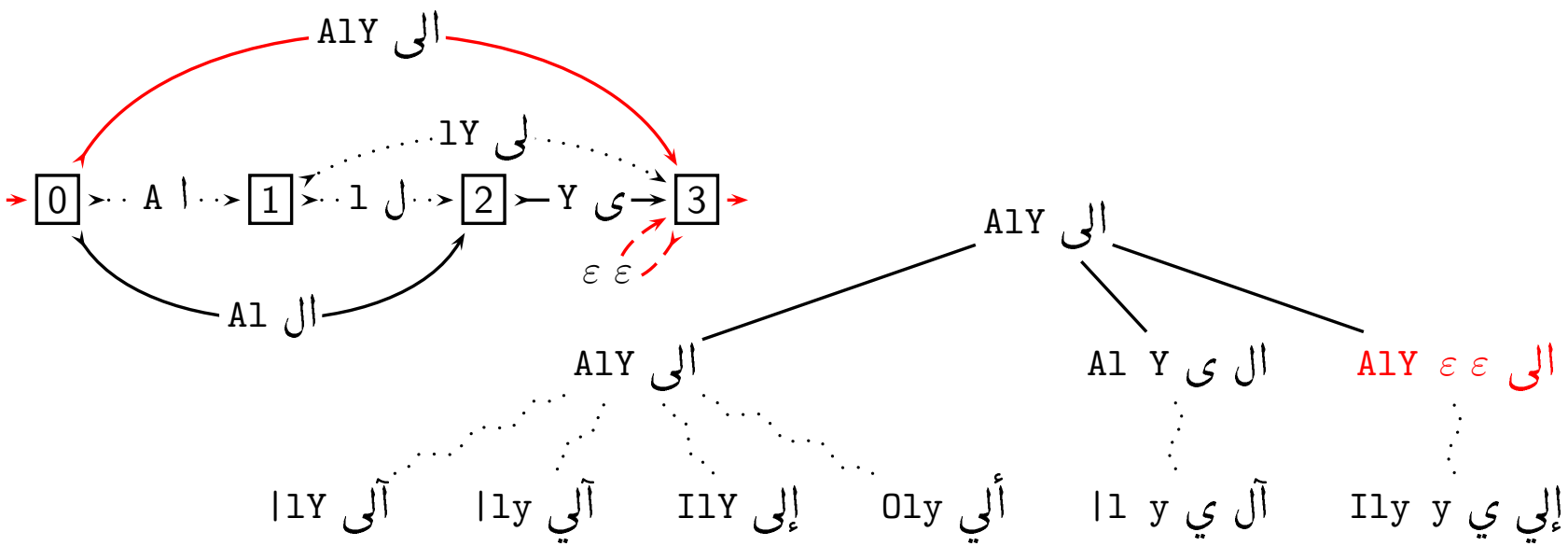
Tknz versus Tknz++

We introduce two measures for tokenization. Tknz is close to the evaluations in (Habash and Rambow, 2005; Diab et al., 2004) which only check the partitioning determined by finding token boundaries between the characters of the original string, and do not, unlike Tknz++, require the tokenization to faithfully reconstruct the canonical non-vocalized forms of tokens, as in MorphoTrees.



Tknz versus Tknz++

We introduce two measures for tokenization. Tknz is close to the evaluations in (Habash and Rambow, 2005; Diab et al., 2004) which only check the partitioning determined by finding token boundaries between the characters of the original string, and do not, unlike Tknz++, require the tokenization to faithfully reconstruct the canonical non-vocalized forms of tokens, as in MorphoTrees.



Conclusions

We have presented five versions of the feature-based tagger of Arabic, developed gradually on all the data of the Penn Arabic Treebank and the Prague Arabic Dependency Treebank. Using the experience with other inflectional languages, we prefer the functional treatment of the morphology of Arabic, which we now only approximate. The pure description with respect to syntactic tokens and their relevant, functional grammatical categories is being further pursued and implemented.

The results of our tagger rank competitively high in the field (Habash and Rambow, 2005). Full morphological tagging is expected to improve with the increasing 'functionality' of the data. Note that applying the conditionally-estimated context-based models set forth in (Smith et al., 2005) to such data is certainly possible and promising, too. The issue of unknown words and lemmatization have only received little attention in our tagger, and can be well improved.

Conclusions

We have presented five versions of the feature-based tagger of Arabic, developed gradually on all the data of the Penn Arabic Treebank and the Prague Arabic Dependency Treebank. Using the experience with other inflectional languages, we prefer the functional treatment of the morphology of Arabic, which we now only approximate. The pure description with respect to syntactic tokens and their relevant, functional grammatical categories is being further pursued and implemented.

The results of our tagger rank competitively high in the field (Habash and Rambow, 2005). Full morphological tagging is expected to improve with the increasing 'functionality' of the data. Note that applying the conditionally-estimated context-based models set forth in (Smith et al., 2005) to such data is certainly possible and promising, too. The issue of unknown words and lemmatization have only received little attention in our tagger, and can be well improved.

Conclusions

We have presented five versions of the feature-based tagger of Arabic, developed gradually on all the data of the Penn Arabic Treebank and the Prague Arabic Dependency Treebank. Using the experience with other inflectional languages, we prefer the functional treatment of the morphology of Arabic, which we now only approximate. The pure description with respect to syntactic tokens and their relevant, functional grammatical categories is being further pursued and implemented.

The results of our tagger rank competitively high in the field (Habash and Rambow, 2005). Full morphological tagging is expected to improve with the increasing 'functionality' of the data. Note that applying the conditionally-estimated context-based models set forth in (Smith et al., 2005) to such data is certainly possible and promising, too. The issue of unknown words and lemmatization have only received little attention in our tagger, and can be well improved.

Conclusions

We have presented five versions of the feature-based tagger of Arabic, developed gradually on all the data of the Penn Arabic Treebank and the Prague Arabic Dependency Treebank. Using the experience with other inflectional languages, we prefer the functional treatment of the morphology of Arabic, which we now only approximate. The pure description with respect to syntactic tokens and their relevant, functional grammatical categories is being further pursued and implemented.

The results of our tagger rank competitively high in the field (Habash and Rambow, 2005). Full morphological tagging is expected to improve with the increasing 'functionality' of the data. Note that applying the conditionally-estimated context-based models set forth in (Smith et al., 2005) to such data is certainly possible and promising, too. The issue of unknown words and lemmatization have only received little attention in our tagger, and can be well improved.

Conclusions

We have presented five versions of the feature-based tagger of Arabic, developed gradually on all the data of the Penn Arabic Treebank and the Prague Arabic Dependency Treebank. Using the experience with other inflectional languages, we prefer the functional treatment of the morphology of Arabic, which we now only approximate. The pure description with respect to syntactic tokens and their relevant, functional grammatical categories is being further pursued and implemented.

The results of our tagger rank competitively high in the field (Habash and Rambow, 2005). Full morphological tagging is expected to improve with the increasing 'functionality' of the data. Note that applying the conditionally-estimated context-based models set forth in (Smith et al., 2005) to such data is certainly possible and promising, too. The issue of unknown words and lemmatization have only received little attention in our tagger, and can be well improved.

Conclusions

We have presented five versions of the feature-based tagger of Arabic, developed gradually on all the data of the Penn Arabic Treebank and the Prague Arabic Dependency Treebank. Using the experience with other inflectional languages, we prefer the functional treatment of the morphology of Arabic, which we now only approximate. The pure description with respect to syntactic tokens and their relevant, functional grammatical categories is being further pursued and implemented.

The results of our tagger rank competitively high in the field (Habash and Rambow, 2005). Full morphological tagging is expected to improve with the increasing 'functionality' of the data. Note that applying the conditionally-estimated context-based models set forth in (Smith et al., 2005) to such data is certainly possible and promising, too. The issue of unknown words and lemmatization have only received little attention in our tagger, and can be well improved.

References I

Tim Buckwalter. Issues in Arabic Orthography and Morphology Analysis. In *Proceedings of the COLING 2004 Workshop on Computational Approaches to Arabic Script-based Languages*, pages 31–34, Geneva, 2004.

Mona Diab, Kadri Hacioglu, and Daniel Jurafsky. Automatic Tagging of Arabic Text: From Raw Text to Base Phrase Chunks. In *HLT-NAACL 2004: Short Papers*, pages 149–152, Boston, 2004.

Nizar Habash and Owen Rambow. 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 2005*, pages 573–580, Ann Arbor, 2005.

Jan Hajič and Barbora Hladká. Tagging Inflective Languages: Prediction of Morphological Categories for a Rich, Structured Tagset. In *Proceedings of COLING-ACL 1998*, pages 483–490, Montreal, 1998.

Jan Hajič, Otakar Smrž, Petr Zemánek, Jan Šnaidauf, and Emanuel Beška. Prague Arabic Dependency Treebank: Development in Data and Tools. In *NEMLAR International Conference on Arabic Language Resources and Tools*, pages 110–117, Cairo, 2004.

References II

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

Mohamed Maamouri, Ann Bies, Tim Buckwalter, and Wigdan Mekki. The Penn Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus. In *NEMLAR International Conference on Arabic Language Resources and Tools*, pages 102–109, Cairo, 2004.

Noah A. Smith, David A. Smith, and Roy W. Tromble. Context-Based Morphological Disambiguation with Random Fields. In *Proceedings of HLT-EMNLP 2005*, pages 475–482, Vancouver, 2005.

Otakar Smrž and Petr Pajas. MorphoTrees of Arabic and Their Annotation in the TrEd Environment. In *NEMLAR International Conference on Arabic Language Resources and Tools*, pages 38–41, Cairo, 2004.

Acknowledgement

This research was supported by the Ministry of Education of the Czech Republic, project MSM0021620838, by the Grant Agency of Charles University in Prague, project 207-10/203333, and through the Fulbright-Masaryk Fellowship of the Fulbright Commission in the Czech Republic.