An Unsupervised Method for Uncovering Morphological Chains

Karthik Narasimhan
Regina Barzilay
Tommi Jaakkola

CSAIL, Massachusetts Institute of Technology

Morphological Chains

Morphological Chains

Chains to model the formation of words.

Morphological Chains

Chains to model the formation of words.

paint → painting → paintings

Morphological Chains

Chains to model the formation of words.

paint → painting → paintings

Richer representation than traditional scenarios

Morphological Chains

Chains to model the formation of words.

paint → painting → paintings

Richer representation than traditional scenarios

Segmentation

Morphological Chains

Chains to model the formation of words.

paint → painting → paintings

Richer representation than traditional scenarios

Segmentation

Paradigms

Morphological Chains

Chains to model the formation of words.

paint → painting → paintings

Richer representation than traditional scenarios

Segmentation

Paradigms

....

Our Approach

Core Idea: Unsupervised discriminative model over pairs of words in the chain.

paint → painting

Our Approach

Core Idea: Unsupervised discriminative model over pairs of words in the chain.

paint → painting

- **Orthographic features**
  Morfessor (Goldwater and Johnson, 2004; Creutz and Lagus, 2007), Poon et al., 2009, Dreyer and Eisner, 2009, Sirts and Goldwater, 2013

Our Approach

Core Idea: Unsupervised discriminative model over pairs of words in the chain.

paint → painting

- **Orthographic features**
  Morfessor (Goldwater and Johnson, 2004; Creutz and Lagus, 2007), Poon et al., 2009, Dreyer and Eisner, 2009, Sirts and Goldwater, 2013

- **Semantic features**
  Schone and Jurafsky, 2000; Baroni et al., 2002

Our Approach

Core Idea: Unsupervised discriminative model over pairs of words in the chain.

paint → painting

- **Orthographic features**
  Morfessor (Goldwater and Johnson, 2004; Creutz and Lagus, 2007), Poon et al., 2009, Dreyer and Eisner, 2009, Sirts and Goldwater, 2013

- **Semantic features**
  Schone and Jurafsky, 2000; Baroni et al., 2002

- Handle **transformations.** (plan → planning)

Textual Cues

Textual Cues

Orthographic

Textual Cues

Orthographic

Patterns in the characters forming words.

Textual Cues

Orthographic

Patterns in the characters forming words.

paint	pain
paints	pains
painted	pained

Textual Cues

Orthographic

Patterns in the characters forming words.

paint paints painted	pain pains pained
pain paint	ran rant

Textual Cues

Orthographic

Patterns in the characters forming words.

paint
paints
painted

pain
pains
pained

pain
paint

ran
rant

Semantic

Meaning embedded as vectors.

Textual Cues

Orthographic

Patterns in the characters forming words.

paint paints painted	pain pains pained
pain paint	ran rant

Semantic

Meaning embedded as vectors.

A	B	cos(A,B)
paint	paints	0.68
paint	painted	0.60
pain	pains	0.60
pain	paint	0.11
ran	rant	0.09

Textual Cues

Orthographic

Patterns in the characters forming words.

paint paints painted	pain pains pained
pain paint	ran rant

Semantic

Meaning embedded as vectors.

A	B	cos(A,B)
paint	paints	0.68
paint	painted	0.60
pain	pains	0.60
pain	paint	0.11
ran	rant	0.09

Task Setup

Training

Unannotated word list with frequencies

a	395134
ability	17793
able	56802
about	524355

Word Vector Learning

Large text corpus

Wikipedia

Multiple chains possible for a word.

nation → national → international → internationally

nation → national → nationally → internationally

Multiple chains possible for a word.

nation → national → international → internationally

nation → national → nationally → internationally

Different chains can share word pairs.

nation → national → international → internationally

nation → national → nationalize

Independence Assumption

Independence Assumption

Treat word-parent pairs separately

Independence Assumption

Treat word-parent pairs separately

national

Word (**w**)

Independence Assumption

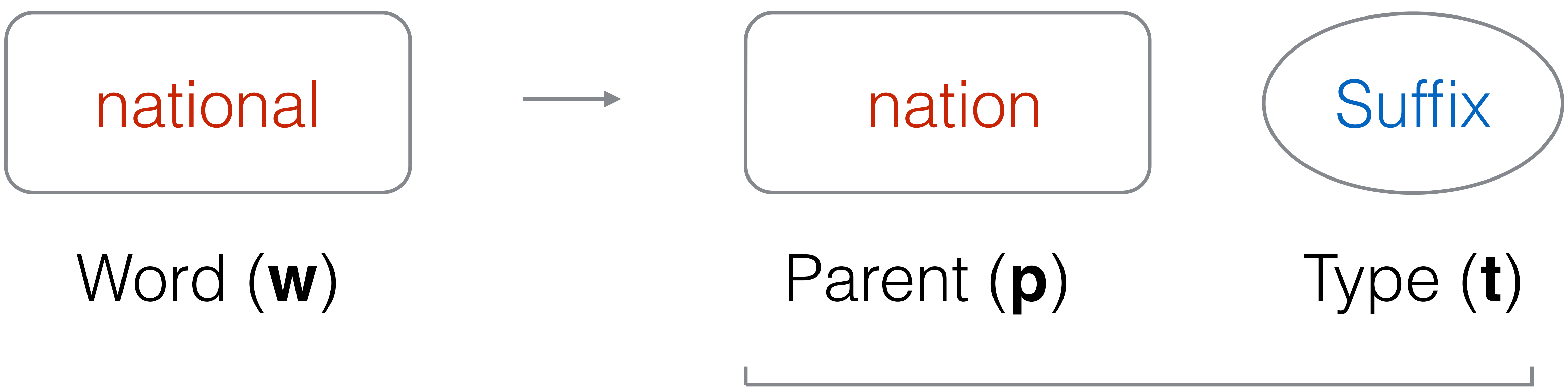
Treat word-parent pairs separately

national → nation | Suffix

Word (**w**) | Parent (**p**) | Type (**t**)

Independence Assumption

Treat word-parent pairs separately



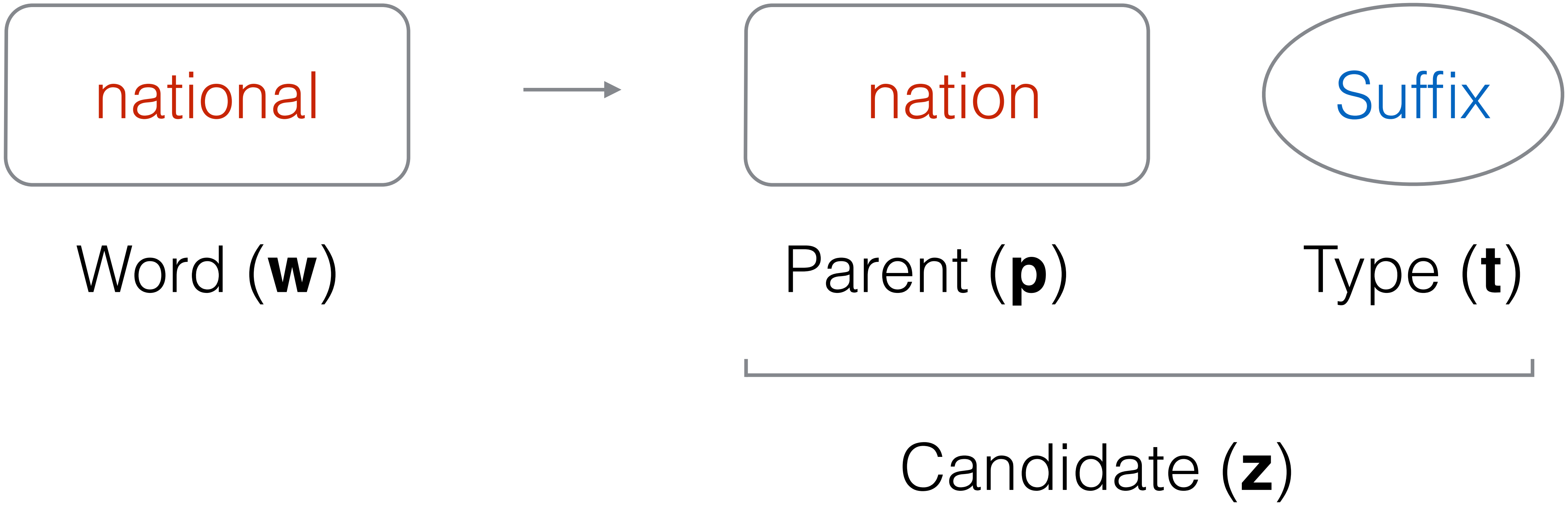
Independence Assumption

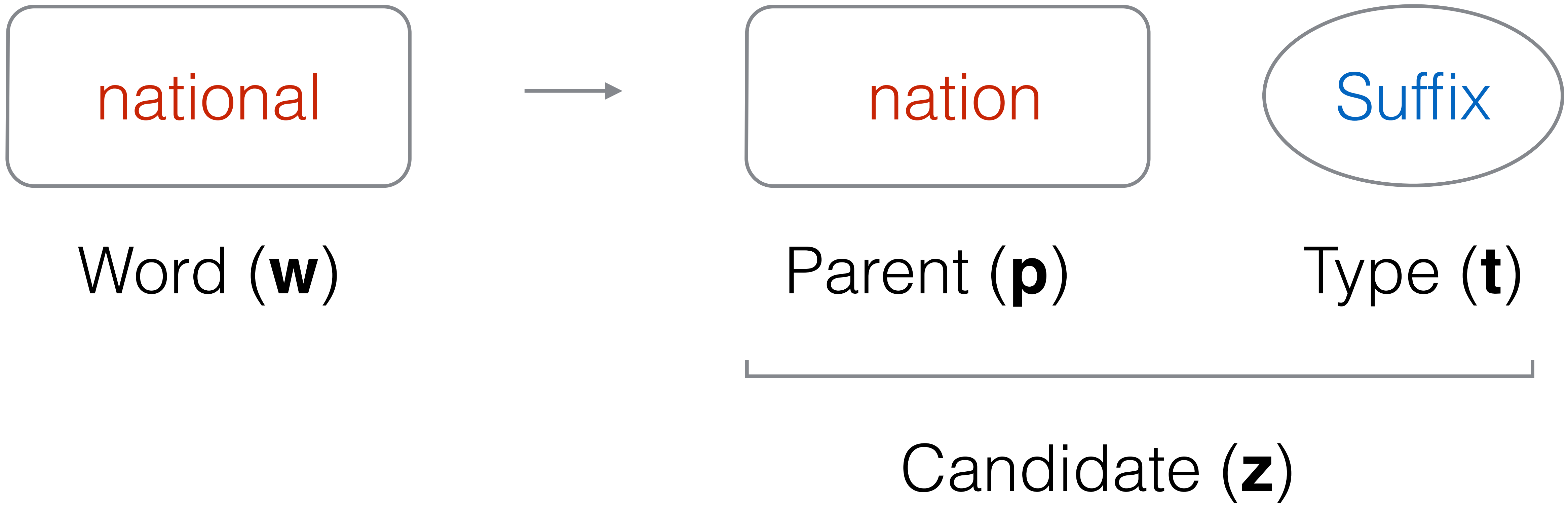
Treat word-parent pairs separately

national → nation Suffix

Word (**w**) Parent (**p**) Type (**t**)

Candidate (**z**)


national
nation
Suffix
Word (**w**)
Parent (**p**)
Type (**t**)
Candidate (**z**)



$$P(w, z) \propto e^{\theta \cdot \phi(w, z)}$$

national → nation Suffix

Word (**w**) Parent (**p**) Type (**t**)

Candidate (**z**)

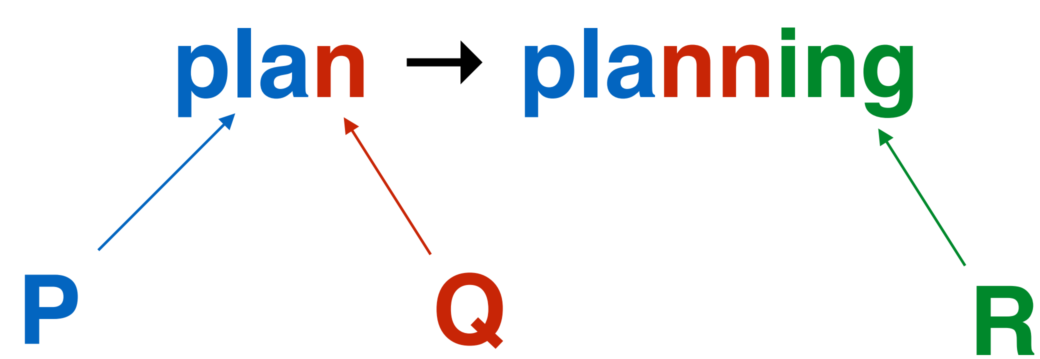
$$P(w, z) \propto e^{\theta \cdot \phi(w,z)}$$

Types - **Prefix, Suffix, Transformations, Stop.**

Transformations

- Templates for handling changes in stem during addition of affixes.

  - Repetition template: **PQ → PQQR** (for each Q in alphabet). Ex.



  - Feature template for each transformation.

Transformation types

Transformation types

3 different transformations:

Transformation types

3 different transformations:

- *Repetition* (plan → planning)

Transformation types

3 different transformations:

- *Repetition* (plan → planning)
- *Deletion* (decide→deciding)

Transformation types

3 different transformations:

- *Repetition* (plan → planning)
- *Deletion* (decide→deciding)
- *Modification* (carry → carried)

Transformation types

3 different transformations:

- *Repetition* (plan → planning)
- *Deletion* (decide→deciding)
- *Modification* (carry → carried)

Trade-off between types of transformation and computational tractability.

Transformation types

3 different transformations:

- *Repetition* (plan → planning)
- *Deletion* (decide→deciding)
- *Modification* (carry → carried)

Trade-off between types of transformation and computational tractability.

- These three do well for a range of languages and are computationally tractable: max $O(|\Sigma|^2)$ for alphabet Σ

Features $\phi(w,z)$

Features $\phi(w,z)$

Orthographic

Features $\phi(w,z)$

Orthographic

- **Affixes**: Indicator feature for top affixes

Features ϕ(w,z)

Orthographic

- **Affixes**: Indicator feature for top affixes
- **Affix Correlation**: pairs of affixes sharing set of stems *(inter-, re-), (under-, over-)*

Features $\phi(w,z)$

Orthographic

- **Affixes**: Indicator feature for top affixes
- **Affix Correlation**: pairs of affixes sharing set of stems *(inter-, re-), (under-, over-)*
- **Word freq.** of parent

Features $\phi(w,z)$

Orthographic

- **Affixes**: Indicator feature for top affixes
- **Affix Correlation**: pairs of affixes sharing set of stems *(inter-, re-), (under-, over-)*
- **Word freq.** of parent
- Transformation types with character bigrams

Features $\phi(w,z)$

Orthographic

- **Affixes**: Indicator feature for top affixes
- **Affix Correlation**: pairs of affixes sharing set of stems *(inter-, re-), (under-, over-)*
- **Word freq.** of parent
- Transformation types with character bigrams

Semantic

- **Cosine similarity** between word vectors of word and parent

Features ϕ(w,z)

Orthographic

- **Affixes**: Indicator feature for top affixes
- **Affix Correlation**: pairs of affixes sharing set of stems *(inter-, re-), (under-, over-)*
- **Word freq.** of parent
- Transformation types with character bigrams

Semantic

- **Cosine similarity** between word vectors of word and parent

Segment	Cosine Similarity
p	0.095
pl	-0.037
pla	-0.041
play	**0.580**
playe	0.000
player	1.000

Cosine similarity with *player*

Learning

Learning

- Objective:

Learning

- Objective:

$$\prod_w P(w) = \prod_w \sum_z P(w, z) = \prod_w \sum_z \frac{e^{\theta \cdot \phi(w,z)}}{\sum_{w' \in \Sigma^*, z'} e^{\theta \cdot \phi(w', z')}}$$

Learning

- Objective:

$$\prod_{w} P(w) = \prod_{w} \sum_{z} P(w, z) = \prod_{w} \sum_{z} \frac{e^{\theta \cdot \phi(w,z)}}{\sum_{w' \in \Sigma^*, z'} e^{\theta \cdot \phi(w',z')}}$$

- Optimize likelihood using convex optimization: LBFGS-B (with regularization)

Learning

- Objective:

$$\prod_w P(w) = \prod_w \sum_z P(w,z) = \prod_w \sum_z \frac{e^{\theta \cdot \phi(w,z)}}{\sum_{w' \in \Sigma^*, z'} e^{\theta \cdot \phi(w',z')}}$$

- Optimize likelihood using convex optimization: LBFGS-B (with regularization)

- Not tractable - requires summing over all possible strings in alphabet to calculate normalization constant, Z.

Learning

- Objective:

$$\prod_w P(w) = \prod_w \sum_z P(w,z) = \prod_w \sum_z \frac{e^{\theta\cdot\phi(w,z)}}{\sum_{w' \in \Sigma^*, z'} e^{\theta\cdot\phi(w',z')}}$$

- Optimize likelihood using convex optimization: LBFGS-B (with regularization)
- Not tractable - requires summing over all possible strings in alphabet to calculate normalization constant, Z.

Contrastive Estimation

Contrastive Estimation

- Instead, we use Contrastive Estimation (Smith and Eisner, 2005):

Contrastive Estimation

- Instead, we use Contrastive Estimation (Smith and Eisner, 2005):
    - Neighborhood of invalid words for each word to take probability mass from.

Contrastive Estimation

- Instead, we use Contrastive Estimation (Smith and Eisner, 2005):
  - Neighborhood of invalid words for each word to take probability mass from.
  - Transpose pair of adjacent chars from first and last k chars in word. (Ex. *painting* → *paintnig*)

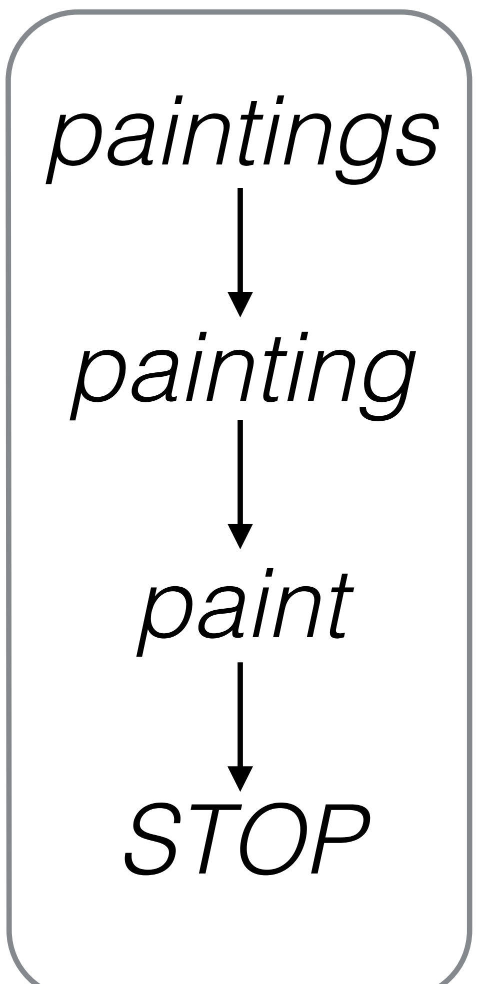
Contrastive Estimation

- Instead, we use Contrastive Estimation (Smith and Eisner, 2005):
  - Neighborhood of invalid words for each word to take probability mass from.
  - Transpose pair of adjacent chars from first and last *k* chars in word. (Ex. *painting* → *paintnig*)

$$P(w, z) = \frac{e^{\theta \cdot \phi(w,z)}}{\sum_{w' \in N(w), z'} e^{\theta \cdot \phi(w', z')}}$$

Prediction

- Predict chain in recursive fashion (argmax parent candidate each time) till stop.



Algorithm 2 Procedure to predict a morphological chain

procedure GETCHAIN(word)
    $candidate \leftarrow \text{PREDICT}(word)$
    $parent, type \leftarrow candidate$
    if $type = STOP$ **then return** $[(word, STOP)]$
    return GETCHAIN(parent)+$[(parent, type)]$

Segmentation Experiments

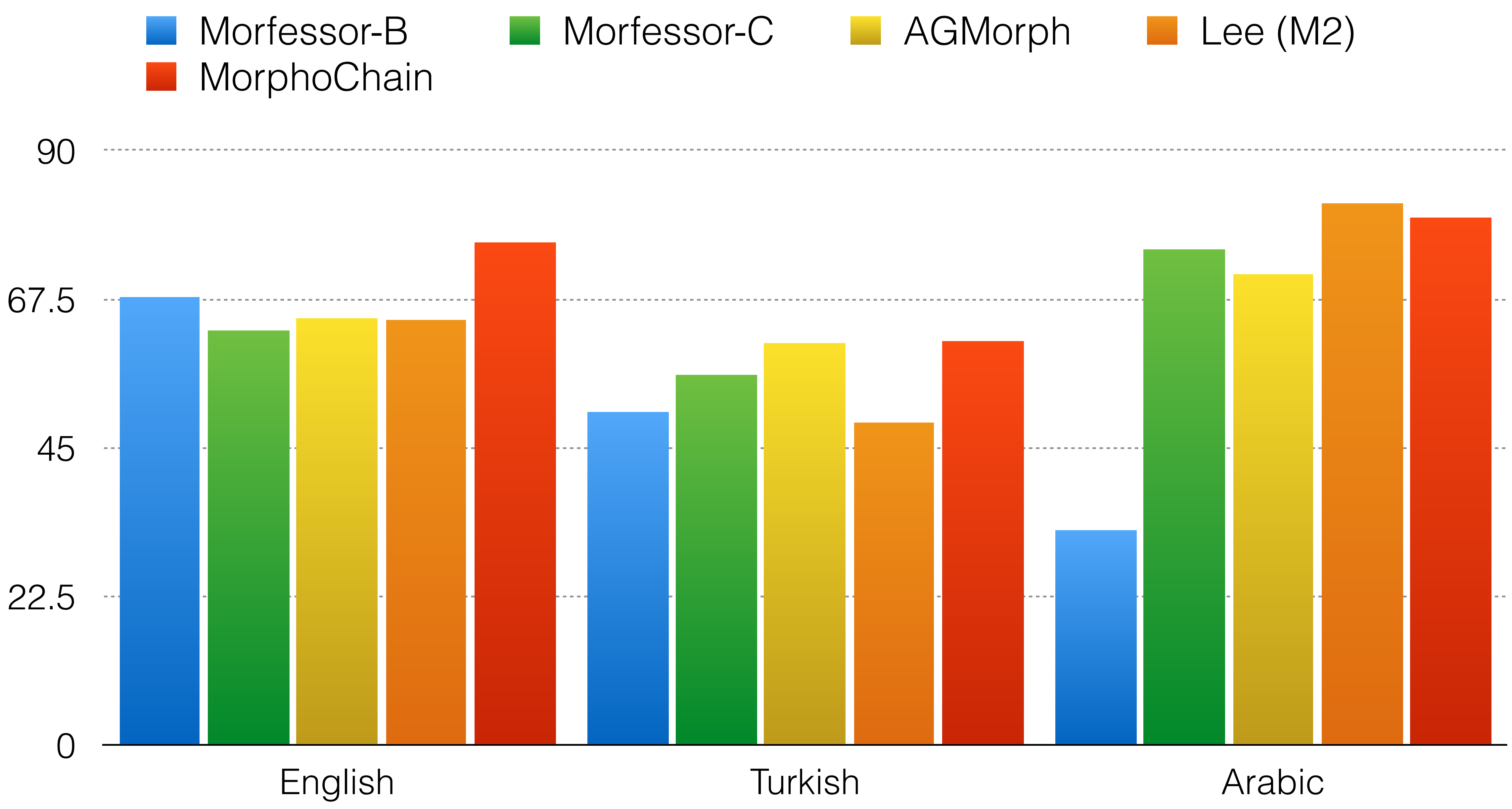
- Three languages - English, Arabic, Turkish

Lang	Train (# words)	Test (# words)	WordVec (# words)
English	MC-10 (878K)	MC-05:10 (2218)	Wikipedia (129M)
Turkish	MC-10 (617K)	MC-05:10 (2534)	BOUN (361M)
Arabic	Gigaword (3.83M)	ATB (119K)	Gigaword (1.22G)

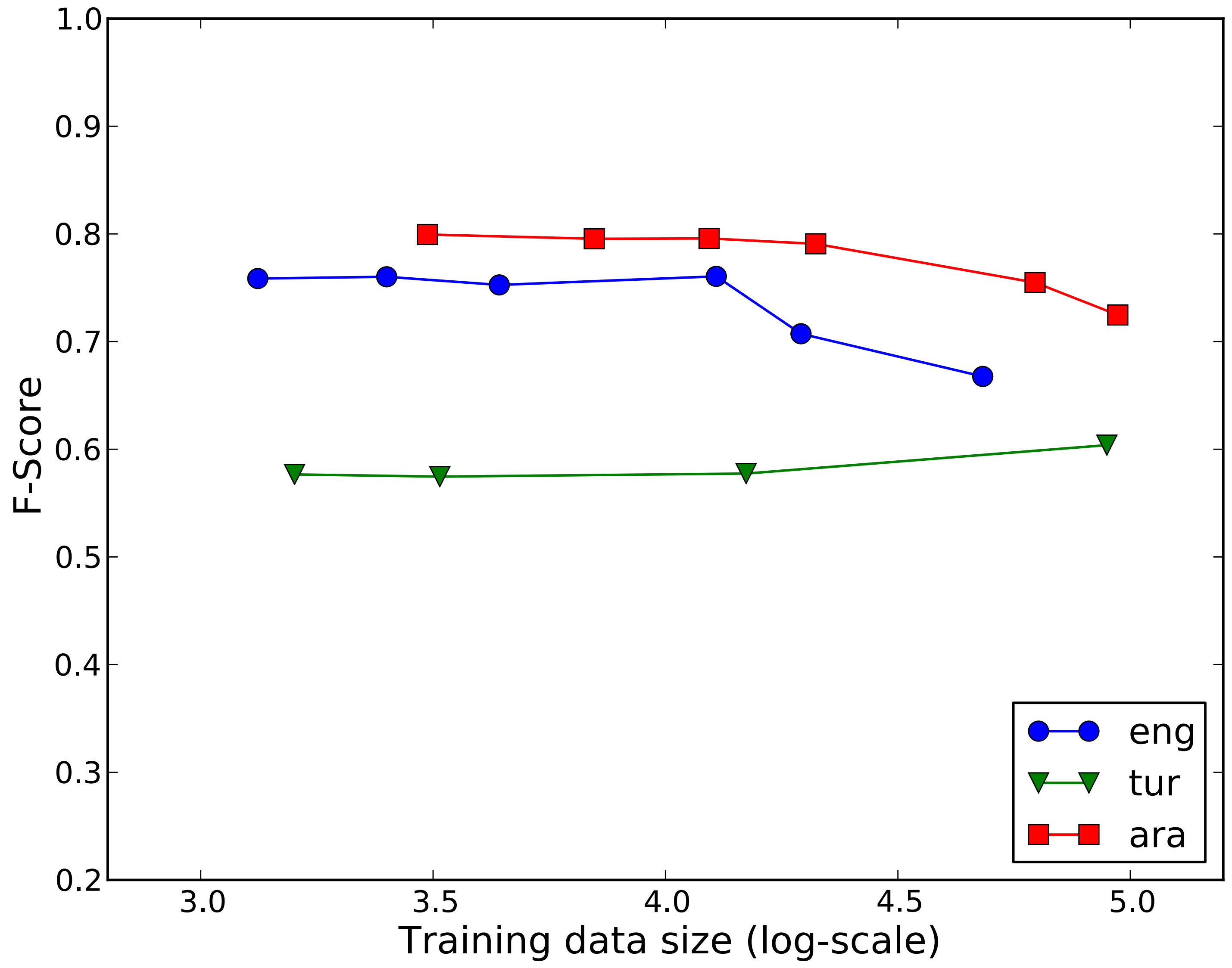
- **Evaluation:** Morphological segmentation - *Precision, Recall, F1* over individual segmentation points

- **Baselines:** Morfessor-Baseline, Morfessor CatMAP, AGMorph (Sirts and Goldwater, 2013) and Lee et al. (2011)

F1 scores on MorphoChallenge



Effect of data size



Affix Analysis

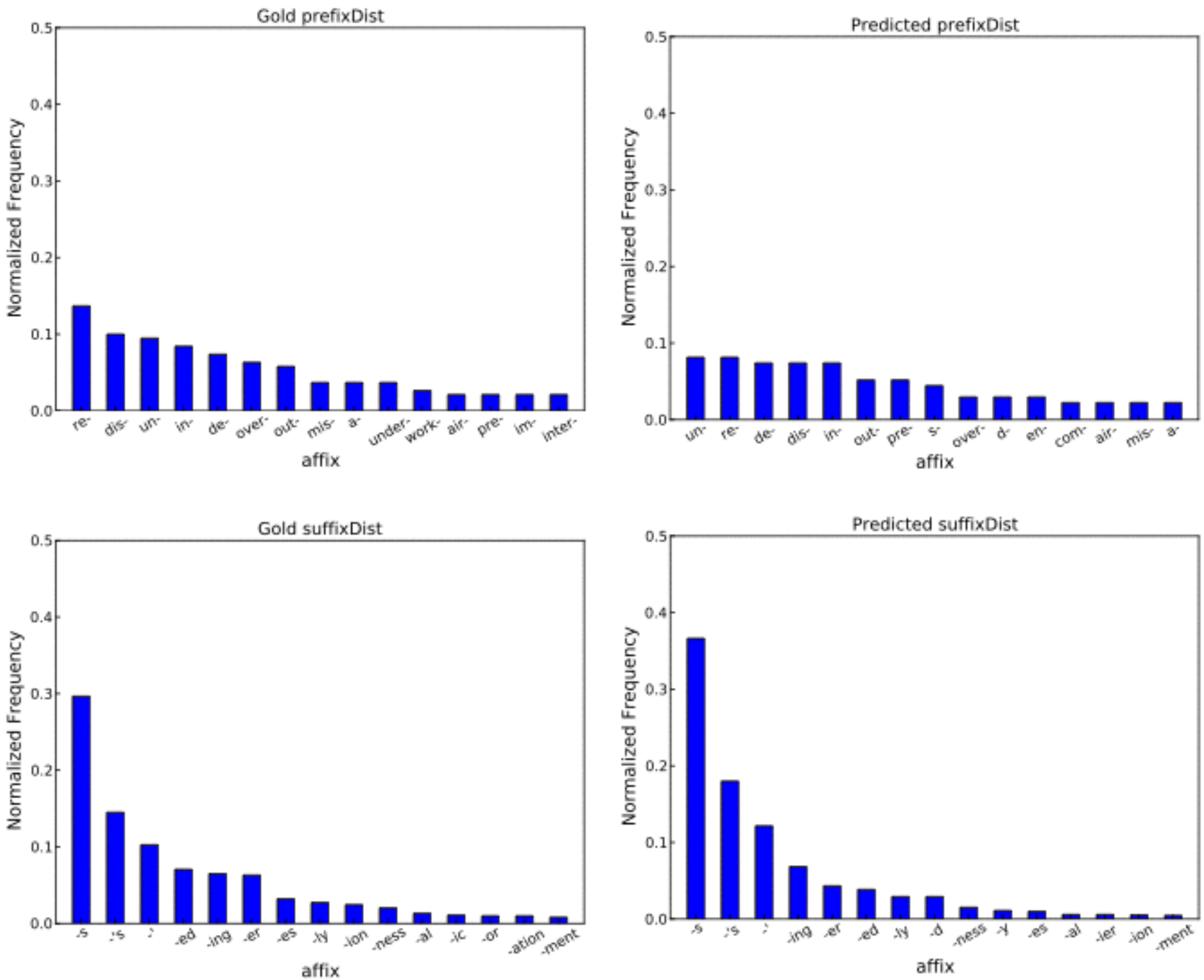


Figure 2: Comparison of gold and predicted frequency distributions of the top 15 affixes for English

Error analysis

- Errors (on a random subset of 50 words per language):

Language	**Over-segment**	**Under-segment**
English	10%	86%
Turkish	12%	78%
Arabic	60%	40%

- Most errors (58%) in Turkish due to parent words not present or having low count.
- Root template morphology of Arabic causes 14% of errors.

Sample segmentations

Correct Segmentations	
Word	**Segmentation**
salvoes	salvo-es
negotiations	negotiat-ion-s
telephotograph	tele-photo-graph
unequivocally	un-equivocal-ly
carsickness's	car-sick-ness-'s

Incorrect Segmentations		
Word	**Predicted**	**Correct**
legacies	legac-ies	lega-ci-es
sterilizing	steriliz-ing	steril-iz-ing
desolating	desolating	desolat-ing
storerooms	storeroom-s	store-room-s
tattlers	tattler-s	tattl-er-s

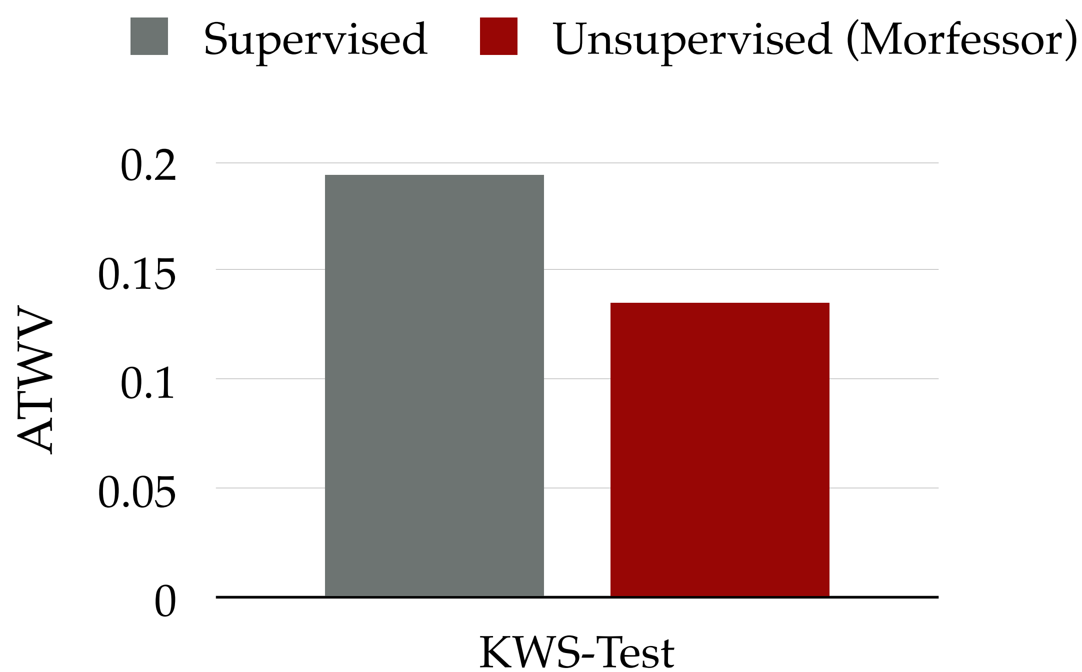
Morphology in Keyword Spotting

Morphology in Keyword Spotting

- Keyword Spotting is the task of identifying keywords in speech utterances
- Major issue: Out of vocabulary words

Morphology in Keyword Spotting

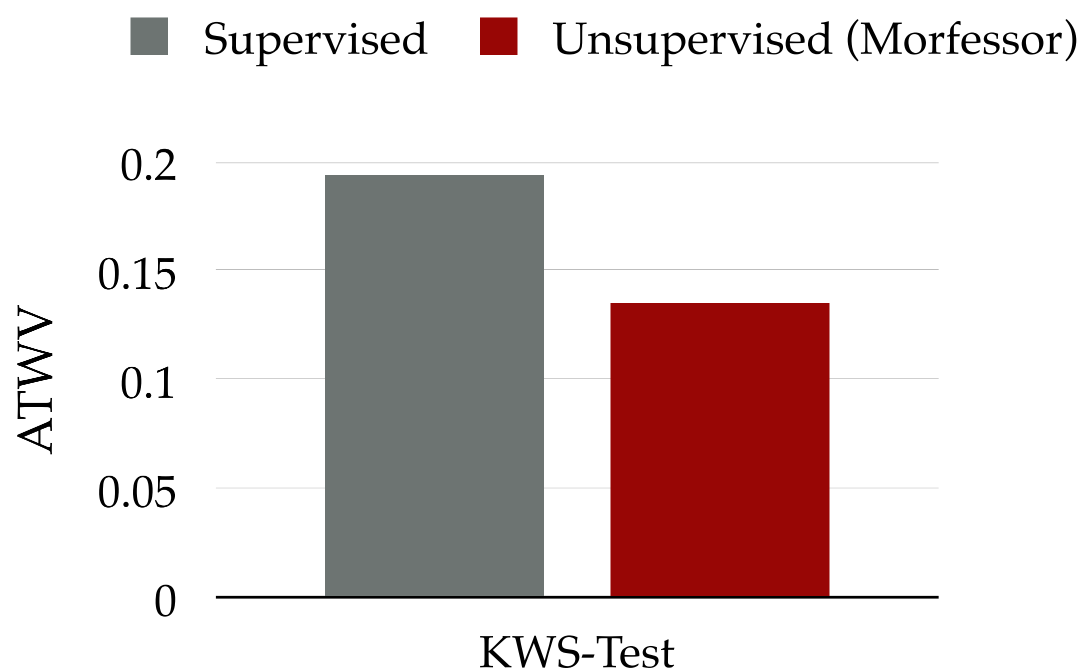
- Keyword Spotting is the task of identifying keywords in speech utterances
- Major issue: Out of vocabulary words



KWS Results on OOV keywords in Turkish LLP (Narasimhan et al., 2014)

Morphology in Keyword Spotting

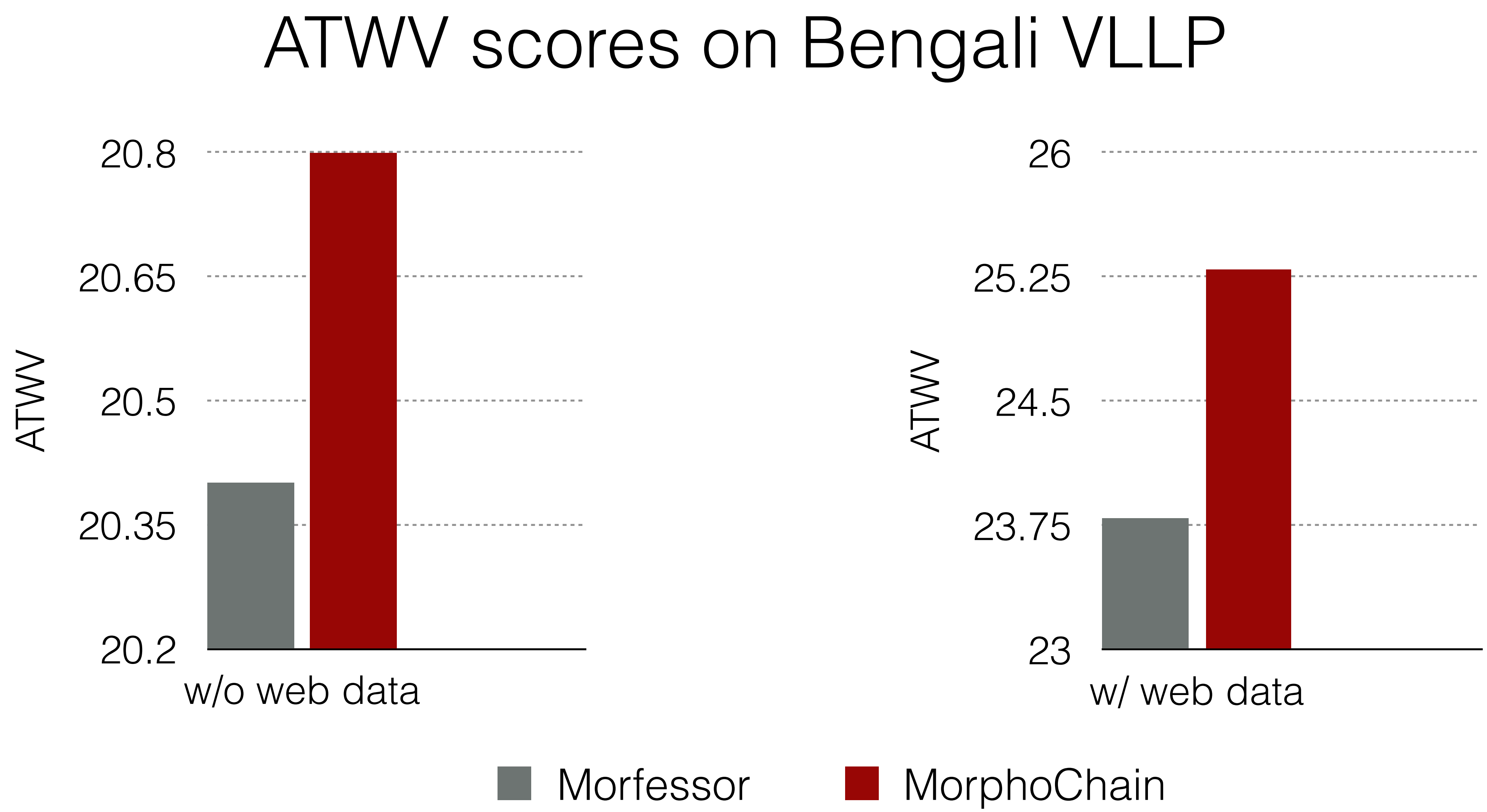
- Keyword Spotting is the task of identifying keywords in speech utterances
- Major issue: Out of vocabulary words



KWS Results on OOV keywords in Turkish LLP (Narasimhan et al., 2014)

- Adding morphemes helps KWS.
- Better morphology can lead to better KWS (supervised vs. unsupervised)
- Need for better unsupervised segmentation.

Morfessor vs MorphoChain for KWS



- MorphoChain outperforms state-of-the-art unsupervised morphological system on KWS

*in collaboration with Damianos Karakos and Rich Schwartz at BBN

Conclusions

- A new method for unsupervised morphological analysis incorporating both orthographic and semantic features.
- Equals or outperforms state-of-the-art systems on morphological segmentation.
- Works well on downstream tasks.

Code: http://people.csail.mit.edu/karthikn/morphochain/