Learning Morphology with **Morfette**

Grzegorz Chrupała[1,2] Georgiana Dinu[2]
Josef van Genabith[1]

[1]National Centre for Language Technology
School of Computing

[2]Department of Computational Linguistics

LREC 2008

Outline

Outline

Approaches to morphological analysis

- Traditional rule-based (finite-state + dictionary lookup)
- Unsupervised learning from raw text
- Supervised learning from annotated corpora
  - Analyze isolated wordforms
  - Analyze word forms in context

Learn model M to assign morphological features and lemmas to each word form in a sentence:

$$M : \mathcal{W}^n \rightarrow (\mathcal{M} \times \Lambda)^n$$

Supervised learning of morphology

- Dictionary or FS morphological analyzer combined with data-driven disambiguation: morphosyntactic tagging:
  - Hajič and Hladká 1998, Hajič 2000, Tufiș 1999, Tufiș and Dragomirescu 2004, Ceausu 2006, Han and Palmer 2004, Habash and Rambow 2005, Hakkani-Tür et al. 2002, Yuret and Türe 2006
- Morphosyntactic tagging followed by lemmatization of unknown words using Inductive Logic Programming
  - Erjavec and Džeroski 2004
- Data-driven context-sensitive lemmatization using a classifier
  - Chrupala 2006

Using a classifier to learn lemmatization

- Learn lemmatization model from running text annotated only with lemmas
- Induce class labels from data
- Use edit script between reversed word forms and lemmas

An edit script of sequences w and w'

sequence of operations which, when applied to sequence w, transform it into sequence w'.

Edit list

Let $w = \textit{pidieron}$ and $w' = \textit{pedir}$. Edit list which transforms w into w':

$$\{\langle \mathrm{D}, i, 2\rangle, \langle \mathrm{I}, e, 3\rangle, \langle \mathrm{D}, e, 5\rangle, \langle \mathrm{D}, o, 7\rangle, \langle \mathrm{D}, n, 8\rangle\}.$$

- Which encodes
  - Delete character *i* at position 2
  - Insert character *e* before position 3
  - ...

Edit list

Let $w = \textit{pidieron}$ and $w' = \textit{pedir}$. Edit list which transforms w into w':

$$\{\langle \mathrm{D}, i, 2\rangle, \langle \mathrm{I}, e, 3\rangle, \langle \mathrm{D}, e, 5\rangle, \langle \mathrm{D}, o, 7\rangle, \langle \mathrm{D}, n, 8\rangle\}.$$

- Which encodes
  - Delete character *i* at position 2
  - Insert character *e* before position 3
  - ...
- Inflectional morphology tends to affect word-endings
- Edit list on reversed strings: better lemma classes
  - *pidieron* : *pedir* :: *repitieron* : *repetir*
  - $\textsc{edit-list}(\textit{pidieron}, \textit{pedir}) \neq \textsc{edit-list}(\textit{repitieron}, \textit{repetir})$
  - But $\textsc{edit-list}(\textit{noreidip}, \textit{ridep}) = \textsc{edit-list}(\textit{noreitiper}, \textit{riteper})$

Outline

Supervised learning of morphology

Morfette

- Architecture
- Features
- Search

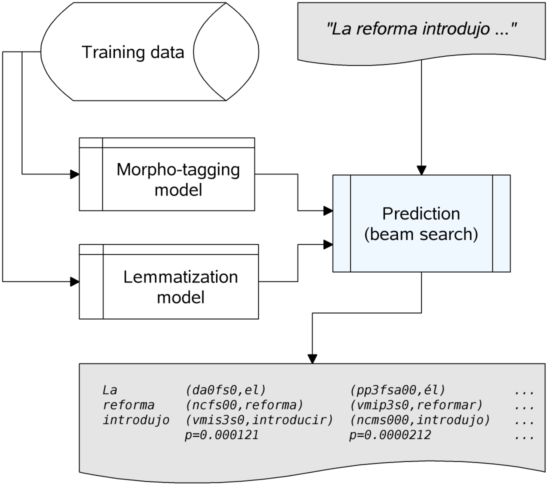
Evaluation and Error Analysis

Conclusion

The Morfette system

- Model:
  - Data driven (trained on annotated corpora)
  - Language independent
  - Modular
- Integrates morphological tagging with lemmatization
  - Both are treated as sequence labeling tasks
  - Lemmatization model uses the lemmatization-as-classification idea
- Predicts probability distributions over sequences of (lemma, morpho-tag) pairs

Architecture



Morpho-tagging and lemmatization models

MaxEnt modeling

$$p(y|x) = \frac{\exp\left(\sum_{i=0}^{d} w_i \Phi(x,y)_i\right)}{\sum_{y' \in Y} \exp\left(\sum_{i=0}^{d} w_i \Phi(x,y')_i\right)} \quad (1)$$

- Use arbitary features of input
- Output probability distribution over the set of labels Y

Feature sets

- Morphological tagging model
  - Lowercased wordform of the focus token
  - Suffixes of length 1..7
  - Prefixes of length 1..5
  - Spelling pattern of the (non-lowercased) wordform
  - Concatenation of the first element of the two previous morpho-tags
  - Lowercased wordform of two previous tokens and of one following token
  - (Predicted) Morpho-tag of two previous tokens
  - (Predicted) Lemma of two previous tokens
  - Set of morpho-tags seen in training data for wordform of next token
- Lemmatization model
  - Lowercased wordform of the focus token
  - Suffixes of length 1..7
  - Prefixes of length 1..5
  - (Predicted) Morpho-tag
  - Spelling pattern of the (non-lowercased) wordform

Prediction: beam search

- For a focus word w_i in context $c \in \mathcal{C}$
  - for each morpho-tag $m \in \mathcal{M}$, the morpho-tagging model gives $P(m|c)$
  - for each lemma-class $l \in \mathcal{L}$, the lemmatization model gives $P(l|c, m)$
- Beam search
  - Keeps n-best sequences of $(m, l) \in \mathcal{M} \times \mathcal{L}$ pairs up to the current position
  - Conditional probability of a candidate sequence for $w_0..w_i$ is given by

$$P(m_0..m_i, l_0..l_i|w_0..w_i) = \tag{2}$$

$$P(l_i|c_i, m_i)P(m_i|c_i)P(m_0..m_{i-1}, l_0..l_{i-1}|w_0..w_{i-1}) \tag{3}$$

Outline

Experimental result – data

- Romanian: MULTEXT-EAST corpus, 13,500 tokens (chapters 1-3) as a test set, 11,800 tokens (chapters 5 and 6) for development and 88,000 tokens (chapters 7-23) for training.
- Spanish: CESS-ECE treebank, 10,000 tokens each for test and development set, and 168,000 tokens for the FULL training set, and 70,000 for the SMALL training set.
- Polish: Korpus Słownika Frekwencyjnego (IPI PAN) 10,000 tokens each for test and development sets, and 219,000 for FULL training set, and 70,000 for the SMALL training set.

Corpus statistics

- Average morpho-tag ambiguity per token
- Percentage of tokens with lemmas identical to word forms

	Avg. morpho-tags	Id. lemmas
Romanian	1.16	58.72%
Spanish	1.46	66.73%
Polish	2.23	44.44%

Experimental results: SMALL

All words

	Morpho-tagging	Lemmatization	Joint
Romanian	96.83	97.78	96.08
Spanish	94.33	97.84	93.83
Polish	81.87	93.29	81.19

Unseen words

	Morpho-tagging	Lemmatization	Joint
Romanian	86.68	82.88	78.50
Spanish	74.79	89.20	71.26
Polish	61.93	76.88	59.17

Comparison to baseline

- Morphological tagging
  - A tagger is generated from training material using MBT (Daelemans et al. 2007)
- Lemmatization
  - For each word in the test set the morpho-tag predicted by MBT is retrieved. If the (word,morpho-tag) pair is in the training set, then it is assigned its predominant lemma; otherwise a lemma identical to the word form is assigned.

Morfette against baseline (FULL)

All words

	Morpho-tagging	Lemmatization	Joint
Romanian	96.83 (+2.34)	97.78 (+4.42)	96.08 (+5.87)
Spanish	95.40 (+2.27)	98.52 (+2.80)	95.02 (+4.32)
Polish	84.91 (+6.49)	95.55 (+7.26)	84.44 (+11.38)

Common sources of errors

- Caused mainly by unknown words/uncommon constructions
  - Named entities (*Chiapas*)
  - Suffix ambiguity (*cruenta lucha*)
- Lack of syntactic (non-local) disambiguation
  - Syncretism (*dziewczyny*)
  - Ambiguous function words (*se, que*)
- Other
  - Annotation problems
  - Prefixal morphology (*bogaty, bogatszy, najbogatszy*)

Outline

Conclusion

- The Morfette is a modular system which integrates morphological tagging and lemmatization
- Both tasks are treated as sequence labeling
- Good performance with no language-specific feature engineering and tuning
- Recent results and work in progress
  - It's hard to beat the **reverse edit list** lemmatization-class
  - But for languages with significant amount of word-initial changes, inducing **richer edit script** versions gives an improvement (Welsh)
    - Encode the bias that inflected word forms tend to consist of roots with prefixes and suffixes
  - Adding features extracted from **lexicons** can give substantial performance gains

Thank you!

Examples lemma-classes for English

English SES	Example		Token freq.	Type freq.
∅	the	→ the	71847	5425
ds0	proles	→ prole	3066	983
dn0 de2	been	→ be	288	1
dd0 de1 dp3	slipped	→ slip	96	22
dh1 dg2 di3 is4 iu4	might	→ must	88	1
da1 iu2	ran	→ run	10	1
do1 ii2	won	→ win	6	1
dy0 dl2	dutifully	→ dutiful	3	3
dd0 d'4 ih5	'eard	→ heard	2	2
dg0 dn1 di2 dd4	nodding	→ nod	1	1
da0 im1 iu1	memoranda	→ memorandum	1	1

Experimental results: FULL

All words

	Morpho-tagging	Lemmatization	Joint
Spanish	95.40 (+1.07)	98.52 (+0.68)	95.02 (+1.19)
Polish	84.91 (+3.04)	95.55 (+2.26)	84.44 (+3.25)

Unseen words

	Morpho-tagging	Lemmatization	Joint
Spanish	75.71 (+4.22)	91.22 (+2.74)	71.84 (+3.99)
Polish	65.87 (+4.33)	81.11 (+4.49)	63.16 (+4.33)

Search algorithm

- For each of N best sequences up to word w_i
  - get morpho-tag distribution for w_i
  - assign probability $P(m_0..m_{i-1}, l_0..l_{i-1}|w_0..w_{i-1})P(m_i|c_i)$
- Keep N best
- For each of N best sequences up to word w_i (including m_i for w_i)
  - get lemma-class distribution for w_i
  - assign probability $P(l_i|c_i, m_i)P(m_i|c_i)P(m_0..m_{i-1}, l_0..l_{i-1}|w_0..w_{i-1})$
- Keep N best
- Advance to word w_{i+1}

To speed up the search both distributions are pre-pruned