Morphological analysis
Kuzman Ganchev
Google Inc.

What is morphology
morphology
mərˈfælədʒi/noun
the study of the forms of things, in particular.
  - LINGUISTICS
  the study of the forms of words.

Parts of speech

What is morphology

- Are these words all related?
  know, knew, knowing, knows?

- What about:
  seeing, sees, see, saw, saws, sawing

- What about:
  eat -> eater; love -> lover
  live -> liver?
Using the wrong variant

- I saw her duck.
- Me saw she duck. *
- I seeing her ducking. *
- I am seeing her ducking.

Agreement

- A cat in a hat.
- The cat in the hat.
- The cats in the hats.
- A cats in a hats. *

Agreement

- John is here. He just arrived.
  - He could refer to John
  - John is here. She just arrived.
  - She wouldn’t usually refer to John.

What is morphology

- This is much more important in free word order languages.
Lemmatization

• Related to morphology
• Mapping a word (in context) to its base form
• I saw a saw:
  see  saw

Example: English

Example: French

Example: Russian
Going deeper
(we’re nog going to)

(1) a. un-lockable = [un- [lock-able]] = that cannot be locked
   Da. unlösig
   Fr. inverrouillable
   Ge. unverschlüsselbar

b. unlock-able = [(un-lock] -able] = that can be unlocked
   Da. oplösebar
   Fr. déverrouillable
   Ge. aufschließbar

(Vikner & Vikner, Hierarchical Morphological Structure and Ambiguity, 2008.)

A few morph features

• Tense
  when something happened: I ate. / I eat.

• Mood
  question/statement/command: Did I eat? I ate. Eat!

• Person
  who are we talking about? I / you / he/she/it

A few morph features

• Voice

• Gender

• Number

• Definiteness
  ... others

A few morph features

• Case
  Grammatical function: English has this for pronouns: I vs me, we vs us, she vs her

• Aspect
  Perfective: She helped him.
  Imperfective: She was helping him.
A few morph features

- **Voice**
  Active vs passive voice:
  I saw you. vs You were seen by me.

- **Gender**
  Grammatical gender - in English most inanimate objects are neuter (it) rather than f/m (she/he)

Ambiguity

- Saw: We saw her. Use a saw.
- A bear bears a burden. Some bears bear a burden.

Quick exercise

Which of these can cause agreement problems in English?

- Tense
- Mood
- Person
- Case
- Aspect
- Voice
- Gender
- Number
- Definiteness

- question/statement/command
- she vs her
- perfective vs imperfective
Quick question

- How would you design a system to predict morphology?
- How would you declare success?

Solution 1: SVM

- Linear classifier
- Hand-design features
- Similar to Maximum Entropy classifier, different learning objective

SVM classifiers

- Linear classifier
  
  \[
  \text{prediction}(x) = \arg \max_y f(x, y) \cdot \theta
  \]

- Objective

\[
\min_{\theta \in \mathbb{R}^d} |\theta| \\
\text{s.t. } f(x_i, y_i) \cdot \theta - f(x_i, y) \cdot \theta \geq 1 \\
\forall i, \forall y \neq y_i
\]
SVM classifiers

- With slack

\[
\min_{\theta \in \mathbb{R}^d, \xi_i \in \mathbb{R}_+} |\theta| + C \sum_i \xi_i \\
\text{s.t. } f(x_i, y_i) \cdot \theta - f(x_i, y) \cdot \theta \geq 1 - \xi_i \\
\forall i, \forall y \neq y_i
\]

- Constant \( C \) is a regularization parameter
- This controls model complexity vs fitting the data

Solution 1: SVM

- Predict for each attribute
- Predict for each analysis

One attribute at a time

Entire label at once

And so on...
SVM encoding

- What are benefits/drawbacks of predicting
  - whole tag
  - value for each attribute

SVM Features

- Take 2 minutes to write down some features

Some Features we came up with...

- word
- digit
- lcword
- capitalization
- bias
- hyphen
- prefixes of length 1..3
- offset -1, 0, 1: suffixes of length 1..3
- offset -2, -1, 0, 1, 2: dictionary features
- offset -3, -2, -1, 0, 1, 2, 3: cluster pairs:
  - cluster offset(-1).cluster
  - cluster offset(1).cluster
  - cluster suffix(length="1")
  - cluster suffix(length="2")
  - cluster suffix(length="3")

Knobs to turn

- Just the slack constant C
What results look like

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Precision</th>
<th>Recall</th>
<th>F-measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>POS</td>
<td>96.71</td>
<td>96.71</td>
<td>96.71</td>
</tr>
<tr>
<td>Case</td>
<td>90.22</td>
<td>90.16</td>
<td>90.19</td>
</tr>
<tr>
<td>Gender</td>
<td>91.97</td>
<td>91.94</td>
<td>91.96</td>
</tr>
<tr>
<td>Number</td>
<td>96.45</td>
<td>96.11</td>
<td>96.28</td>
</tr>
<tr>
<td>Person</td>
<td>96.65</td>
<td>95.49</td>
<td>96.06</td>
</tr>
<tr>
<td>Mood</td>
<td>95.65</td>
<td>93.93</td>
<td>94.79</td>
</tr>
<tr>
<td>Tense</td>
<td>96.12</td>
<td>94.39</td>
<td>95.25</td>
</tr>
<tr>
<td>IsReflexive</td>
<td>98.63</td>
<td>98.82</td>
<td>98.72</td>
</tr>
<tr>
<td>MACRO AVG</td>
<td>95.30</td>
<td>94.69</td>
<td>94.99</td>
</tr>
<tr>
<td>FULL TOKEN</td>
<td>89.24</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Bilinear model

- Start with input and output
- Compute features $f(x)$ and $f'(y)$
- Embed $f(x)$ and $f'(y)$ in joint space
  $\Rightarrow Mf(x)$ and $Mf'(y)$
- Score is dot product:
  $s(x, y) = (Mf(x))^\top Mf'(y)$

Bilinear model

SVM Results
Learning: WSABIE

- Try to rank the correct label above the wrong ones.
- In terms of loss $L(\text{rank}_y(x))$
- Objective is to minimize:

$$\sum_i \sum_{y \neq y_i} L(\text{rank}_{y_i}(x_i)) \max(0, \gamma - s(x_i, y_i) + s(x_i, y))$$

Learning: WSABIE

- Minimizing, we want:
- Add a margin: $\gamma - s(x_i, y_i) + s(x_i, y)$
- If correct label wins by enough, no loss:
  $$\max(0, \gamma - s(x_i, y_i) + s(x_i, y))$$
- Multiply by loss, for every alternate label

$$\sum_{y \neq y_i} L(\text{rank}_{y_i}(x_i)) \max(0, \gamma - s(x_i, y_i) + s(x_i, y))$$

A few more details

- Online learning: how do we do updates: stocastic gradient, adagrad, something else
- What dimensionality for joint space $M$
- Regularization on model parameters
- How to initialize

Knobs to turn

- Learning rate
- Iterations
- Margin
- Initialization
- Regularization
What are the features

- Features of the input:
  - Same as before +
  - Embeddings of words at offset ±3
- Features of the output:
  - One-hot

Input features

- Word features
- Cluster features
- Word Embeddings

Neural Network

- Similar to WSABIE
- Non-linear model
- Also uses Embeddings

WSABIE Results

<table>
<thead>
<tr>
<th></th>
<th>ar</th>
<th>cs</th>
<th>da</th>
<th>de</th>
<th>es</th>
<th>fi</th>
<th>fr</th>
<th>pl</th>
<th>ru</th>
<th>avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Per attribute</td>
<td>80</td>
<td>82</td>
<td>84</td>
<td>86</td>
<td>88</td>
<td>90</td>
<td>92</td>
<td>94</td>
<td>96</td>
<td>98</td>
</tr>
<tr>
<td>Full-tag</td>
<td>90</td>
<td>92</td>
<td>94</td>
<td>96</td>
<td>98</td>
<td>100</td>
<td>102</td>
<td>104</td>
<td>106</td>
<td>108</td>
</tr>
<tr>
<td>Wsabie</td>
<td>85</td>
<td>87</td>
<td>90</td>
<td>92</td>
<td>94</td>
<td>96</td>
<td>98</td>
<td>100</td>
<td>102</td>
<td>104</td>
</tr>
</tbody>
</table>

ar, cs, da, de, es, fi, fr, pl, ru, avg
Neural Network Structure

Output layer

Hidden Layer

Embedding Layer

Features

Input . . . , Berlin, befinded, sich, . . .

Neural Net Learning

- Compute gradient using back-propagation
- Choices:
  - Hidden dimension
  - Embedding dimension
  - Features

Knobs to turn

- What kind of update (sg, adagrad, ...)
- Momentum, Averaging, learning rate
- Hidden + embedding dimensions
- Initialization, decay rate

Grid search

- Reserve 5% of training data: hold-out set
- Train many models
- Evaluate on hold-out set
- Pick best model from hold-out set
Grid Search

- Training hundreds/thousands of models

NN Results

**SVM vs WSABIE**

- WSabie has lower model capacity than linear model

\[
(M f(x))^\top M f'(y) = f(x)^\top M^\top M f'(y) = f(x, y) \cdot \theta
\]

**SVM vs WSABIE**

- Block features
WSABIE vs NN

NN from before

Output
Hidden
Embedding
Features
Input

... Berlin, befinded, sich, ...

WSABIE

No Extra Linear Layer

ReLU
SoftMax
Linear

Different loss

NN from before

Word=befinded
Word=Berlin
Word+1=sich
Cluster=231
Suffix=ed
Features...

No Extra Linear Layer