Viterbi

CS 585, Fall 2016

Introduction to Natural Language Processing

http://people.cs.umass.edu/~brenocon/inlp2016

Brendan O'Connor

College of Information and Computer Sciences

University of Massachusetts Amherst

- Project proposals due tomorrow!
- Today
  - HW3 NB
  - Viterbi
  - Learning: the Perceptron Algorithm
- Next week
  - Tues: CRF / Structured Perceptron
  - Thurs: In-class project work & OH
- HW4 released tomorrow:
  - Part 1 due next week
  - Part 2 due in two weeks

Viterbi

- (notes & worksheet)

(Discrim.) Log-linear models

- The form will generalize to multiclass and sequences...
  - x: Text Data
  - y: Proposed class
  - θ: Feature weights (model parameters)
  - f(x,y): Feature extractor, produces feature vector

dot product notation:

$$Goodness(y) = \sum_i \theta_i f_i(x, y) \qquad \equiv \theta^{\mathsf{T}} f(x, y)$$

(Discrim.) Log-linear models

- The form will generalize to multiclass and sequences...
  - x: Text Data
  - y: Proposed class
  - θ: Feature weights (model parameters)
  - f(x,y): Feature extractor, produces feature vector

dot product notation:

$$Goodness(y) = \sum_i \theta_i f_i(x,y) \qquad \equiv \theta^{\mathsf{T}} f(x,y)$$

$$p(y|x) = \frac{1}{Z} e^{G(y)} \quad \Leftrightarrow \quad \log p(y|x) = C + G(y)$$

(Discrim.) Log-linear models

- The form will generalize to multiclass and sequences...
  - x: Text Data
  - y: Proposed class
  - θ: Feature weights (model parameters)
  - f(x,y): Feature extractor, produces feature vector

dot product notation:

$$Goodness(y) = \sum_i \theta_i f_i(x, y) \qquad \equiv \theta^{\mathsf{T}} f(x, y)$$

$$p(y|x) = \frac{1}{Z} e^{G(y)} \quad \Leftrightarrow \quad \log p(y|x) = C + G(y)$$

Decision rule: $y^{(pred)} = \arg\max_{y'} G(y', x)$

(Discrim.) Log-linear models

- The form will generalize to multiclass and sequences...
  - x: Text Data
  - y: Proposed class
  - θ: Feature weights (model parameters)
  - f(x,y): Feature extractor, produces feature vector

dot product notation:

$$Goodness(y) = \sum_i \theta_i f_i(x, y) \qquad \equiv \theta^{\mathsf{T}} f(x, y)$$

$$p(y|x) = \frac{1}{Z} e^{G(y)} \quad \Leftrightarrow \quad \log p(y|x) = C + G(y)$$

Decision rule: $y^{(pred)} = \arg\max_{y'} G(y', x)$

NB and LogReg can be expressed in this form...

Log-linear notation

$G(y,x) = \theta^T f(x,y)$

f(x,y) based on these feature templates:
key: (class=y AND word=w)
value: count of w (or, indicator...)

θ

```
{"POS_The": +0.01,
 "NEG_The": -0.01,
 "POS_awesome": +2.2,
 "NEG_awesome": -2.2,
...}
```

f(x, POS)

```
{"POS_The": 3,
 "POS_awesome": 7,
 "POS_quizzical": 0,
...}
```

f(x, NEG)

```
{"NEG_The": 3,
 "NEG_awesome": 7,
...}
```

$\theta^T f(x,POS) = ...$

$\theta^T f(x,NEG) = ...$