Computational Linguistics 1 CMSC/LING 723, LBSC 744

STATERSITE OF

Kristy Hollingshead Seitz Institute for Advanced Computer Studies University of Maryland Lecture 13: 13 October 2011

Agenda



Discriminative Training

- Statistical model training involves maximizing some objective function
- For an HMM, we use maximum likelihood training
 Maximize the probability of the training set
- Reduction in errors is the true objective of learning
- Another option is to try to directly optimize error rate or some other closely related objective
- Consider not just truth, but also other candidates

Perceptron

- One approach that has been around since late 60s is the perceptron
- Basic idea:

omputational Linguistics 1

- Find the best scoring analysis
- (e.g. POS tag sequence)
- Make its score lower, by penalizing its *features*
- $\boldsymbol{\cdot}$ Make the score of the truth better, by rewarding its features
- · Go onto the next example

mputational Linguistics 1

Formal Definition of Perceptron Algorithm

Formally, perceptron approach assumes:

- Training examples (x_i, y_i) for i = 1...N where x_i is the input and y_i is the true output.
- + e.g. $(w_1 \dots w_k, \tau_1 \dots \tau_k)$, where $\tau_1 \dots \tau_k$ is the true tag sequence
- A function GEN which enumerates a set of candidates GEN(*x*) for an input *x*.
- e.g., run the tagger over input word sequence x, to output tagsequence candidates
- A representation Φ mapping each (x, y) ∈ X × Y to a ddimensional *feature* vector Φ(x, y) ∈ R^d.
- A parameter vector $\alpha \in \mathbb{R}^d$.
 - e.g., a vector of weights, one for each feature in $\boldsymbol{\Phi}$

Computational Linguistics 1

Computational Linguistics 1

Perceptron Algorithm

- Inputs: Training examples (x_i, y_i)
- Initialization: Set α = 0
- Algorithm:
- For *t* = 1 . . . *T* , i = 1 . . . *N* Calculate z_i = argmax_{z∈GEN(xi)} Φ(x_i, z) · α
- If $(z_i \neq y_i)$ then $\alpha = \alpha + \Phi(x_i, y_i) \Phi(x_i, z_i)$
- Output: Parameters α

Perceptron: Notes

- Because this technique is optimizing (sequence) error rate, it does not involve a normalization factor
- · Thus, it will overtrain
- i.e. it will do very well on the training set, but not so well on new data, like unsmoothed maximum likelihood
- Techniques exist for controlling overtraining, such as regularization, voting, and averaging
- Perceptron models outperform maximum likelihoodoptimized models on a range of tasks
- POS-tagging, NP-chunking

Computational Linguistics 1

Computational Linguistics

Computational Linguistics 1

Agenda

- Homework
- Supervised Learning Discriminative Training
- CRFs
- Features

Computational Linguistics 1

Midterm Review

Conditional Random Fields (CRFs)

- The perceptron algorithm only pays attention to bestscoring (argmax) path
- What if there were two top analyses, very close in score?
 Should penalize features on both
- · How do we allocate the penalty?
- CRFs are a way to do this, by optimizing the conditional log-likelihood of the truth

Formal Definition of CRFs

• Define a conditional distribution over the members of GEN(*x*) for a given input *x*:

$$p_{ar{lpha}}(y|x) = rac{1}{Z(x,ar{lpha})} \exp{(\Phi(x,y)\cdotar{lpha})}$$

where

$$Z(x,ar{lpha}) = \sum_{oldsymbol{y}\in \operatorname{GEN}(x)} \exp\left(\Phi(x,y)\cdotar{lpha}
ight)$$

• (Can be calculated with forward-backward algorithm!)

The objective function is convex and there is a globally

· Can use general numerical optimization techniques to find

- e.g. for a language modeling project we used a general *limited* memory variable metric method to optimize LL_R from a publically

· The optimizer needs the function value and the derivative

omputational Linguistics

CRF Optimization

optimal solution

the global optimum

available software library

CRF objective function

- Choose $\boldsymbol{\alpha}$ to maximize the conditional log-likelihood of the training data:

$$LL(ar{lpha}) = \sum_{i=1}^N \log p_{ar{lpha}}(y_i|x_i) = \sum_{i=1}^N \left[\Phi(x_i,y_i) \cdot ar{lpha} - \log Z(x_i,ar{lpha})
ight]$$

Use a zero-mean Gaussian prior on the parameters resulting in the regularized objective function:

$$LL_R(ar{lpha}) = \sum_{i=1}^N \left[\Phi(x_i,y_i) \cdot ar{lpha} - \log Z(x_i,ar{lpha})
ight] - rac{||ar{lpha}||^2}{2\sigma^2}$$

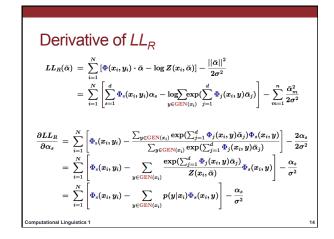
- where the value $\boldsymbol{\sigma}$ is typically estimated on heldout data.

(or gradient)

Computational Linguistics 1

Derivative of *LL_R*: Refresher

Remember the chain rule: $\frac{df(g(x))}{dx} = \frac{df}{dg}\frac{dg}{dx}$ Also remember derivative of (natural) log: $\frac{d\log(x)}{dx} = \frac{1}{x}$ And don't forget the derivative of exp: $\frac{d\exp(ax)}{dx} = a\exp(ax)$



Perceptron vs CRFs

Training time

Computational Linguistics 1

- More expensive (calculating derivative) for CRFs...
- ...but can be parallelized
- Performance

Computational Linguistics 1

 In Sha & Pereira, perceptron performance not statistically significantly different from CRF with same feature set

Agenda

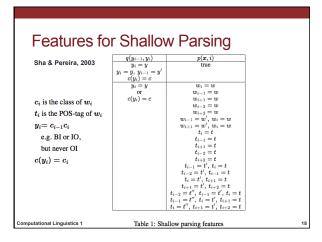
- Homework
- Supervised Learning Discriminative Training
 Perceptron
- CRFs

mputational Linguistics 1

- Features
- Midterm Review

Features (**Φ**)

- · Good feature sets matter a lot
- These discriminative methods allow for easy use of many features
- Unlike HMM based methods
- Examples of feature sets



Computational Linguistics 1

| parkhi, 1993 Condition | Features | |
|---------------------------|-------------------------------------|--------------|
| | | |
| w_i is not rare | $w_i = X$ | $\& t_i = T$ |
| w_i is rare | X is prefix of w_i , $ X \leq 4$ | $\& t_i = T$ |
| | X is suffix of w_i , $ X \le 4$ | $\& t_i = T$ |
| | w_i contains number | $\& t_i = T$ |
| | w_i contains uppercase character | $\& t_i = T$ |
| | w_i contains hyphen | $\& t_i = T$ |
| $\forall w_i$ | $t_{i-1} = X$ | $\& t_i = T$ |
| | $t_{i-2}t_{i-1} = XY$ | $\& t_i = T$ |
| | $w_{i-1} = X$ | $\& t_i = T$ |
| | $w_{i-2} = X$ | $\& t_i = T$ |
| | $w_{i+1} = X$ | $\& t_i = T$ |
| | $w_{i+2} = X$ | $\& t_i = T$ |

| Word: | the | stories | about | well-heele | d communities | and | developer |
|-----------|-----|---------|---|--|---|-----|-----------|
| Tag: | DT | NNS | IN | JJ | NNS | cc | NNS |
| Position: | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| | | | $w_i = abou$ $w_{i-1} = st$ | cories | $ \begin{array}{l} \& \ t_i = \texttt{IN} \\ \& \ t_i = \texttt{IN} \end{array} $ | | |
| | | | $w_i = abou$ $w_{i-1} = st$ $w_{i-2} = th$ | it cories | $\begin{array}{l} \& \ t_i = \texttt{IN} \\ \& \ t_i = \texttt{IN} \\ \& \ t_i = \texttt{IN} \end{array}$ | | |
| | | | $w_i = abou$ $w_{i-1} = st$ $w_{i-2} = th$ $w_{i+1} = we$ | it cories | $ \begin{array}{l} \& t_i = \texttt{IN} \\ \& t_i = \texttt{IN} \end{array} $ | | |
| | | | $w_i = abot$ $w_{i-1} = st$ $w_{i-2} = tt$ $w_{i+1} = we$ $w_{i+2} = cc$ $t_{i-1} = NN$ | it cories he ell-heeled pmmunities | $\& t_i = IN$ $\& t_i = IN$ | | |

| Instar | otic | bot | Foo | turos | | | | |
|--------------------|--------|-----------|------------------------------|-------------------|-------------------|--------|------------|---|
| instal | IUC | licu | i ca | iui es | | | | |
| Word: | the | stories | about | well-heele | d communities | and | developers | |
| Tag: | DT | NNS | IN | JJ | NNS | cc | NNS | |
| Position: | 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
| | | | $w_{i-1} = at$ | | & $t_i = JJ$ | | | |
| | | | | | $\& t_i = JJ$ | | | |
| | | | | | $\& t_i = JJ$ | | | |
| | | | $w_{i+2} = ar$ | | $\& t_i = JJ$ | | | |
| | | | $t_{i-1} = IN$ | | $\& t_i = JJ$ | | | |
| | | | | | $\& t_i = JJ$ | | | |
| | | | $\operatorname{prefix}(w_i)$ | | $\& t_i = JJ$ | | | |
| | | | $\operatorname{prefix}(w_i)$ | | $\& t_i = JJ$ | | | |
| | | | | | $\& t_i = JJ$ | | | |
| | | | | | $\& t_i = JJ$ | | | |
| | | | $\operatorname{suffix}(w_i)$ | | $\& t_i = JJ$ | | | |
| | | | $\operatorname{suffix}(w_i)$ | | $\& t_i = JJ$ | | | |
| | | | $\operatorname{suffix}(w_i)$ | | $\& t_i = JJ$ | | | |
| | | | $suffix(w_i)$ | =eled | $\& t_i = JJ$ | | | |
| | | | w_i contain | ns hyphen | $\& t_i = JJ$ | | | |
| Table | 4: Fea | tures Gen | erated Fro | om h_4 (for tag | ging well-heeled) | from ' | Table 2 | |
| putational Linguis | tics 1 | | | | | | | 2 |

Agenda

- Homework
- Supervised Learning Discriminative Training
- Perceptron
- CRFs

Computational Linguistics 1

- Features
- Midterm Review

Midterm Topics

- Sequences and n-grams
- FSAs, FSTs
 - Construction
- Composition
- Smoothing
- Algorithms
- Interpolation, Backoff
- HMMs
- Tagging
- Viterbi
- Forward-Backward

Computational Linguistics 1

Midterm Format

- Some short answer questions
- Some basic numerical computation
- Questions from the homeworks
- No programming
- Ground rules:
- Work completely independently no communication of any kind
- No communication with the TA or instructor
- Open book, open note. *Not* open internet, except for web pages
 explicitly linked from the class webpage.
- Turn in a hard copy on Tuesday October 25 (or earlier, to Kristy)

Computational Linguistics 1

23

Agenda: Summary

Supervised Learning – Discriminative Training
 Perceptron
 CRFs

25

- Features
- Midterm Review

Computational Linguistics 1