CSCI 544, lecture 12: More Tagging: Brill, Maximum Entropy, Conditional Random Fields, Unsupervised

Ron Artstein

2022-09-29

Administrative notes: deadlines

Written Assignment Peer Grading has been released

Coding Assignment 2 was due today (really!)

Coding Assignment 3 due October 11

Project:

Due Date	Task
September 20	Form project teams (52 teams)
September 20–29	**Initial discussion with TA**
October 4	**Project proposal**
November 3	Project status report
Nov 29/Dec 1	Poster presentations (in class)
December 1	Final report
December 3	Self-evaluation and peer grading

HMM tagging

Tag sentences using co-occurrence statistics: tag–tag, tag–word

Better performance than previous models:

- Most common tag for each word
- Rule-based

HMM can model some linguistic knowledge

Not a very satisfying model

- Not a clear relation to language knowledge

Can we have a model that performs as well as HMM, but is more interpretable?

Brill tagger

Brill (1992): A Simple Rule-Based Part of Speech Tagger

Model that is more understandable than HMMs

Initial tagger learned on 90% of the corpus

- Most common tag for each word
- Capitalized unseen words $\rightarrow$ proper nouns
- Other unseen words $\rightarrow$ most common tag for last three characters

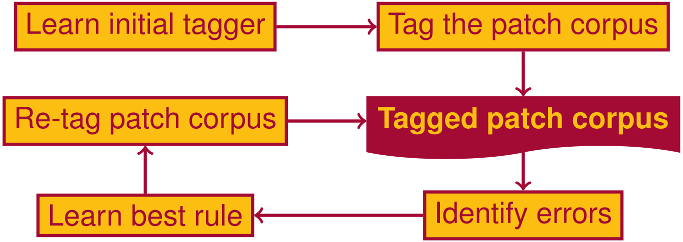
Patch rules learned on 5% of the corpus

- Fix output of the initial tagger
- Learn linguistic rules that minimize error

Test on 5% of the corpus

Brill tagger: iterative process

Each iteration learns one rule, which minimizes error on the patch corpus



Rules must be applied in the order they were learned

Brill tagger rule templates

Templates generate many possible rules

- Change tag A to tag B in context C
  e.g., if previous (or following) word is tagged Z
- Change tag A to tag B if a word has property P
  e.g., if word is capitalized
- Change tag A to tag B if a word in region R has property P
  e.g., if previous (or following) word is capitalized

Procedure for finding best rule

- Find most common error (e.g., noun tagged as verb)
- Find best rule to correct that error

Brill tagger rules

Rules are expressed with Brown corpus tags

- **TO IN NEXT-TAG AT**
  TO: *to* eat, *to* drink (complementizer)
  IN : *to* the market (preposition)
  "A word tagged TO is likely a preposition (IN) when occurring before an article (AT)"
- **VBN VBD PREV-WORD-IS-CAP YES**
  "A word tagged as a passive participle (VBN) is likely a past-tense verb (VBD) when occurring after a capitalized word"

Most rules learned on patch corpus also reduce error on test corpus

Maximum entropy tagger

Ratnaparkhi (1996): A Maximum Entropy Model for Part-Of-Speech Tagging

Demonstrate the advantages of a maximum entropy model

- Maximum entropy = logistic regression

Individually classify each instance

- Sequence properties in the features
  e.g., previous one or two tags, previous and following words
- Spelling features for rare and unseen words

Calculate probabilities for the full path

- Can't remember too many paths: **beam search**
- Individual features only look within small window

Beam search

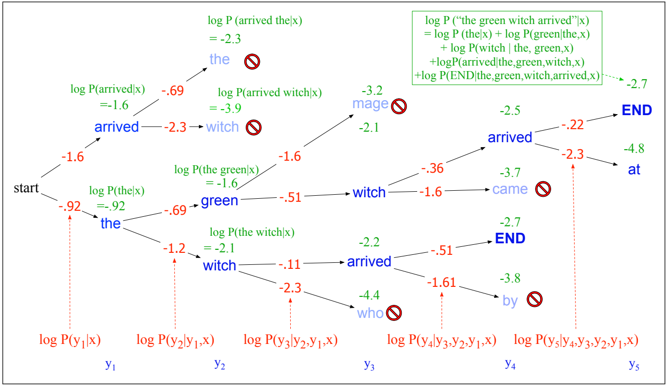


Figure 11.13 Scoring for beam search decoding with a beam width of $k = 2$. We maintain the log probability of each hypothesis in the beam by incrementally adding the logprob of generating each next token. Only the top k paths are extended to the next step.

Jurafsky and Martin, formerly chapter 11, now chapter 10

Maximum entropy tagger (cont.)

Tag dictionary to reduce search space

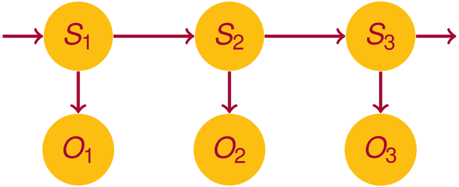
- Known words only consider tags with which they were seen
  - Minimal effect on accuracy
  - Substantial reduction in runtime

Upper bound on possible performance

- Some words are still difficult to tag
  e.g., *about*: preposition or adverb?
- Specialized features for these words show little improvement
- Hypothesis: these words are inconsistently tagged in the Penn Treebank
  - ☛ Tag distribution varies by annotator

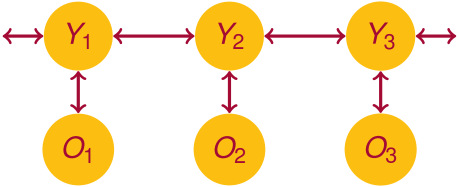
Discriminative sequence labeling

Hidden Markov Models



- Transition probabilities
- Emission probabilities
- Model generates observations
- Viterbi decoding

Conditional Random Fields (Lafferty, McCallum and Pereira 2001)



- Conditional (discriminative) model: $P(\text{labels}|\text{observations})$
- Edge features
- Vertex features

Conditional Random Fields

Conditional model learns probability of entire sequence of labels

- Avoids the **label bias problem** of next-state conditional models

Markov property for CRF: the only labels that affect the probability of a given label are the immediately preceding and following labels

Parameter estimation for CRFs: learn weights for edge features and vertex features

Tested on synthetic data, generated using a second-order Markov model

- Data cannot be learned perfectly by first-order models

Rich features relate observations to labels

- Better performance on out-of-vocabulary items

Field segmentation

Task: Learn to identify information fields in a classified ad (or bibliographic citation, or other similarly structured text) in an **unsupervised** way.

Size | Features
Spacious 1 Bedroom apt. newly remodeled, gated, new appliance,
Features | Location | Rent
new carpet, near public transportion, close to 580 freeway, $500.00
Rent | Contact
Deposit (510)655-0106

Trond Grenager, Dan Klein, and Christopher Manning. Unsupervised Learning of Field Segmentation Models for Information Extraction. ACL 2005

Unsupervised sequence labeling

Unsupervised Hidden Markov Models

Learning probability matrices

- Forward-backward algorithm = special case of EM
  - Expectation: estimate states based on parameters
  - Maximization: estimate parameters based on states

Learning state structure

- Start with fully connected, then learn probabilities

But HMMs can also model part-of-speech tags, which have a very different structure. So which structure will the model learn?

Constraining the model 1

Adjacent words tend to belong to the same field

- Bias transition matrix towards same-state transitions
  σ: probability of staying within the field

$$P(s_t|s_{t-1}) = \begin{cases} \sigma + \dfrac{1-\sigma}{|S|} & \text{if } s_t = s_{t-1} \\ \dfrac{1-\sigma}{|S|} & \text{otherwise} \end{cases}$$

Accuracy 49% $\rightarrow$ 70%

Is this still a Hidden Markov Model?

Constraining the model 2

Punctuation, function words occur in all fields

- Mix general emission with state emission
  α: probability of selecting a common term

$$P(w|s) = \alpha P_{common}(w) + (1 - \alpha) P_{any}(w|s)$$

Accuracy $\rightarrow$ 71%

Is this still a Hidden Markov Model?

Constraining the model 3

State transitions tend to happen after boundary symbols

- Create separate boundary (s^+) and non-boundary (s^-) states
  σ, λ: probability of staying within the field;
  μ: probability of transitioning to boundary

$$P(s'|s^+) = \begin{cases} \sigma + \frac{1-\sigma}{|S^-|} & \text{if } s' = s^- \\ \frac{1-\sigma}{|S^-|} & \text{if } s' \in S^- \setminus s^- \\ 0 & \text{otherwise} \end{cases} \qquad P(s'|s^-) = \begin{cases} (1-\mu)(\lambda + \frac{1-\lambda}{|S^-|}) & \text{if } s' = s^- \\ \mu(\lambda + \frac{1-\lambda}{|S^-|}) & \text{if } s' = s^+ \\ \frac{1-\lambda}{|S^-|} & \text{if } s' \in S^- \setminus s^- \\ 0 & \text{otherwise} \end{cases}$$

Accuracy $\rightarrow$ 73%

Is this still a Hidden Markov Model?

Lessons from unsupervised field segmentation

Performance still below supervised methods

- But that's not the point

Learning the desired structure is possible

- Need more explicit constraints than in supervised learning