Natural Language Processing

Neural Language Models

Instructor: Yangqiu Song

Slides credits: Piotr Mirowski, Noah Smith, Dan Roth, and Chris Manning

Recap: what is a statistical LM?

- A model specifying probability distribution over word sequences
  - p("*Today is Wednesday*") ≈ 0.001
  - p("*Today Wednesday is*") ≈ 0.0000000000001
  - p("*The eigenvalue is positive*") ≈ 0.00001
- It can be regarded as a probabilistic mechanism for "generating" text, thus also called a "generative" model

Probabilistic Language Models

- Probability of a sequence of words:

- Chain rule of probability:

- (n-1)th order Markov assumption

Learning probabilistic language models

- Learn joint likelihood of training sentences under (n-1)th order Markov assumption using **n-grams**

where =(word token , word type in vocabulary)

is word history

- Maximize the **log-likelihood**:
  - Now, given the above reformulation, we will change the notations again (to derive neural language models)!

Featurized Language Models: Re-parameteriazation

- Maximize the **log-likelihood**:
- Assuming a parametric model **w** (note here **w** is the parameter vector similar use as perceptron)

- Consider as features instead of just a sequences of historical words
  - Modeling with log-linear models
  - Moving from generative models to discriminative models

Log-linear Models

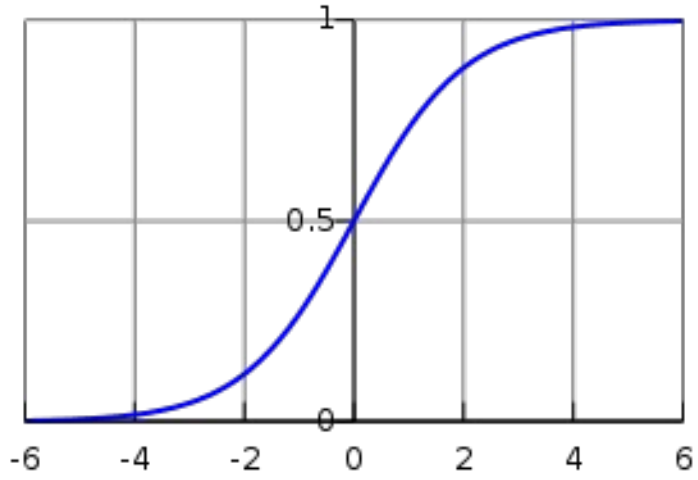
- Linear score
- Nonnegative exponential:
- Normalizer
- Log-linear comes from the fact that

is a constant in

- This is an instance of the family of generalized linear models

Special Case: Logistic Regression

- Consider the case where {+1,-1}
  - where



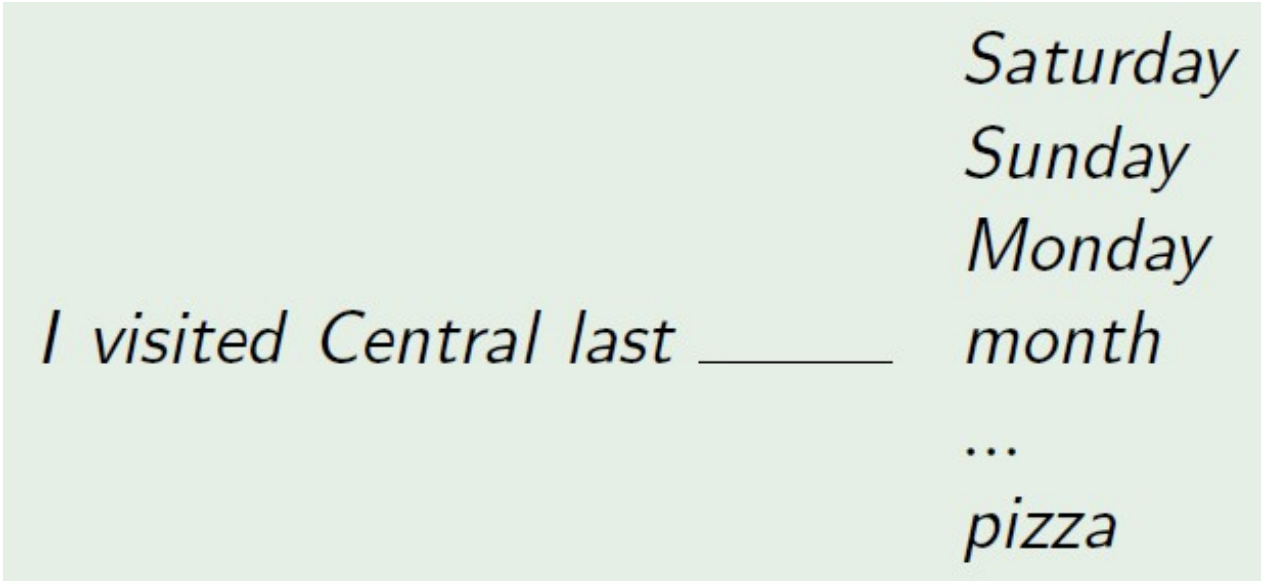
log-linear models are often called multinomial logistic regression (softmax function)

Special Case: N-gram Language Model

- Consider an n-gram language model
  - as n-1 historical words
  - One hot feature vector:
  - (all one vector for all)

- What features are there used in more than traditional n-gram language models?

What features in



- Traditional n-gram features: last ^ Saturday
- “Gappy” n-gram features: Central ^ Saturday
- Spelling features: first character is capitalized
- Class features: whether it is a member of class 132
- Gazetteer features: whether it is listed as a geographic place name, date/time, person name, organization name, etc.

What features in

- You can define any features you want!
  - Too many features, and your model will overfit
    - “Feature selection” methods, e.g., ignoring features with very low counts, can help
  - Too few (good) features, and your model will not learn

Parameter Estimation

- Gradient descent!
  - no closed form as traditional n-gram language models

- Further Reading
  - Berger et al. (1996).A Maximum Entropy Approach to Natural Language Processing.
  - Collins (2011). Course notes for COMS w4705: Log-linear models, MEMMs, and CRFs, 2011.
    - http://www.cs.columbia.edu/~mcollins/crf.pdf
  - Smith (2004). Log-linear models, 2004.
    - https://homes.cs.washington.edu/~nasmith/papers/smith.tut04.pdf

Extension: Neural Language Models

- Feedforward Neural Network Language Model Bengio et al. (2003)
  - A generalization of featurized language model
  - Word embeddings can be learnt!

Feedforward Neural Network Language Model
Bengio et al. (2003)