#ml-papers October 2019

Hierarchical Topic Models and the Nested Chinese Restaurant Process

Section 1: Current State

Current Topic Modeling tools are too Rigid

- Need a set “K” number of topics before modeling
- Data can continue to accrue and new topics should be considered with new data.

We construct a topic hierarchy from data

- Generative probabilistic model for hierarchical structures where **every hierarchy is a random variable**
- Extension of the Chinese Restaurant Process to hierarchies

Section 2: Chinese Restaurant Process

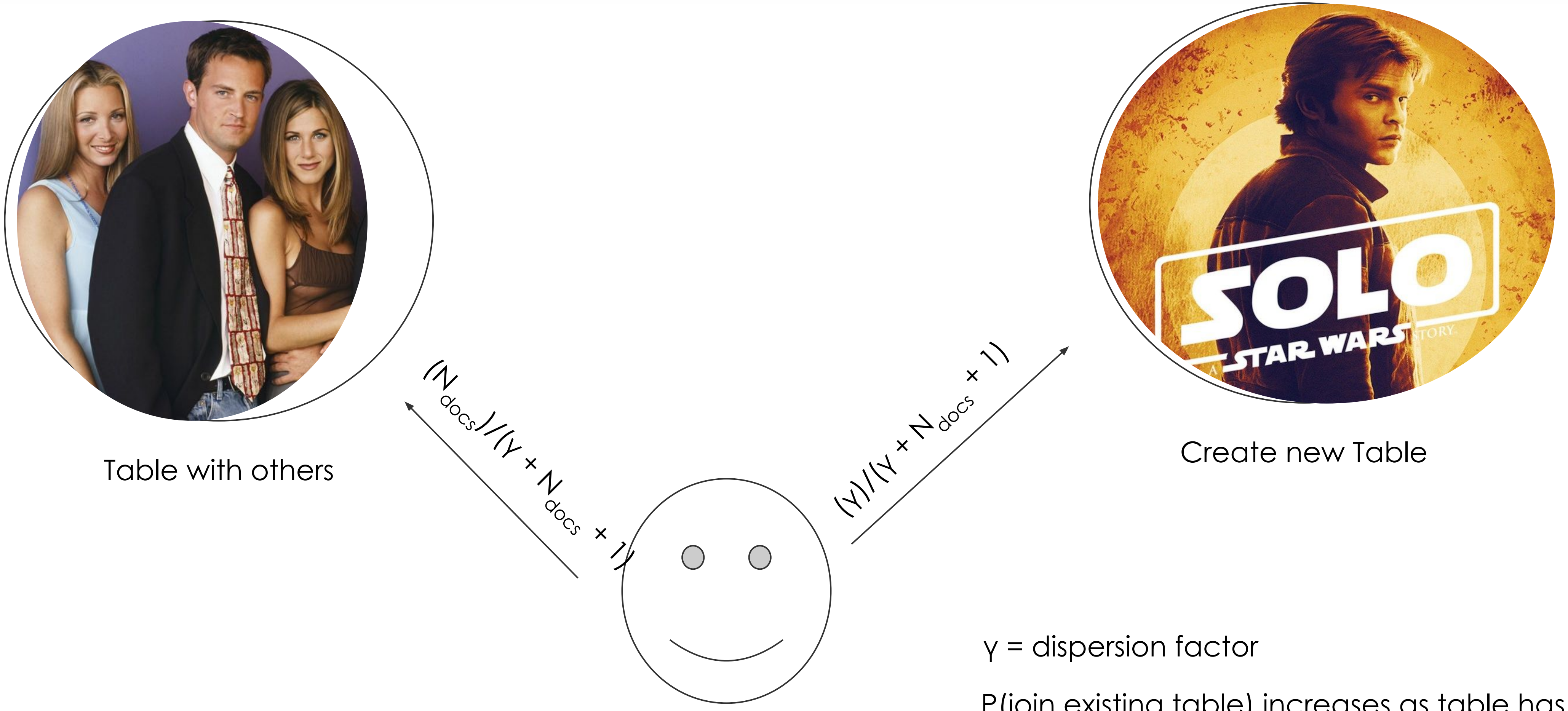


Table with others

$(N_{docs})/(\gamma + N_{docs} + 1)$

Create new Table

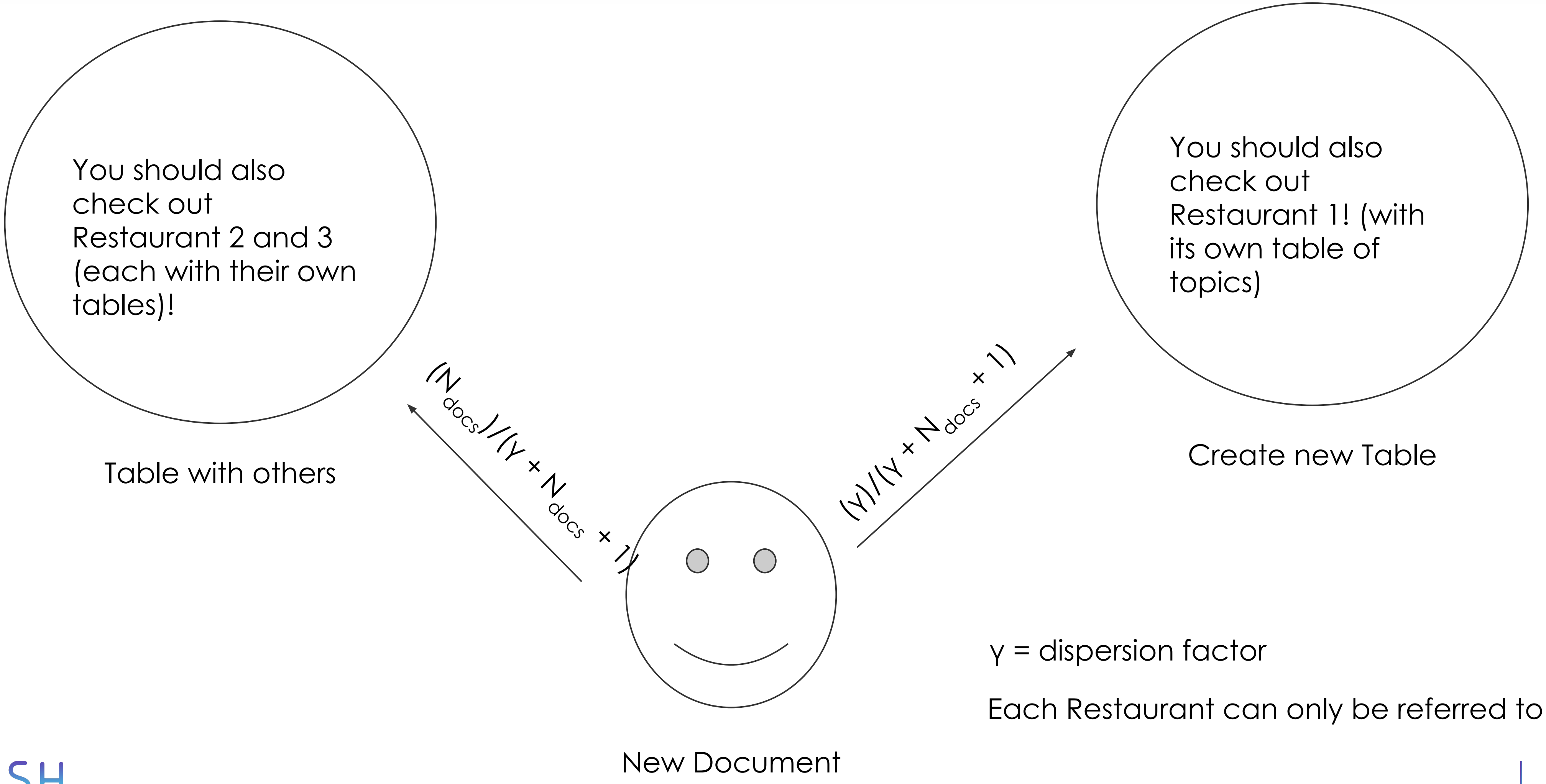
$(\gamma)/(\gamma + N_{docs} + 1)$

New Document

γ = dispersion factor

P(join existing table) increases as table has more docs

Section 2: Extension to CRP



You should also check out Restaurant 2 and 3 (each with their own tables)!

Table with others

$(N_{docs})/(\gamma + N_{docs} + 1)$

You should also check out Restaurant 1! (with its own table of topics)

Create new Table

$(\gamma)/(\gamma + N_{docs} + 1)$

New Document

γ = dispersion factor

Each Restaurant can only be referred to once

Section 3: A hierarchical topic model

Assumption:

Words in a document are generated according to a mixture model
mix proportion are document specific

Word distribution for given topic:

$$p(w \mid z, \beta)$$

w: words
z: topics. Multinomial distributed
β: parameter

Document specific mixture distribution:

$$p(w \mid \theta) = \sum_{i=1}^{K} \theta_i p(w \mid z = i, \beta_i)$$

θ: proportion of each topic in a document
drawn from $p(\theta \mid \alpha)$ (Dirichlet) based on corpus-level parameter α

Procedure:

1. choose θ
2. repeatedly sample words from $p(w \mid \theta)$

Section 3: A hierarchical topic model

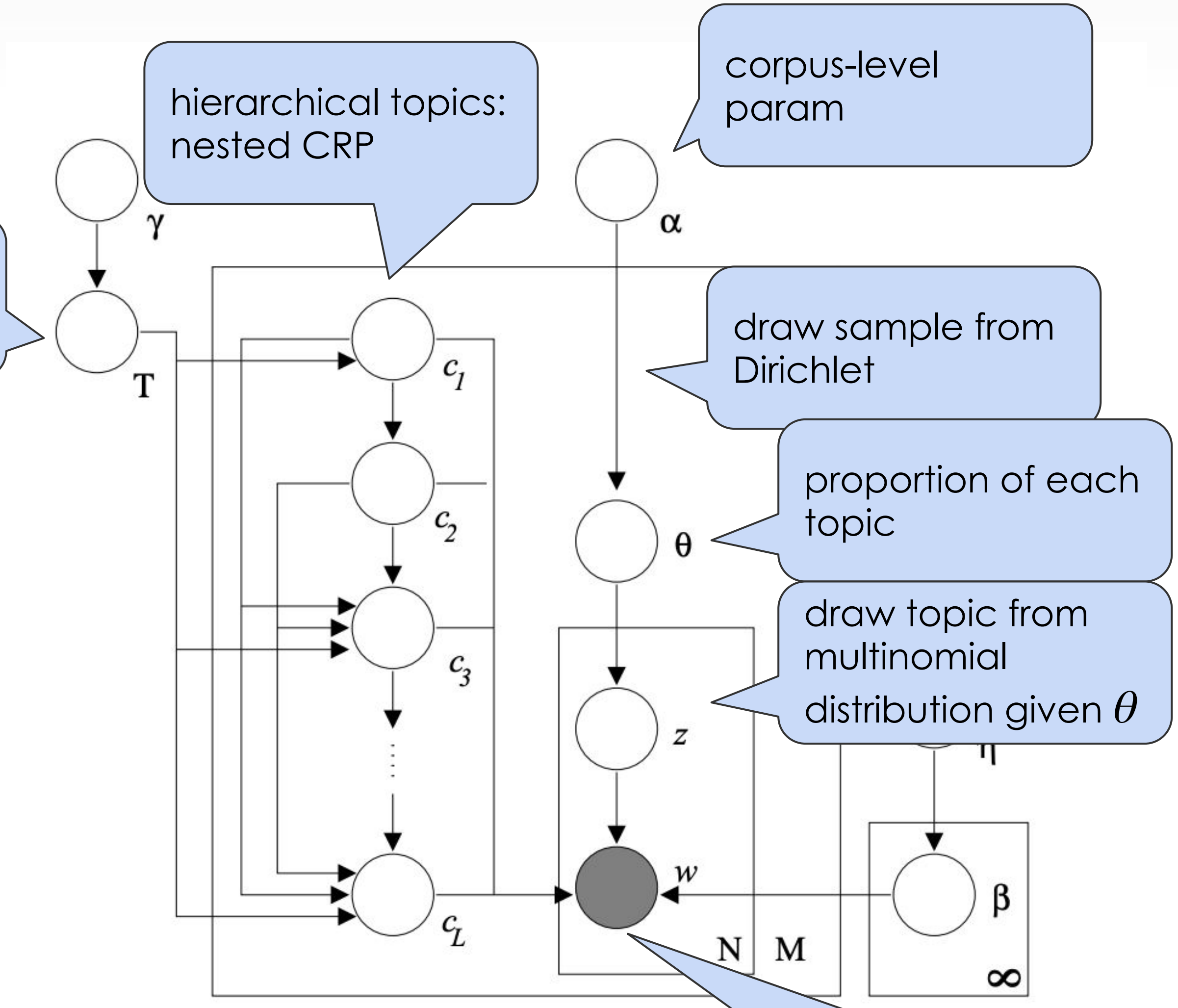
Extension I:

L-level tree with each node associated with a topic

Steps:

1. choose a path from root to leaf
2. draw θ from L-dimensional Dirichlet
3. generate topics from multinomial(θ)
4. generate words from a mixture of topic along the path of tree
5. place a prior on β

Section 3: A hierarchical topic model



(b)

infinite number of L-dimensional trees

hierarchical topics: nested CRP

corpus-level param

draw sample from Dirichlet

proportion of each topic

draw topic from multinomial distribution given θ

draw words from topic associated with c

Section 4: Approximate inference by Gibbs sampling

Background & Motivation for this:

- Goal of Bayesian inference is to maintain a full posterior probability distribution over a set of random variables. Maintaining and using this distribution often involves computing integrals
- The idea in Gibbs sampling is to generate posterior samples by sweeping through each variable (or block of variables) to sample from its conditional distribution with the remaining variables fixed to their current values.
- Works bc the conditional distribution of one variable given all others is proportional to the joint distribution

Section 4: Approximate inference by Gibbs sampling

Nitty Gritties:

- parameters of their model & definitions:
    - $w_{m,n}$: the *n*th word in the *m*th document (the only observed variables in the model)
    - $c_{m,l}$: the restaurant corresponding to the *l*`th topic in document *m* (that's a lower case L)
    - $z_{m,n}$: the assignment of the *n*th word in the *m*th document to one of the *L* available topics

- Gibbs sampling is used to sample $c_{m,l}$ and $z_{m,n}$

Section 4: Approximate inference by Gibbs sampling

Process

1) Sample $z_{m,n}$ using LDA approach
2) Use samples of $z_{m,n}$ to sample $c_{m,l}$

We can express **C_m**, the **L** topics associated with document **m**, as a conditional distribution and integrate it to obtain the likelihood

$$p(\mathbf{c}_m \,|\, \mathbf{w}, \mathbf{c}_{-m}, \mathbf{z}) \propto p(\mathbf{w}_m \,|\, \mathbf{c}, \mathbf{w}_{-m}, \mathbf{z}) p(\mathbf{c}_m \,|\, \mathbf{c}_{-m})$$

$$p(\mathbf{w}_m \,|\, \mathbf{c}, \mathbf{w}_{-m}, \mathbf{z}) = \prod_{\ell=1}^{L} \left(\frac{\Gamma(n^{(\cdot)}_{c_{m,\ell},-m} + W\eta)}{\prod_w \Gamma(n^{(w)}_{c_{m,\ell},-m} + \eta)} \frac{\prod_w \Gamma(n^{(w)}_{c_{m,\ell},-m} + n^{(w)}_{c_{m,\ell},m} + \eta)}{\Gamma(n^{(\cdot)}_{c_{m,\ell},-m} + n^{(\cdot)}_{c_{m,\ell},m} + W\eta)} \right)$$

Section 5: Examples and empirical results (1 of 3)

Authors conduct a series of experiments to validate hLDA as an approach

1. demonstrate hLDA as feasible for learning text hierarchies
2. compare CRP method on LDA models to “standard” approach
3. test hLDA on real data: 1717 NIPS abstracts from 1987-1999

Section 5: Examples and empirical results (2 of 3)

- Tests “a” and “b” involve generating synthetic data from distributions with pre-specified hierarchical and probabilistic structures.
  - They demonstrate that hLDA is able to recover the hierarchical structure in the simulated data
  - They find that hLDA was more effective than traditional LDA in finding the “correct” model for the data
- Test “c” involved running hLDA on NIPS abstracts from 1987-1999
  - hLDA captures “function” words without the need for an auxiliary list
  - separates abstracts into “neuroscience” and “ML” abstracts and identifies coherent subtopics within those fields

Section 5: Examples and empirical results (3 of 3)

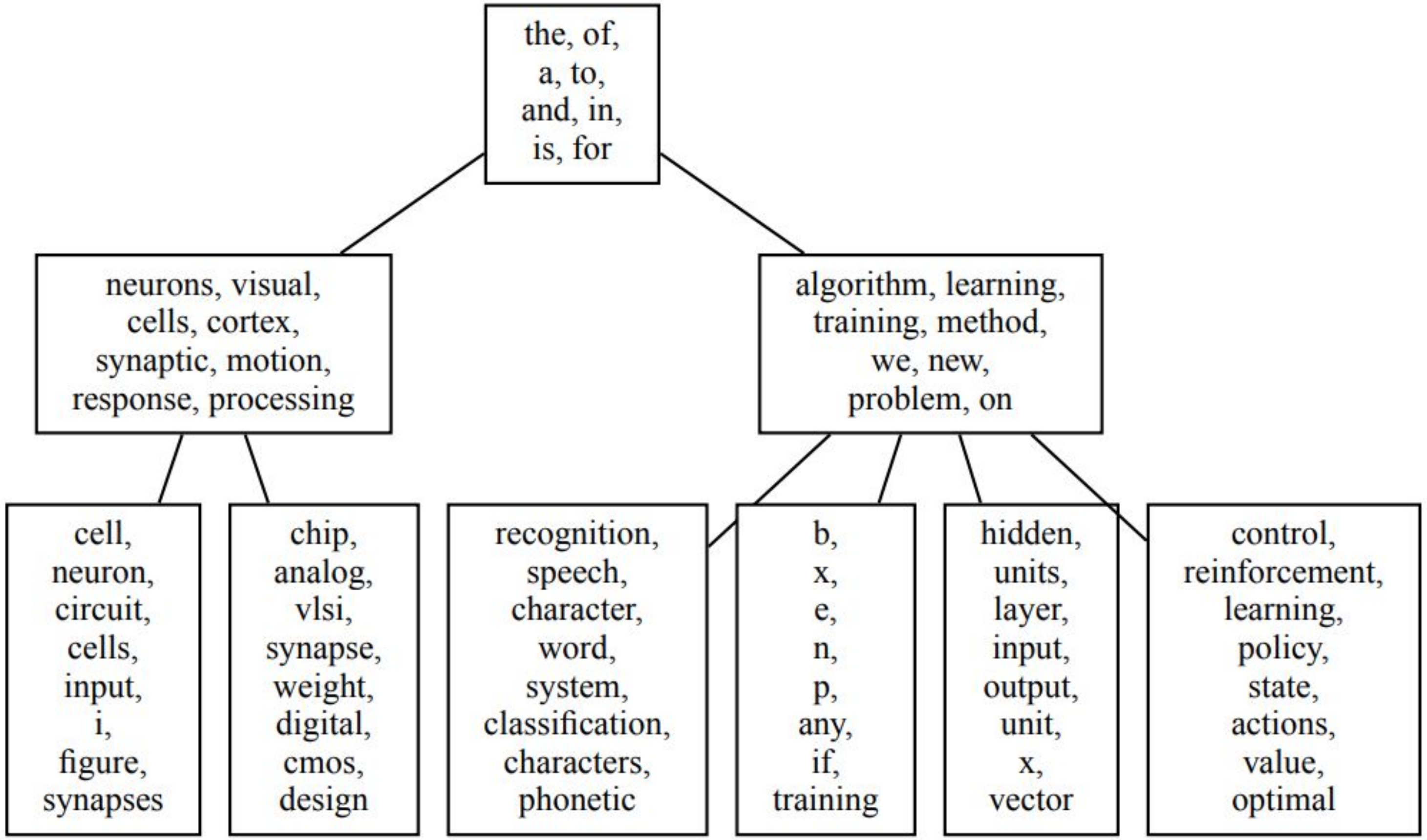

the, of, a, to, and, in, is, for
neurons, visual, cells, cortex, synaptic, motion, response, processing
algorithm, learning, training, method, we, new, problem, on
cell, neuron, circuit, cells, input, i, figure, synapses
chip, analog, vlsi, synapse, weight, digital, cmos, design
recognition, speech, character, word, system, classification, characters, phonetic
b, x, e, n, p, any, if, training
hidden, units, layer, input, output, unit, x, vector
control, reinforcement, learning, policy, state, actions, value, optimal

Figure 5: A topic hierarchy estimated from 1717 abstracts from NIPS01 through NIPS12. Each node contains the top eight words from its corresponding topic distribution.

Section 6: Summary

- hLDA represents a flexible approach to topic models that can accommodate growing data collections
- Extensions:
    - generalize to allow for variable tree depth
    - generalize to allow documents to mix over various paths