Low-Rank Factorizations for Fast Inference in Structured Models

Justin Chiu* [1]   Yuntian Deng* [2]   Alexander Rush [1]

[1]Cornell Tech

[2]Harvard University

October 19, 2021

Structured Models

▶ Explicitly model output associations
  ▶ Directly or through latent variables

▶ Focus on combinatorially large latent <u>discrete structures</u>
  ▶ Complementary to continuous, deterministic representations

▶ More difficult to scale than alternative representations
 ▶ Bottlenecked by time and space complexity of inference

Scaling Structured Models

▶ Scaling (to the point of overparameterization) is key

▶ Target tractable models
  ▶ Admit dynamic programs for exact inference

▶ Impose a low-rank model constraint
  ▶ Trades off model expressivity for cheaper inference
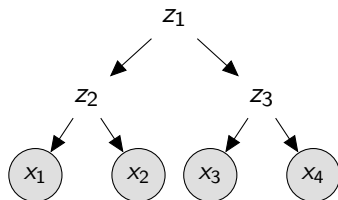
▶ Only constrain parameters used in key steps of inference

Inference in Structured Models

▶ Model an observation $x = (x_1, \ldots, x_T)$ via latent structure z
  ▶ Latent nodes z_i
  ▶ Nodes have discrete label set $[L]$

▶ Perform training and evaluation via marginalization

$$p(x) = \sum_z p(x, z)$$
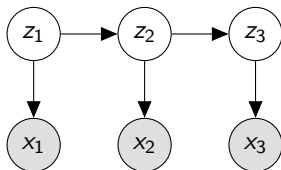
Hidden Markov models

Probabilistic context-free grammars

Hypergraphs for Inference

▶ Represent dynamic programs for inference as hypergraphs

▶ Hypergraphs consist of nodes and hyperedges
  ▶ Hyperedge consists of a head node and set of tail nodes

▶ Perform inference by traversing hypergraph
  ▶ Aggregate marginals from tails to head via a matrix-vector product
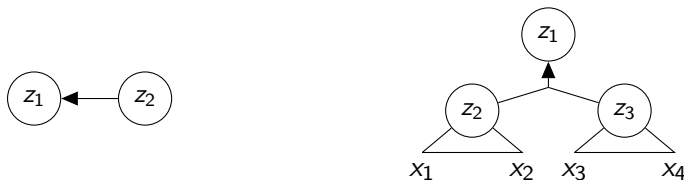
Hyperedge representations for HMMs and PCFGs

Hypergraph Marginalization

For each hyperedge e in topological order,

▶ Combine tail marginals α_1, α_2 into joint tail marginal β_v

▶ Apply score matrix Ψ_e and aggregate in head marginal α_u
  ▶ Multiple hyperedges may have the same head node

Algorithm 1 Hypergraph marginalization

$\quad$ **for** $u \leftarrow v$ hyperedge e topologically **do**
$\qquad \beta_v \leftarrow \alpha_{v_1} \alpha_{v_2}^\top$ $\hfill \triangleright O(L^{|e|})$
$\qquad \alpha_u \overset{+}{\leftarrow} \Psi_e \beta_v$ $\hfill \triangleright O(L^{|e|+1})$
$\quad$ **return** $\alpha_S^\top \mathbf{1}$

Scaling with Low-rank Factorizations

▶ Factor matrices $\Psi = UV^\top$, $U \in \mathbb{R}^{L \times R}$, $V \in \mathbb{R}^{L^{|e|} \times R}$

$$\boxed{\Psi} \times \boxed{\beta} = \boxed{U} \times \left(\boxed{V^\top} \times \boxed{\beta} \right)$$

▶ Two matrix-vector products of cost $O(LR)$ and $O(L^{|e|}R)$
  ▶ Reduced from $O(L^{|e|+1})$

▶ Potentially large speedups for inference
  ▶ HMM from $O(L^2)$ to $O(LR)$
  ▶ PCFG from $O(L^3)$ to $O(L^2R)$

Low-rank Hypergraph Marginalization

Algorithm 2 Low-rank marginalization

---

  for $u \leftarrow v_1, v_2$ hyperedge e topologically **do**

    $\beta_v \leftarrow \alpha_{v_1} \alpha_{v_2}^{\top}$ $\triangleright$ $O(L^{|e|})$

    $\gamma \leftarrow V_e^{\top} \beta_v$ $\triangleright$ $O(L^{|e|}R)$

    $\alpha_u \overset{+}{\leftarrow} U_e \gamma$ $\triangleright$ $O(LR)$

  return $\alpha_S^{\top} \mathbf{1}$

Expressivity of Rank-constrained Models

▶ Rank constraints limit expressivity

▶ Only apply to a subset of parameters
    ▶ Transition matrix for HMMs
    ▶ Subset of the transition matrix for PCFGs

▶ Is it more expressive than a smaller model?
   ▶ An L-state HMM with rank R ($< L$) is more expressive than an unconstrained R-state HMM

Experiments

▶ Language modeling on PENN TREEBANK[1]

▶ Compare size vs speed and accuracy
  ▶ Size = 1k to 16k state HMM, 90 to 300 state PCFG
  ▶ Speed = Sec/Batch
  ▶ Accuracy = Perplexity (function of likelihood)

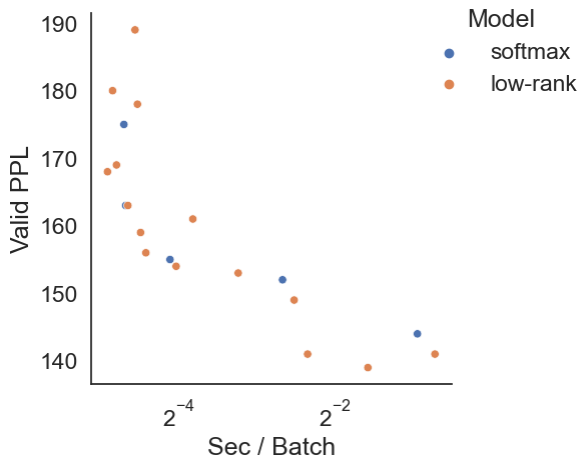▶ Unconstrained softmax HMM, PCFG vs low-rank versions

---

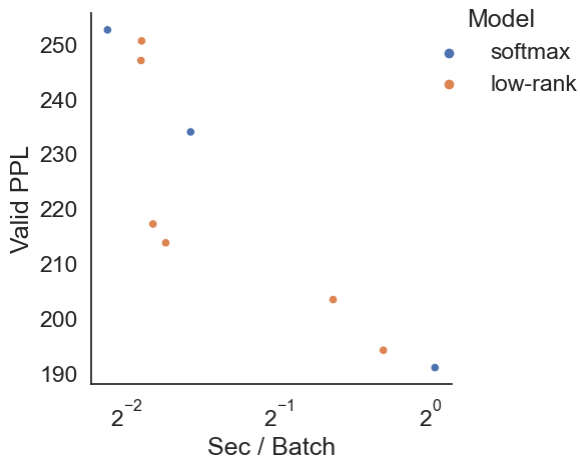[1]Marcus, Santorini, and Marcinkiewicz, 'Building a Large Annotated Corpus of English: The Penn Treebank'.

HMM Results
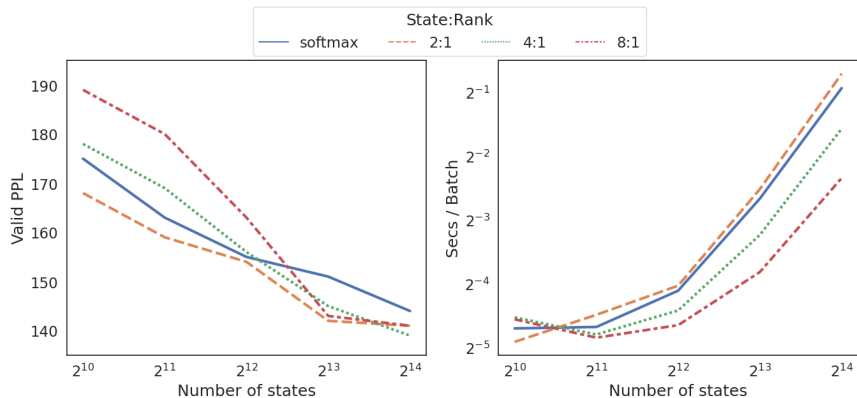
HMM Speed vs Accuracy Frontier

PCFG Speed vs Accuracy Frontier

Conclusion

▶ Introduce a low-rank factorization to speed up inference in hypergraph models

▶ Constrain only parameters used in inference bottlenecks

▶ Most effective with large models

▶ Please see the paper for more experiments and analysis!

Citations I

Marcus, Mitchell P., Beatrice Santorini, and Mary Ann Marcinkiewicz. 'Building a Large Annotated Corpus of English: The Penn Treebank'. In: *Computational Linguistics* 19.2 (1993), pp. 313–330. URL: https://www.aclweb.org/anthology/J93-2004.