Rational Recurrences for Empirical Natural Language Processing

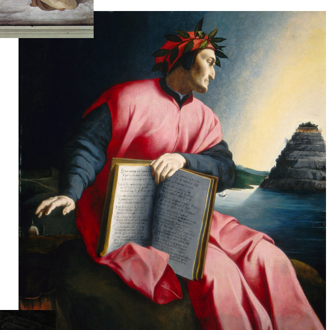
Noah Smith

University of Washington & Allen Institute for Artificial Intelligence

nasmith@cs.washington.edu noah@allenai.org @nlpnoah

A Bit of History

Interpretability? Guarantees?



Rule-based NLP (1980s and before)

- E.g., lexicons and regular expression pattern matching
- Information extraction

Statistical NLP (1990s-2000s)

- Probabilistic models over features derived from rule-based NLP
- Sentiment/opinion analysis, machine translation

Neural NLP (2010s)

- Vectors, matrices, tensors, and lots of nonlinearities

Outline

1. An interpretable neural network inspired by rule-based NLP: SoPa
   "Bridging CNNs, RNNs, and weighted finite-state machines," Schwartz et al., ACL 2018
2. A restricted class of RNNs that includes SoPa: rational recurrences
   "Rational recurrences," Peng et al., EMNLP 2018
3. More compact rational RNNs using sparse regularization
   work under review
4. A few parting shots

Patterns

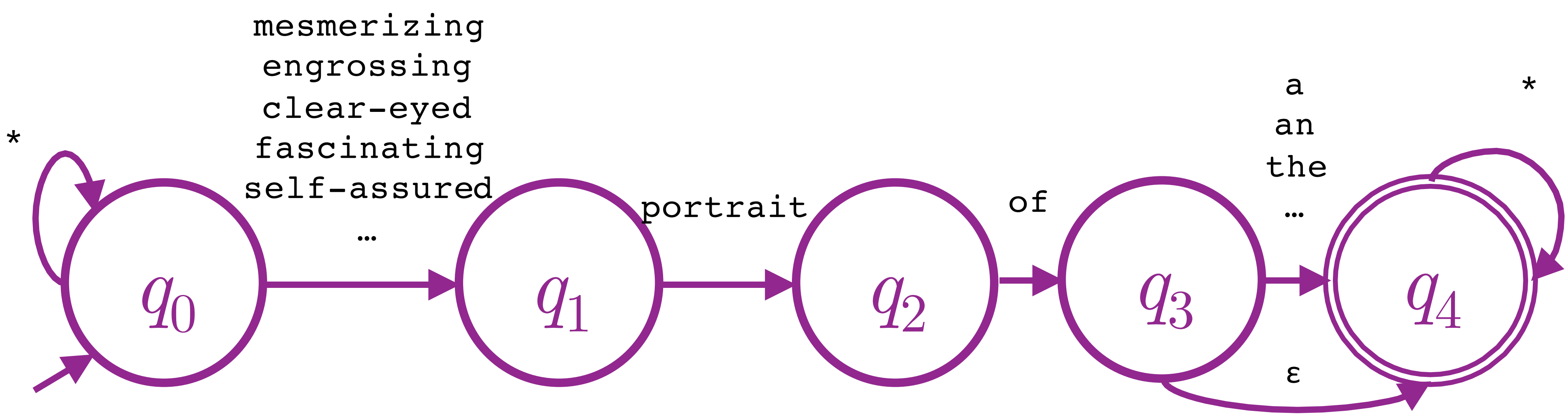
- Lexical semantics
  (Hearst, 1992; Lin et al., 2003; Snow et al., 2006; Turney, 2008; Schwartz et al., 2015)
- Information extraction
  (Etzioni et al., 2005)
- Document classification
  (Tsur et al., 2010; Davidov et al., 2010; Schwartz et al., 2013)
- Text generation
  (Araki et al., 2016)

good fun, good action, good acting, good dialogue, good pace, good cinematography.

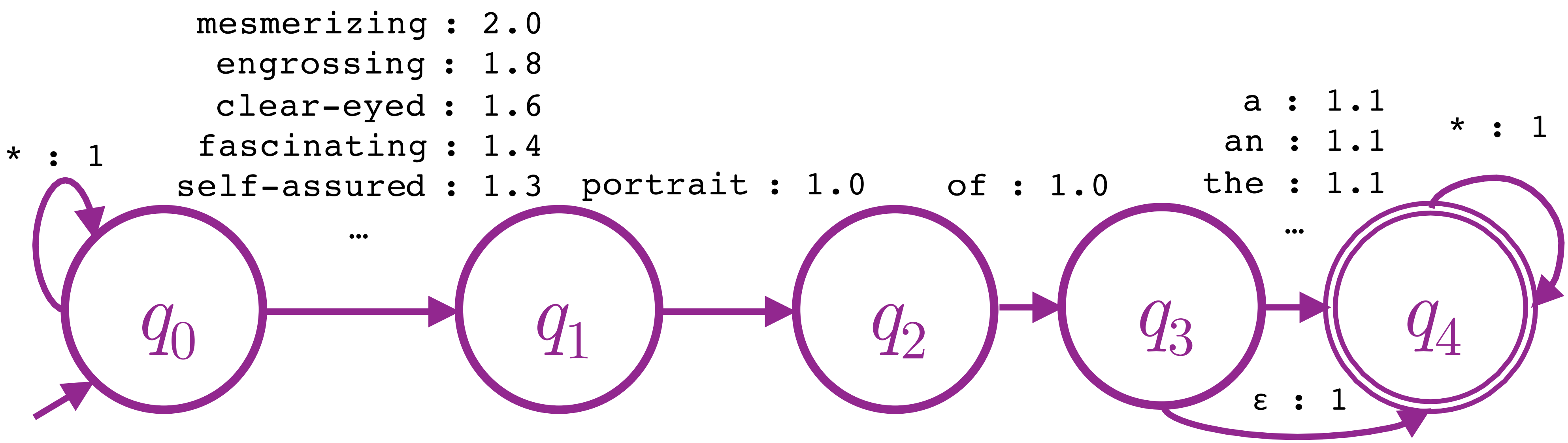
flat, misguided comedy.

long before it 's over, you'll be thinking of 51 ways to leave this loser.

Patterns from Lexicons and Regular Expressions



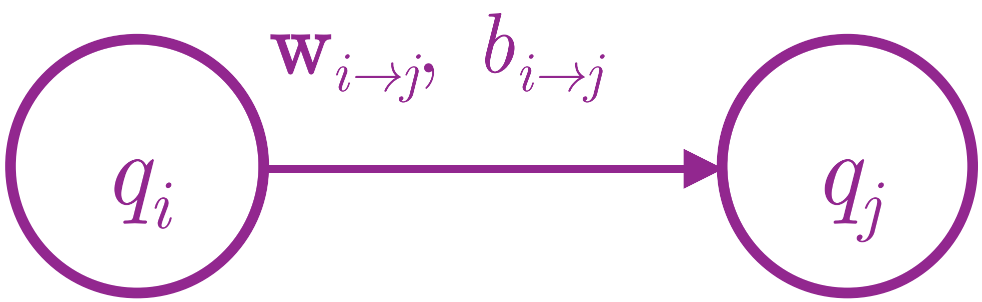
Weighted Patterns



a mesmerizing portrait of an engineer : 1 × 2.0 × 1 × 1 × 1.1 × 1 = 2.2
the most fascinating portrait of students : 1 × 1 × 1.4 × 1 × 1 × 1.1 × 1 = 1.5
a clear-eyed picture of the modern : 0
flat , misguided comedy : 0

Soft Patterns (SoPa)

Score word **vectors** instead of a separate weight for each word

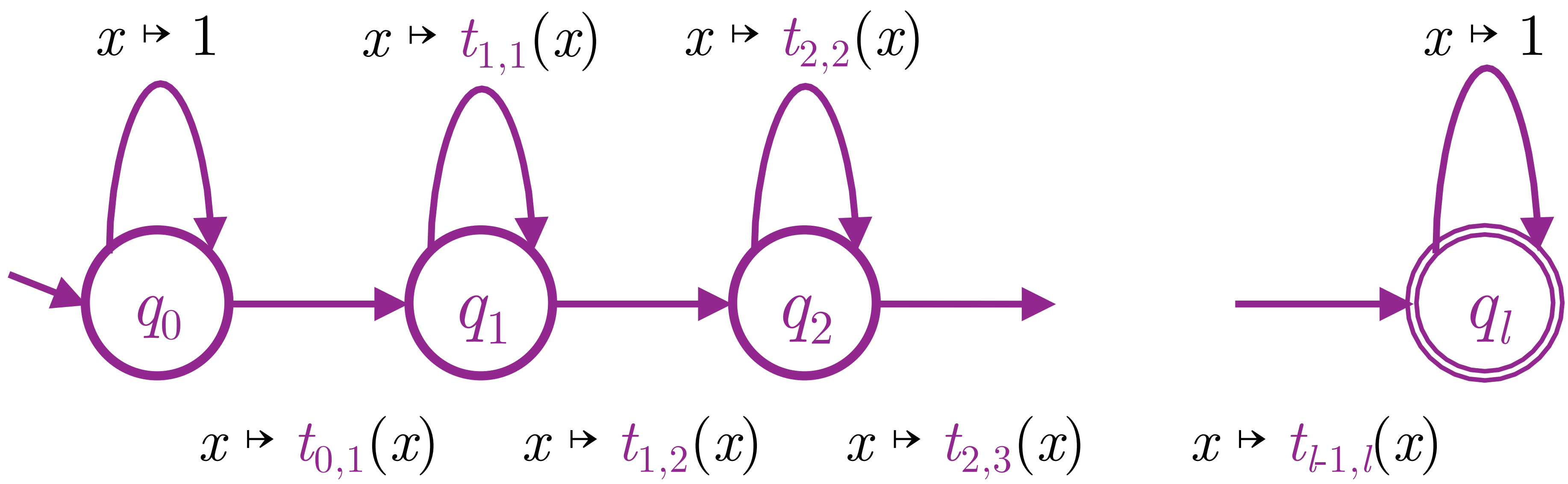


your favorite embedding for word x goes here

$$t_{i,j}(x) = \sigma(\mathbf{w}_{i \to j} \cdot \mathbf{v}_x + b_{i \to j})$$

Soft Patterns (SoPa)

Flexible-length patterns: $l + 1$ states with self-loops



Soft Patterns (SoPa)

Transition matrix has $O(l)$ parameters

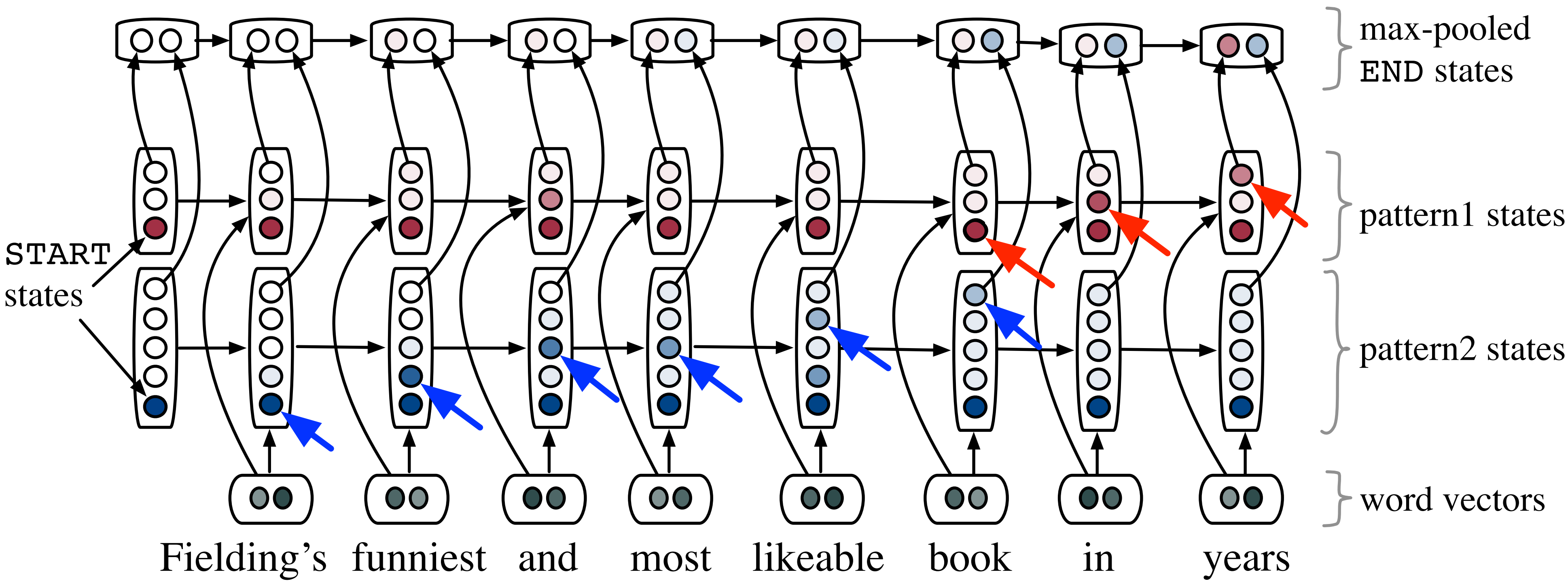
$$\mathbf{T}(x) = \begin{bmatrix} 1 & t_{0,1}(x) & 0 & 0 & 0 & 0 \\ 0 & t_{1,1}(x) & t_{1,2}(x) & 0 & 0 & 0 \\ 0 & 0 & t_{2,2}(x) & t_{2,3}(x) & 0 & 0 \\ 0 & 0 & 0 & t_{3,3}(x) & \ddots & 0 \\ 0 & 0 & 0 & 0 & \ddots & t_{l\text{-}1,l}(x) \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

SoPa Sequence-Scoring: Matrix Multiplication

$$matchScore(\text{“}\texttt{flat , misguided comedy .}\text{”}) =$$

$$\mathbf{w}_{start}{}^{\top}\, \mathbf{T}(\texttt{flat})\, \mathbf{T}(\texttt{,})\, \mathbf{T}(\texttt{misguided})\, \mathbf{T}(\texttt{comedy})\, \mathbf{T}(\texttt{.})\, \mathbf{w}_{end}$$

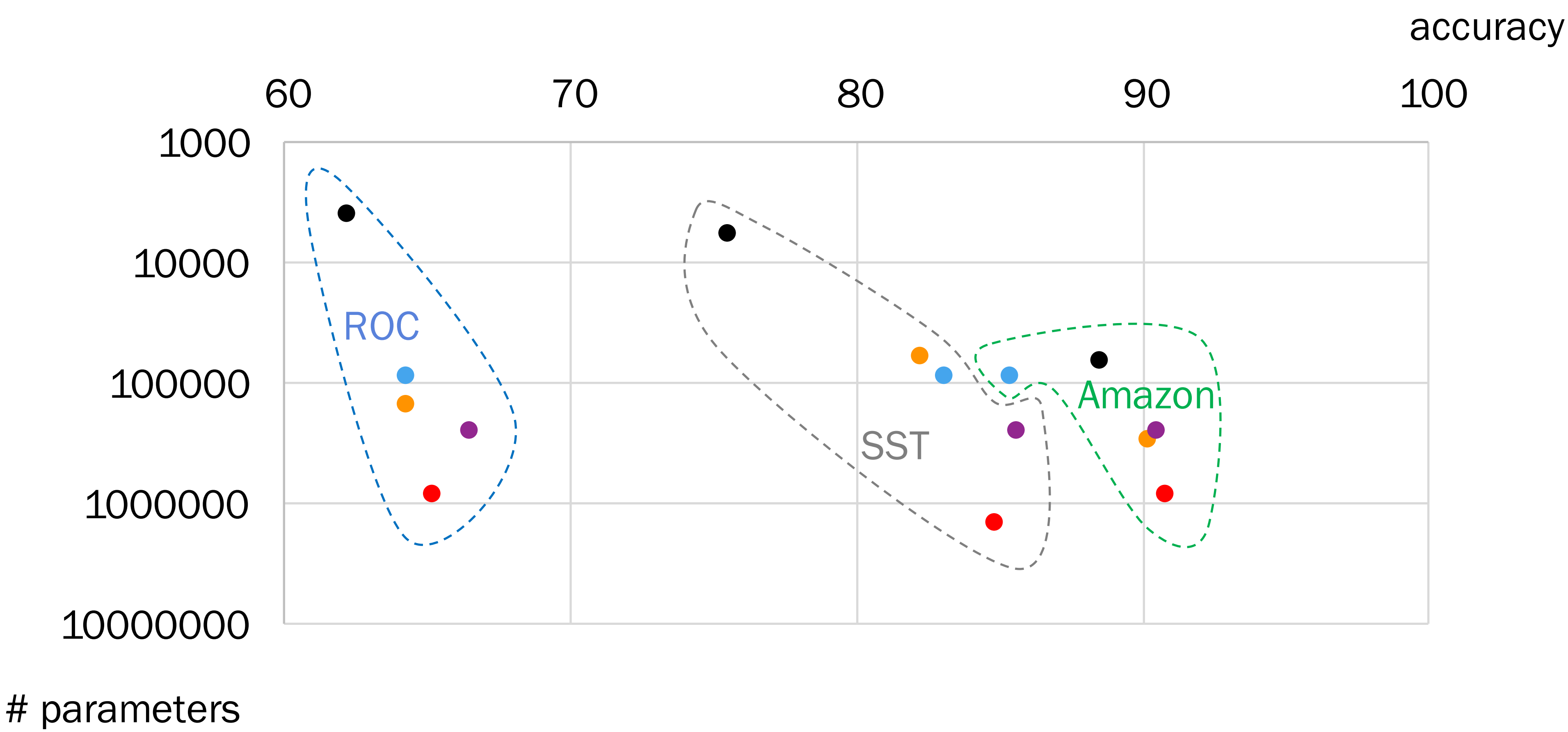
Two-SoPa Recurrent Neural Network



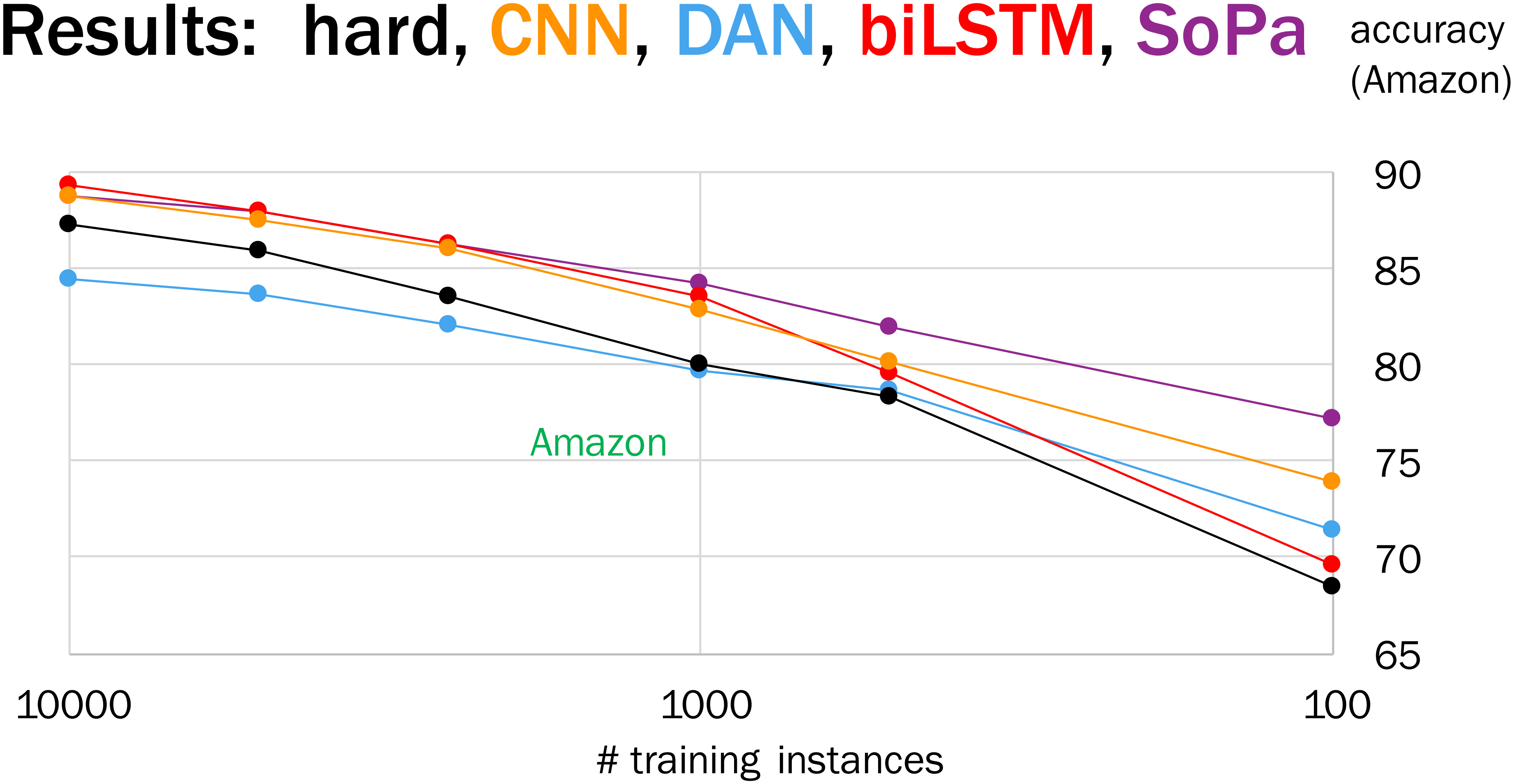
Experiments

- 200 SoPas, each with 2–6 states
- Text input is fed to all 200 patterns in parallel
- Pattern match scores fed to an MLP, with end-to-end training
- *Datasets*:
  - Amazon electronic product reviews (20K), binarized (McAuley &Leskovec, 2013)
  - Stanford sentiment treebank (7K): movie review sentences, binarized (Socher et al., 2013)
  - ROCStories (3K): story cloze, only right/wrong ending, no story prefix (i.e., style) (Mostafazadeh et al., 2016)
- *Baselines*:
  - LR with hard patterns (Davidov & Rappaport, 2008; Tsur et al., 2010)
  - one-layer CNN with max-pooling (Kim, 2014)
  - deep averaging network (Iyyer et al., 2015)
  - one-layer biLSTM (Zhou et al., 2016)
- *Hyperparameters* tuned for all models by **random search**; see the paper's appendix

Results: hard, CNN, DAN, biLSTM, SoPa



Results: hard, CNN, DAN, biLSTM, SoPa



Notes

- We also include ε-transitions.
- We can replace addition operations with max, so that the recurrence equates to the **Viterbi** algorithm for WFSAs.
- Without self-loops, ε-transitions, and the sigmoid, SoPa becomes a convolutional neural network (LeCun, 1998).

Lots more experiments and details in the paper!

Interpretability (Negative Patterns)

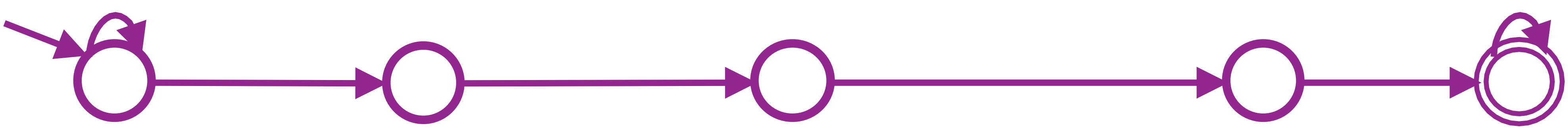
- it's dumb, but more importantly, it's just not scary
- though moonlight mile is replete with acclaimed actors and actresses and tackles a subject that's potentially moving , the movie is too predictable and too self-conscious to reach a level of high drama
- While its careful pace and seemingly opaque story may not satisfy every moviegoer's appetite, the film 's final scene is soaringly, transparently moving
- the band's courage in the face of official repression is inspiring, especially for aging hippies (this one included).

Interpretability (Positive Patterns)

- it's dumb, but more importantly, it's just not scary
- though moonlight mile is replete with acclaimed actors and actresses and tackles a subject that's potentially moving , the movie is too predictable and too self-conscious to reach a level of high drama
- While its careful pace and seemingly opaque story may not satisfy every moviegoer's appetite, the film 's final scene is soaringly, transparently moving
- the band's courage in the face of official repression is inspiring, especially for aging hippies (this one included).

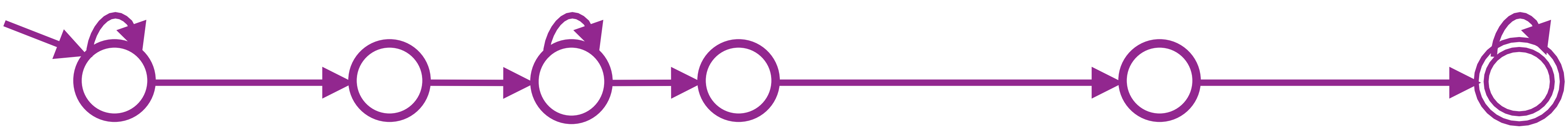
Interpretability (One SoPa)

mesmerizing	portrait	of	a
engrossing	portrait	of	a
clear-eyed	portrait	of	an
fascinating	portrait	of	a
self-assured	portrait	of	small



Interpretability (One SoPa)

honest	,		and	enjoyable
soulful	,	scathing$_{SL}$	and	joyous
unpretentious	,	charming$_{SL}$	,	quirky
forceful	,		and	beautifully
energetic	,		and	surprisingly



Interpretability (One SoPa)

is	deadly	dull
a	numbingly	dull
is	remarkably	dull
is	a	phlegmatic
an	utterly	incompetent

Summary So Far

- SoPa: an RNN that
  - equates to WFSAs that score sequences of word vectors
  - calculates those scores in parallel
  - works well for text classification tasks
- RNNs don't have to be inscrutable and disrespectful of theory.

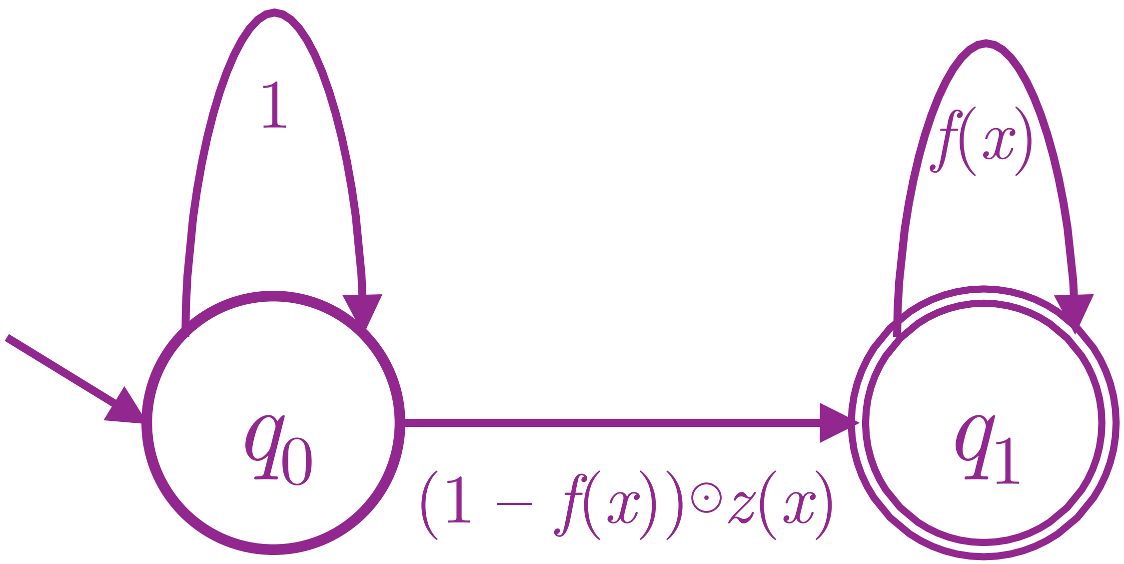
https://github.com/Noahs-ARK/soft_patterns

Rational Recurrences

A recurrent network is rational if its hidden state can be calculated by an array of weighted FSAs
over some semiring
whose operations take constant time and space.

*We are using standard terminology. "**Rational**" is to **weighted FSAs** as "regular" is to (unweighted) FSAs (e.g., "rational series," Sakarovitch, 2009; "rational kernels," Cortes et al., 2004).

Simple Recurrent Unit (Lei et al., 2017)

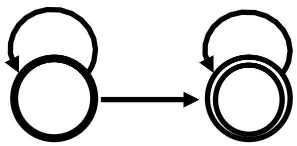


Some Rational Recurrences

- SoPa (Schwartz et al., 2018)
- Simple recurrent unit (Lei et al., 2017)
- Input switched affine network (Foerster et al., 2017)
- Structurally constrained (Mikolov et al., 2014)
- Strongly-typed (Balduzzi and Ghifary, 2016)
- Recurrent convolution (Lei et al., 2016)
- Quasi-recurrent (Bradbury et al., 2017)
- New models!

Rational Recurrences and Others

Rational recurrences



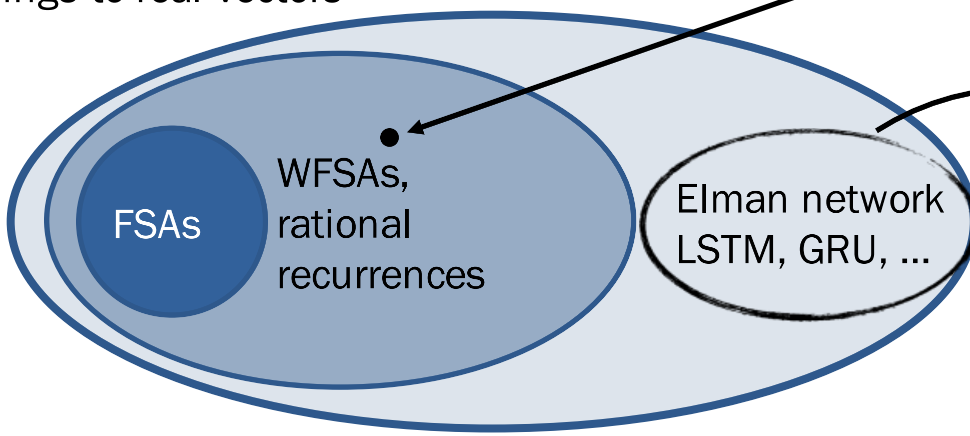
Elman-style networks

$$\mathbf{h}_t = \tanh\left(\mathbf{W}\mathbf{h}_{t-1} + \mathbf{U}\mathbf{x}_t\right)$$

and LSTMs, GRUs...

Functions mapping strings to real vectors

Convolutional neural nets
(Schwartz et al., 2018)

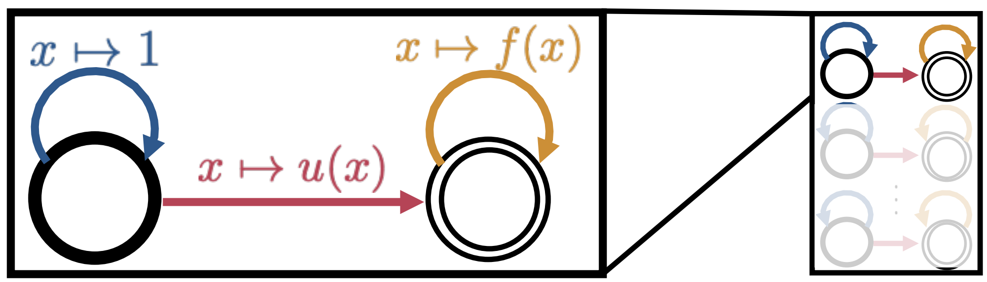


Conjecture

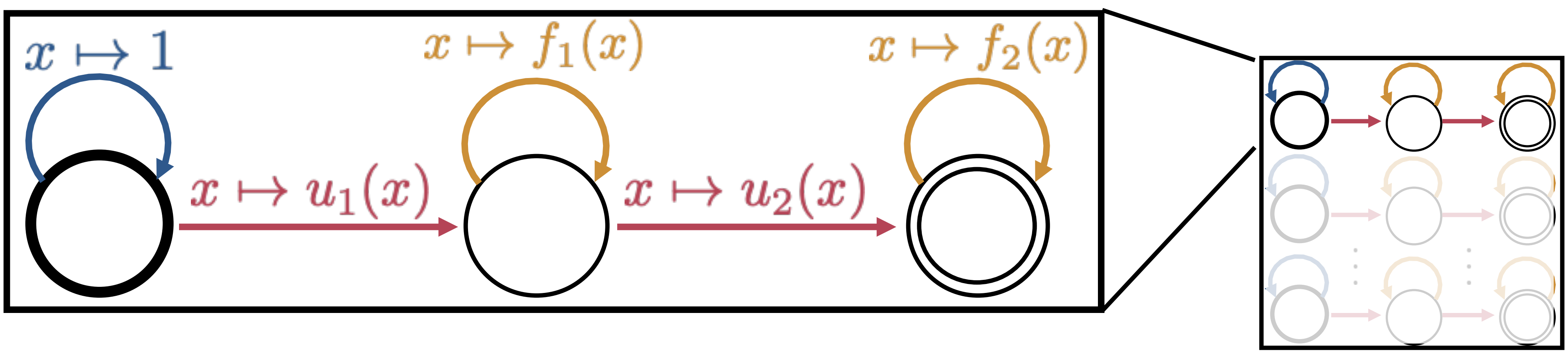
(this morning, Ariadna talked about the connection between WFSAs and *linear* Elman networks)

“Unigram” and “Bigram” Models

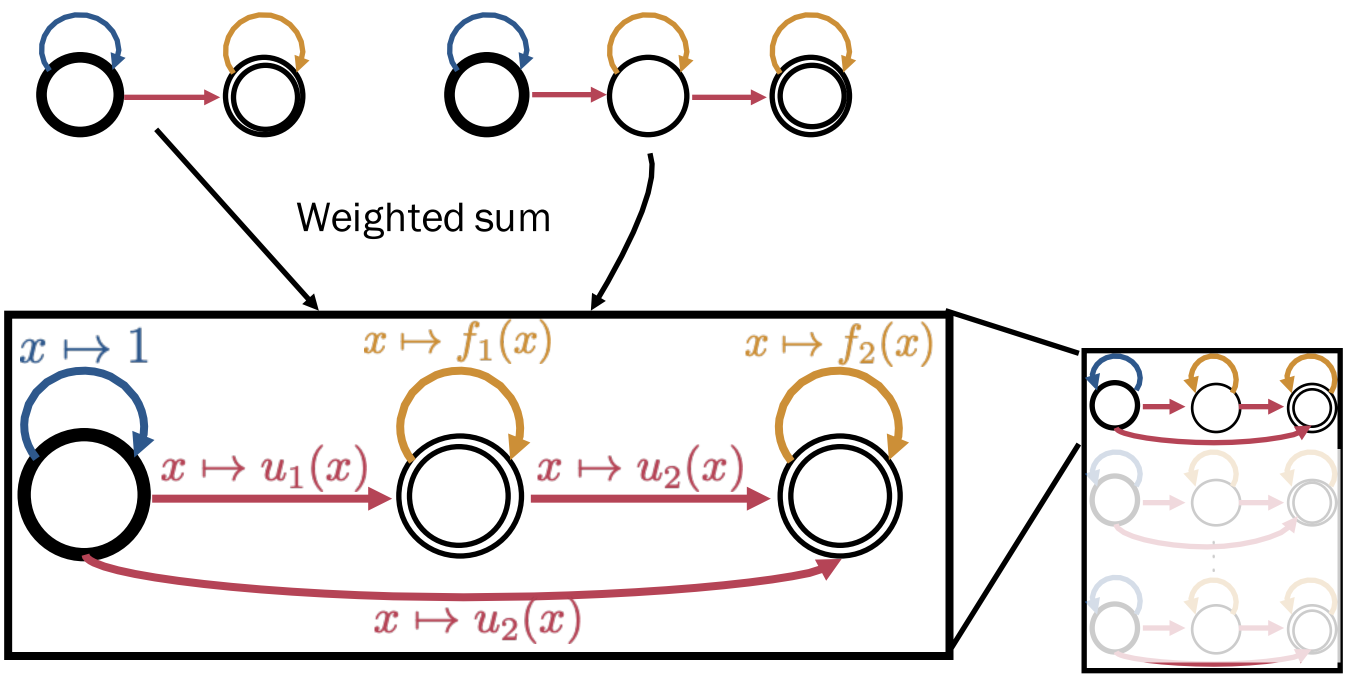
Unigram: At least one transition from the initial state to final.
(“Example 6” in the paper, close to SRU, T-RNN, and SCRN.)



Bigram: At least two transitions from the initial state to final.



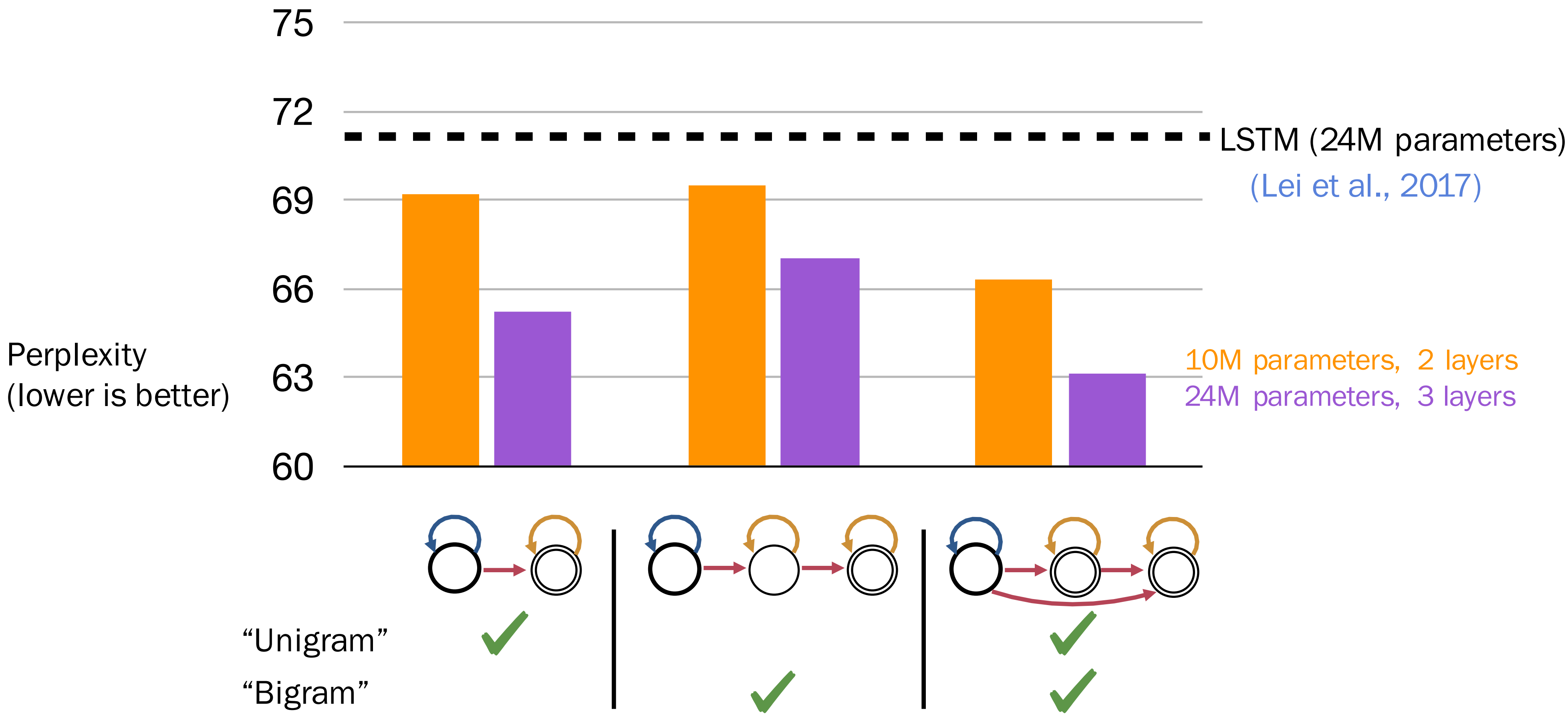
Interpolation



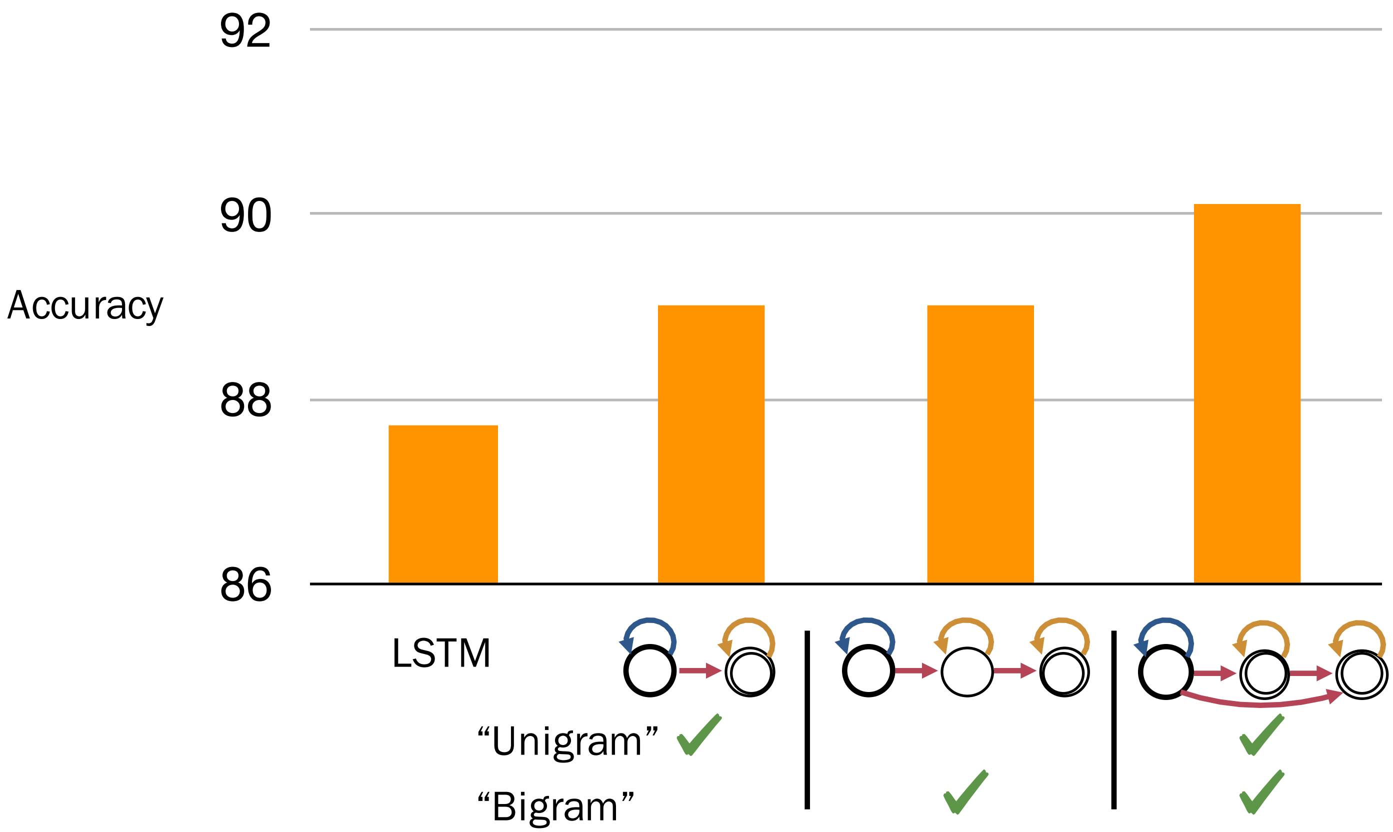
Experiments

- *Datasets*: PTB (language modeling);
  Amazon, SST, Subjectivity, Customer Reviews (text classification)
- *Baseline*:
  - LSTM reported by Lei et al. (2017)
- *Hyperparameters* follow Lei et al. for language modeling;
  tuned for text classification models by **random search**;
  see the paper's appendix

Results: Language Modeling (PTB)



Results: Text Classification
(Average of Amazon, SST, Subjectivity, Customer Reviews)



Summary So Far

- Many RNNs are arrays of WFSAs.
- Reduced capacity/expressive power can be beneficial.
- Theory is about *one-layer* RNNs; in practice 2+ layers work better.

https://github.com/Noahs-ARK/rational-recurrences

Increased Automation

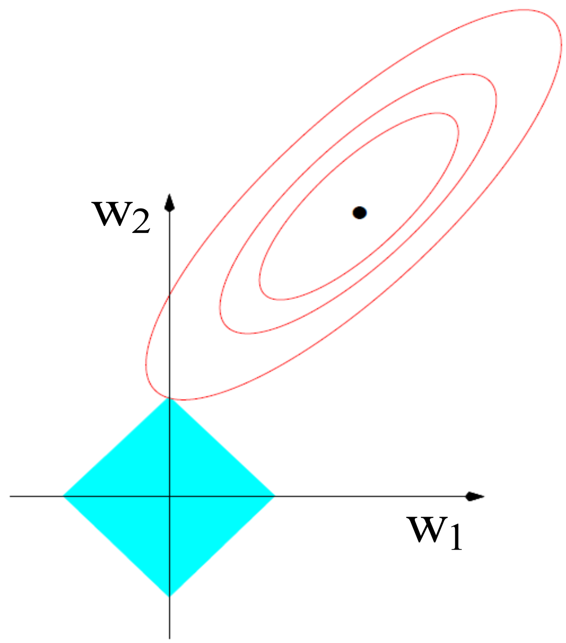
- Original SoPa experiments: “200 SoPas, each with 2–6 states”
- Can we *learn* how many states each pattern needs?
- Relatedly, can we learn smaller, more compact models?

Sparse regularization lets us do this *during* parameter learning!

Sparsity and Structured Sparsity

- In linear models, the lasso (Tibshirani, 1996) penalizes each weight/parameter vector by its L_1 norm.
  - Classic use in NLP: Kazama and Tsujii (EMNLP 2003)

$$\sum_i |w_i|$$



$$\widehat{\mathbf{w}} = \arg\min_{\mathbf{w}} \|\mathbf{A}\mathbf{w} - \mathbf{y}\|_2^2$$
$$\text{subject to } \|\mathbf{w}\|_1 \leq \tau$$

Sparsity and Structured Sparsity

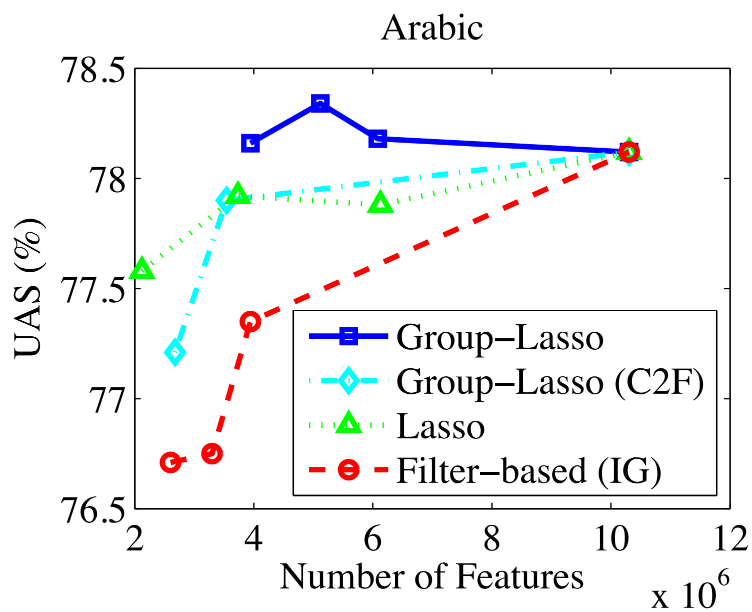
- In linear models, the lasso (Tibshirani, 1996) penalizes each weight/parameter vector by its L_1 norm.
  - Classic use in NLP: Kazama and Tsujii (EMNLP 2003)
- A generalization is the group lasso (Bakin, 1999; Yuan and Lin, 2006), which penalizes each group's L_2 norm.
  - If every parameter is in its own group, equivalent to lasso
  - If all parameters are in one group, equivalent to ridge

$$\sum_i |w_i|$$

$$\sum_g \lambda_g \|\boldsymbol{w}_g\|_2$$

subvector of parameters in group g

Benefit of Sparse Lasso

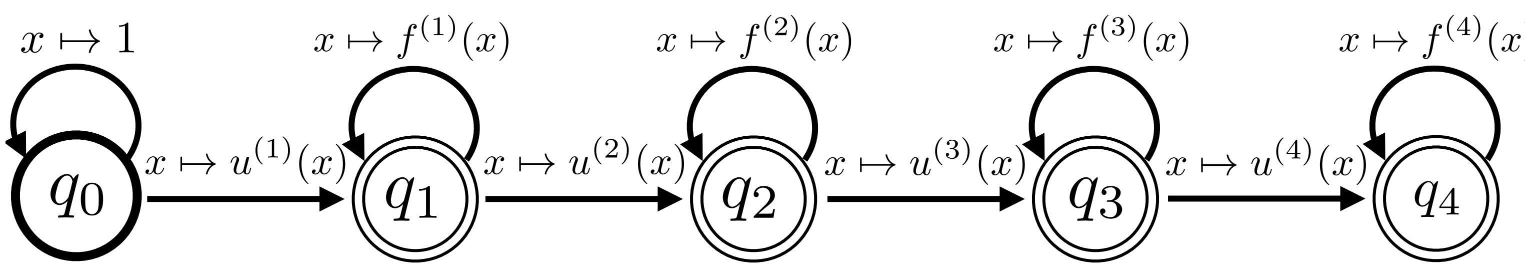


- With appropriate hyperparameter assignments, many groups are driven to zero.
- E.g., we grouped weights by feature template.
- Can this work for neural models?

Arabic dependency parsing: UAS vs. millions of features (Martins et al., EMNLP 2011)

Procedure

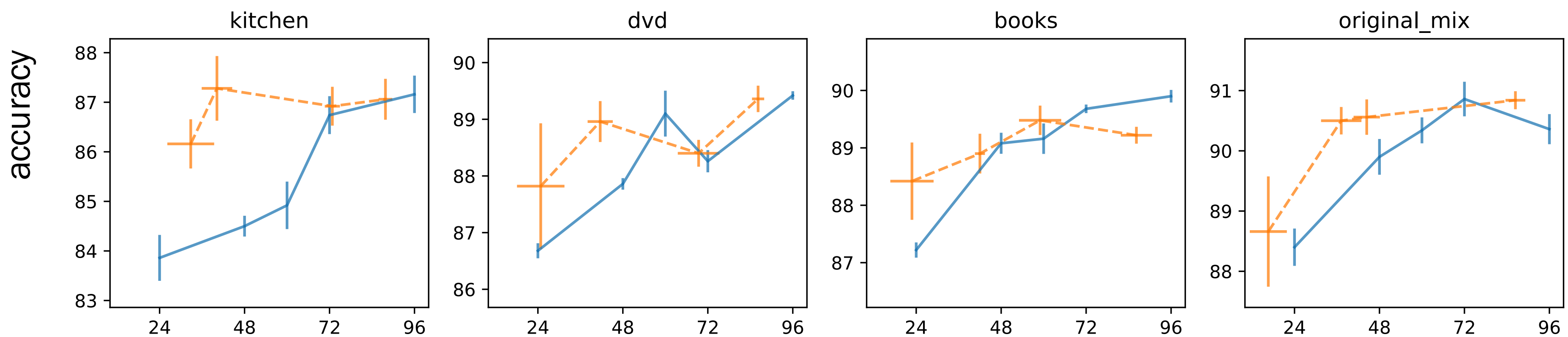
1. Train the model with group lasso, one group per state.
2. Eliminate states whose weights are close to zero.
3. Finetune the remaining model by minimizing unregularized loss.

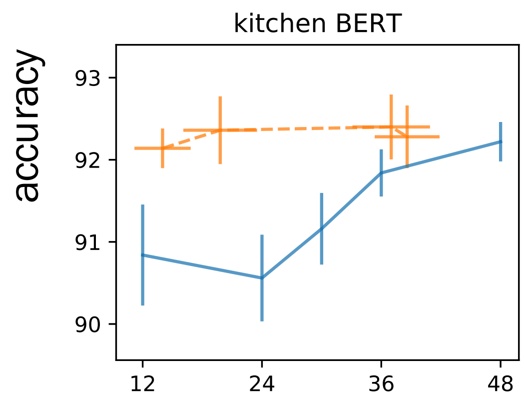


Baselines

	embeddings	*unigrams*	*bigrams*	*trigrams*	*4-grams*
baseline 1	GloVe	24			
baseline 2	GloVe		24		
baseline 3	GloVe			24	
baseline 4	GloVe				24
baseline 5	GloVe	6	6	6	6
baseline 6	BERT	12			
baseline 7	BERT		12		
baseline 8	BERT			12	
baseline 9	BERT				12
baseline 10	BERT	3	3	3	3

Classification Accuracy vs. # Transitions





Our method in orange; baselines in blue.

Visualization

A four-pattern model for the Amazon kitchen dataset (3300 training examples).

It achieves 92.0% accuracy; the best baseline was 90.8%.

		transition$_1$	**transition$_2$**	**transition$_3$**
Patt. 1	Top	are	perfect	$\ldots_{SL}$ [CLS]
		definitely	recommend	$\ldots_{SL}$ [CLS]
		excellent	product	$\ldots_{SL}$ [CLS]
		highly	recommend	$\ldots_{SL}$ [CLS]
	Bottom	not	$\ldots_{SL}$ [SEP]	$\ldots_{SL}$ [CLS]
		very	disappointing	$!_{SL}$ [SEP]$_{SL}$ [CLS]
		was	defective	$\ldots_{SL}$ had
		would	not	$\ldots_{SL}$ [CLS]
Patt. 2	Top	[CLS]	mine	broke
		[CLS]	it	$\ldots_{SL}$ heat
		[CLS]	thus	it
		[CLS]	it$_{SL}$ does	it$_{SL}$ heat
	Bottom	[CLS]	perfect	$\ldots_{SL}$ cold
		[CLS]	sturdy	$\ldots_{SL}$ cooks
		[CLS]	evenly	,$_{SL}$ withstand$_{SL}$ heat
		[CLS]	it	is
Patt. 3	Top	‘	pops	’$_{SL}$ ’$_{SL}$ escape
		‘	gave	out
		that	had	escaped
		‘	non	-
	Bottom	simply	does	not
		[CLS]	useless	equipment$_{SL}$!
		unit	would	not
		[CLS]	poor	to$_{SL}$ no
Patt. 4	Top	[CLS]	after	
		[CLS]	our	
		mysteriously	jammed	
		mysteriously	jammed	
	Bottom	[CLS]	i	
		[CLS]	i	
		[CLS]	i	
		[CLS]	we	

Summary

- Regularization techniques from pre-neural times can be applied to increase automation/speed and decrease footprint.

Parting Shots

- Interpretability matters!
  - NLP isn't just for researchers anymore.
  - It's hard to improve a model you don't understand.
- Constrained model families may lead to ...
  - better generalization (inductive bias)
  - guarantees (but not today)
- Computational cost matters!
  - Reducing energy footprint
  - Inclusiveness in research

see Schwartz et al.,
"Green AI,"
arXiv:1907.10597

Thanks!

- Drivers of this work:
  - Jesse Dodge (CMU LTI)
  - Hao Peng (UW CSE)
  - Roy Schwartz (UW CSE/AI2 → Hebrew University)
  - Sam Thomson (CMU LTI → Semantic Machines)
- Sponsors:
  - NSF IIS-1562364 and REU supplement
  - UW Innovation award
  - NVIDIA (GPU)