Exploration of the lexical capacity of recurrent neural networks

Gaël Le Godais[1,2], Tal Linzen[1], Emmanuel Dupoux[1]

April 12, 2017

[1]École Normale Supérieure / PSL Research University / EHESS / CNRS, France
[2]ENSIMAG, Grenoble

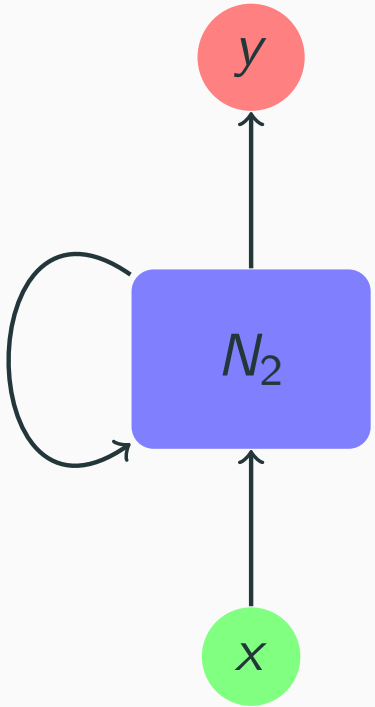
Outline

Text generating RNNs

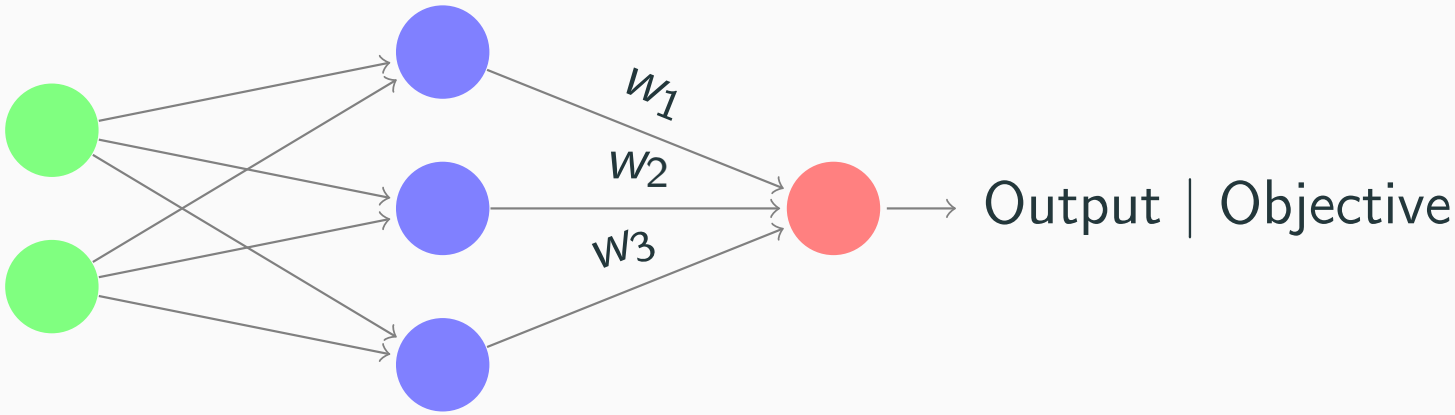
How to mesure the lexicon size of any system?

Lexical capacity of various neural networks

Recurrent neural networks



Training neural networks



$$loss = loss\_function(output, objective)$$

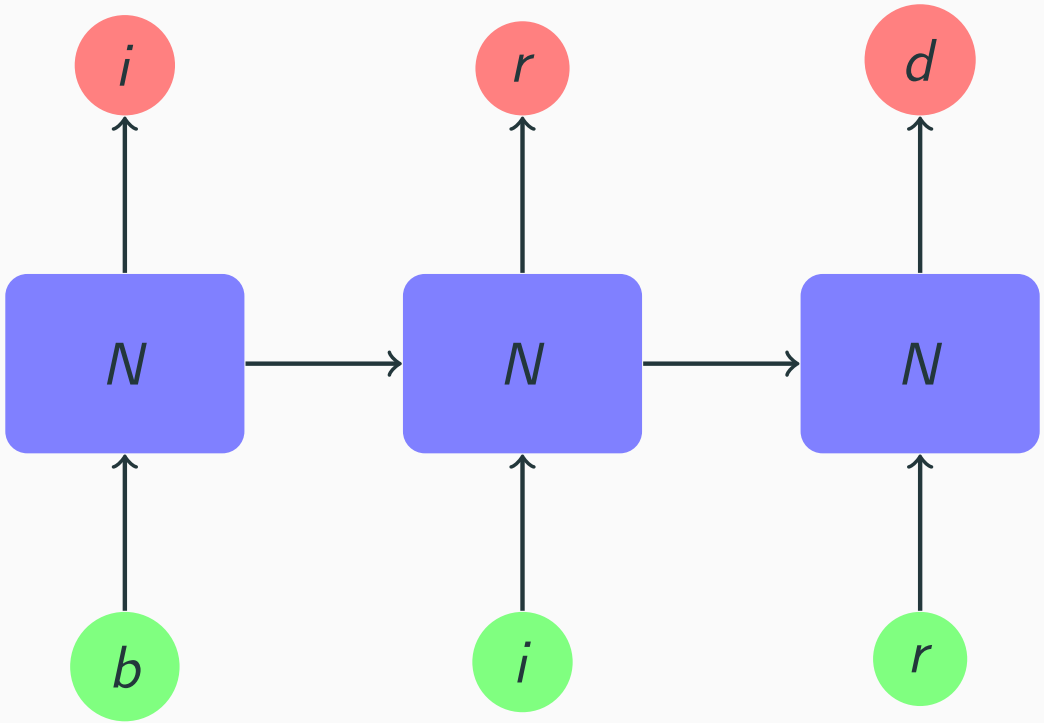
Training a character-level language model

Nearly a century before, a group of ordinary men s_

Training a character-level language model

early a century before, a group of ordinary men st_

Character-level language model using RNNs



An example from Karpathy

VIOLA:
Why, Salisbury must find his flesh and thought
That which I am not aps, not a man and in fire,
To show the reining of the raven and the wars
To grace my hand reproach within, and not a fair are hand,
That Caesar and my goodly father's world;
When I was heaven of presence and our fleets,
We spare with hours, but cut thy council I am great,
Murdered and by thy master's ready there
My power to give thee but so much as hell:
Some service in the noble bondman here,
Would show him to her wine.

Credit: *karpathy.github.io*

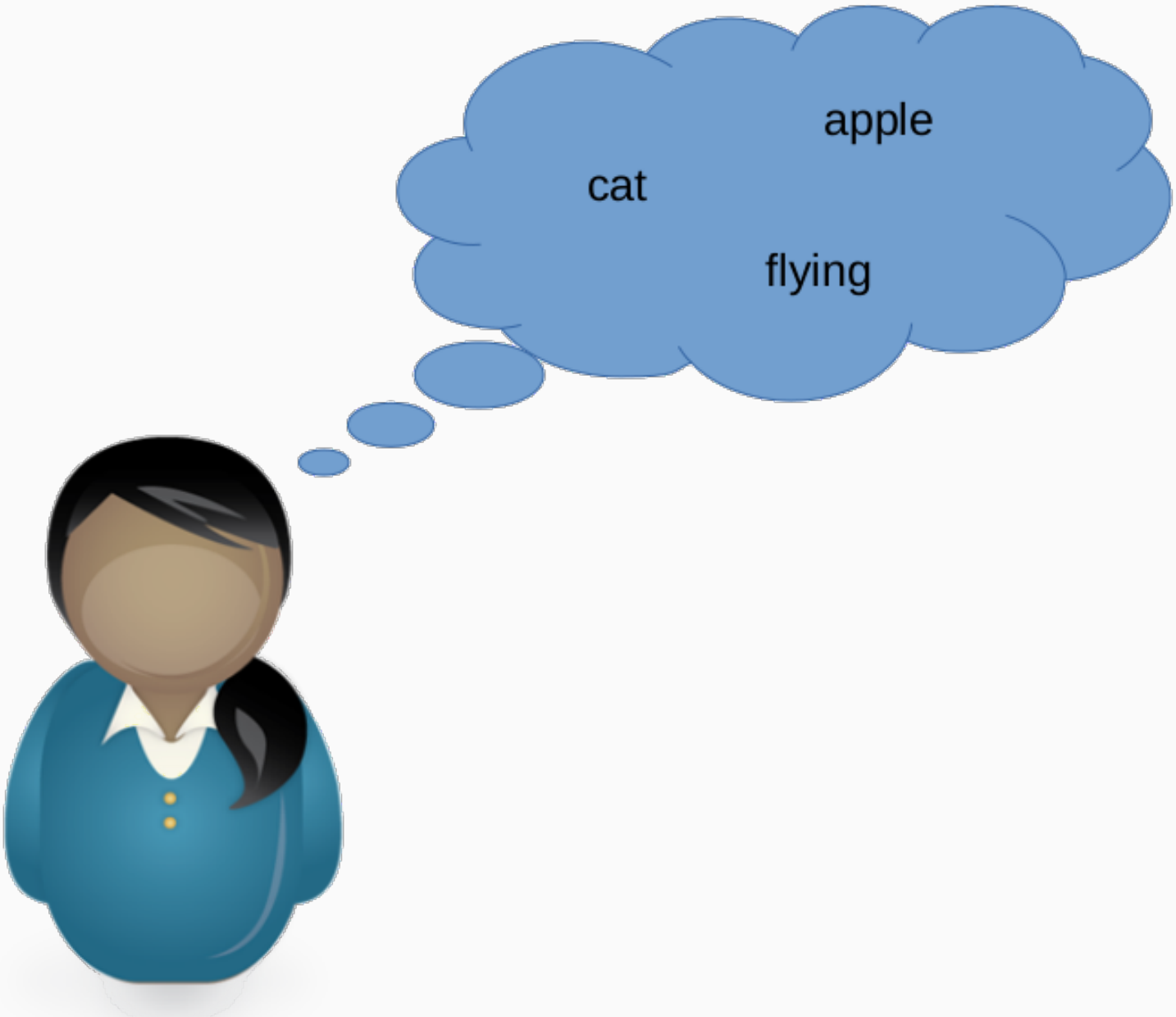
Outline

Text generating RNNs

How to mesure the lexicon size of any system?

Lexical capacity of various neural networks

The human case



Lexical decision task

fish

Lexical decision task

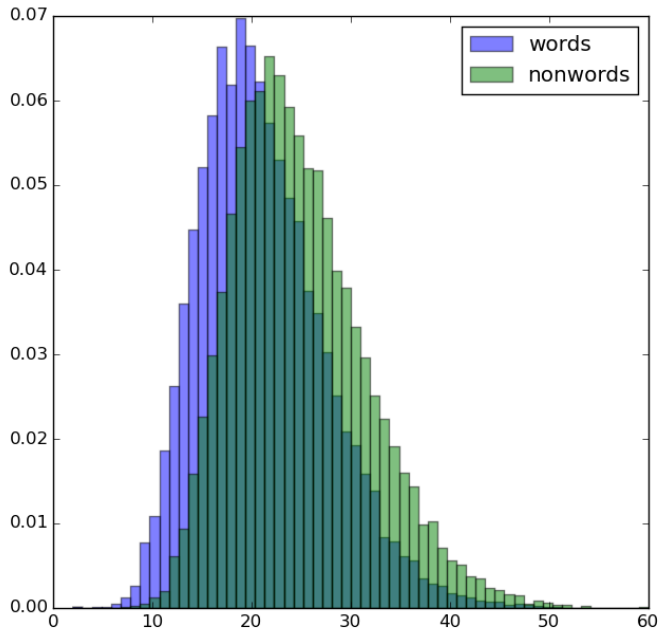
wug

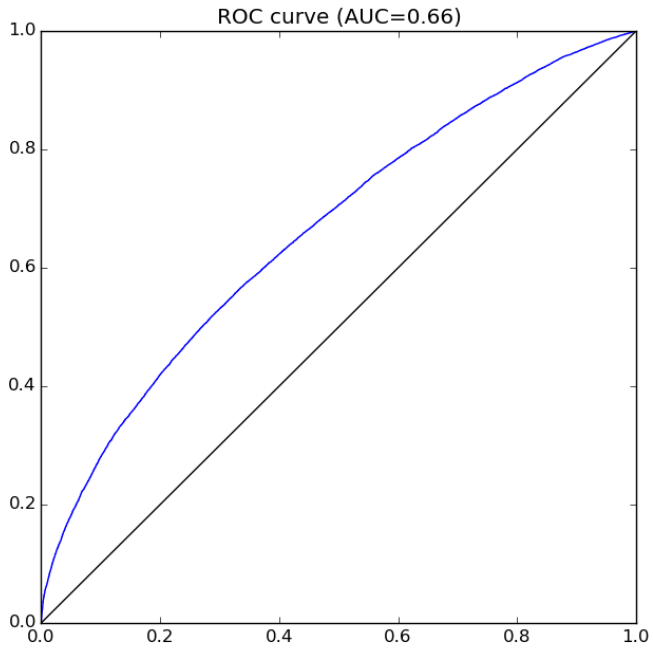
The neural network case

Lexical indicator

$$\textit{indicator} : \text{word} \rightarrow \textit{numerical score}$$

Evaluation of a lexical indicator





ROC Curve

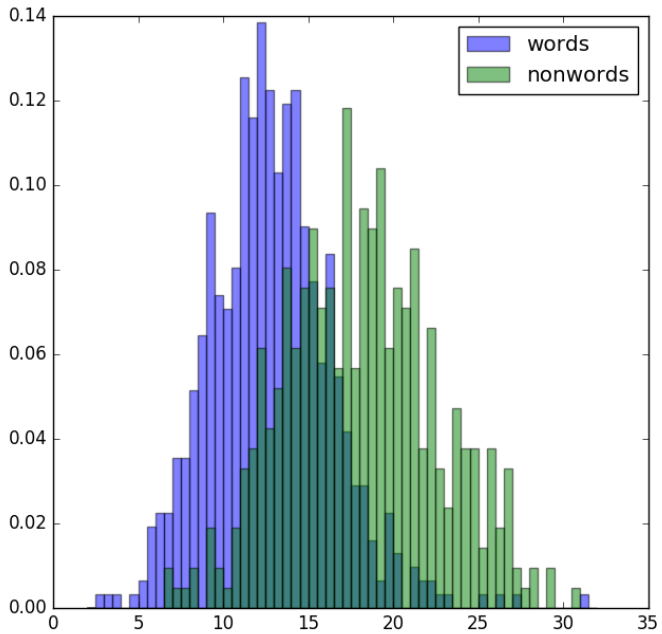
$$TPR = \frac{\text{number of actual words classified as words}}{\text{number of words}}$$

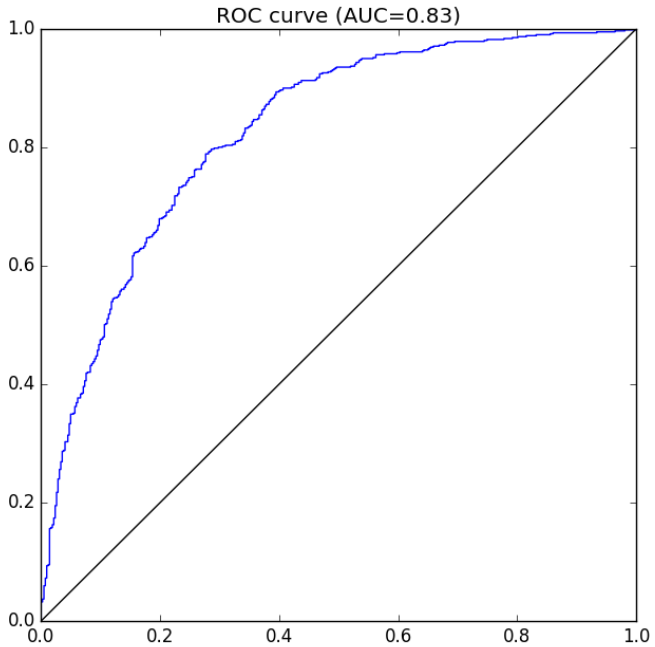
$$FPR = \frac{\text{number of non words classified as words}}{\text{number of non words}}$$

Choice of lexical indicator

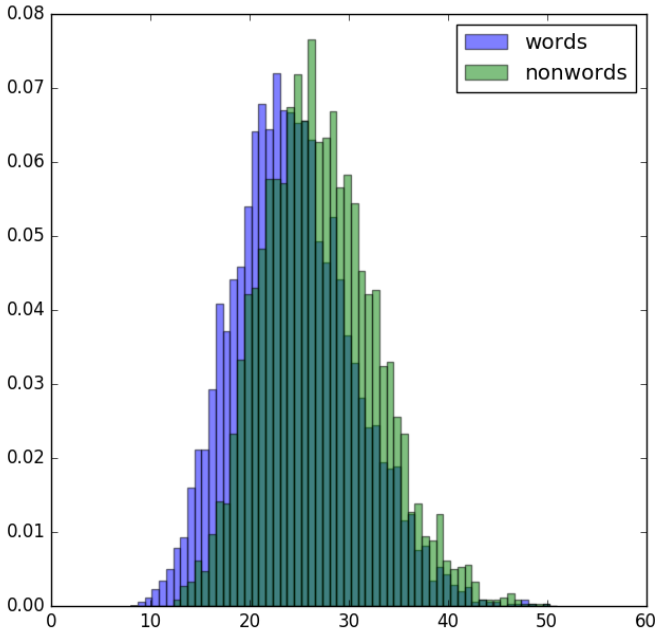
$$\mathit{indicator}(\text{wug}) = -\log p(\_\,\mathit{wug}\,\_)$$

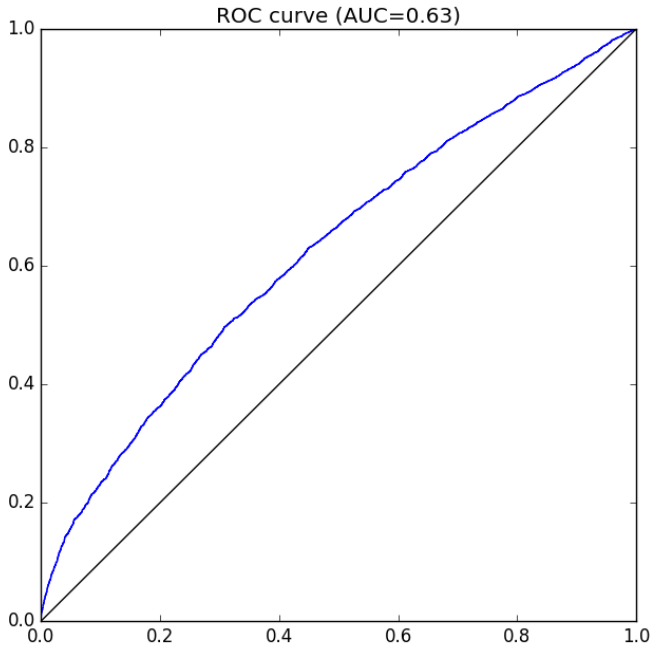
Lexical decision: 3-letter words





Lexical decision: 8-letter words





Spot-the-Word[1]

clear **knick**

Lexical capacity

$$accuracy = \frac{correct\ \ decisions}{total\ \ number\ \ of\ \ decisions}$$

$$lexical\ \ capacity = 100 * (2 * accuracy - 1)$$

Lexicon size

$$lexical\ capacity = 100 * (2 * accuracy - 1)$$

$$lexicon\ size = lexical\ capacity * total\ number\ of\ words\ encountered$$

Outline

Text generating RNNs

How to mesure the lexicon size of any system?

Lexical capacity of various neural networks

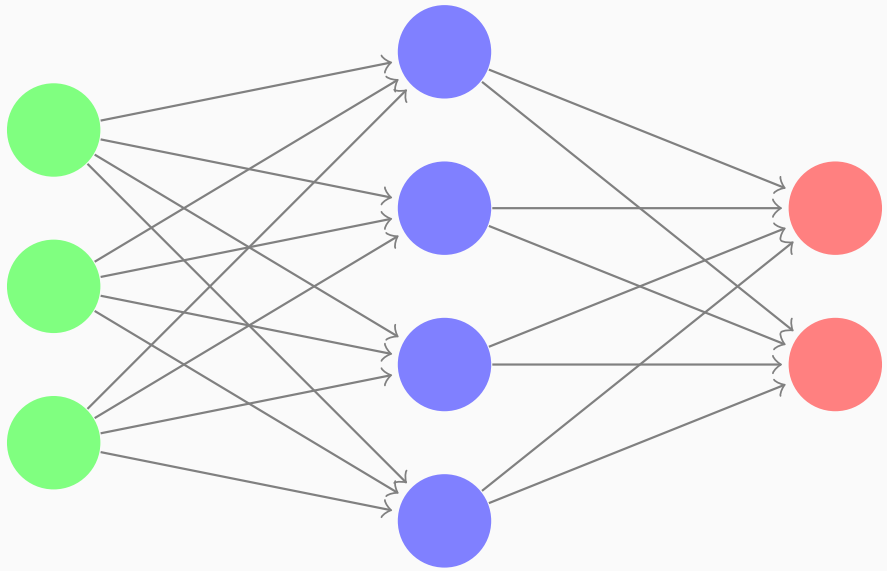
Networks

Corpus subset of the moviebook project[5]

Architecture RNN, LSTM

Layers 1, 2, 3

Hidden units 2, 4, 8, 16, 32, 64, 128, 256, 512



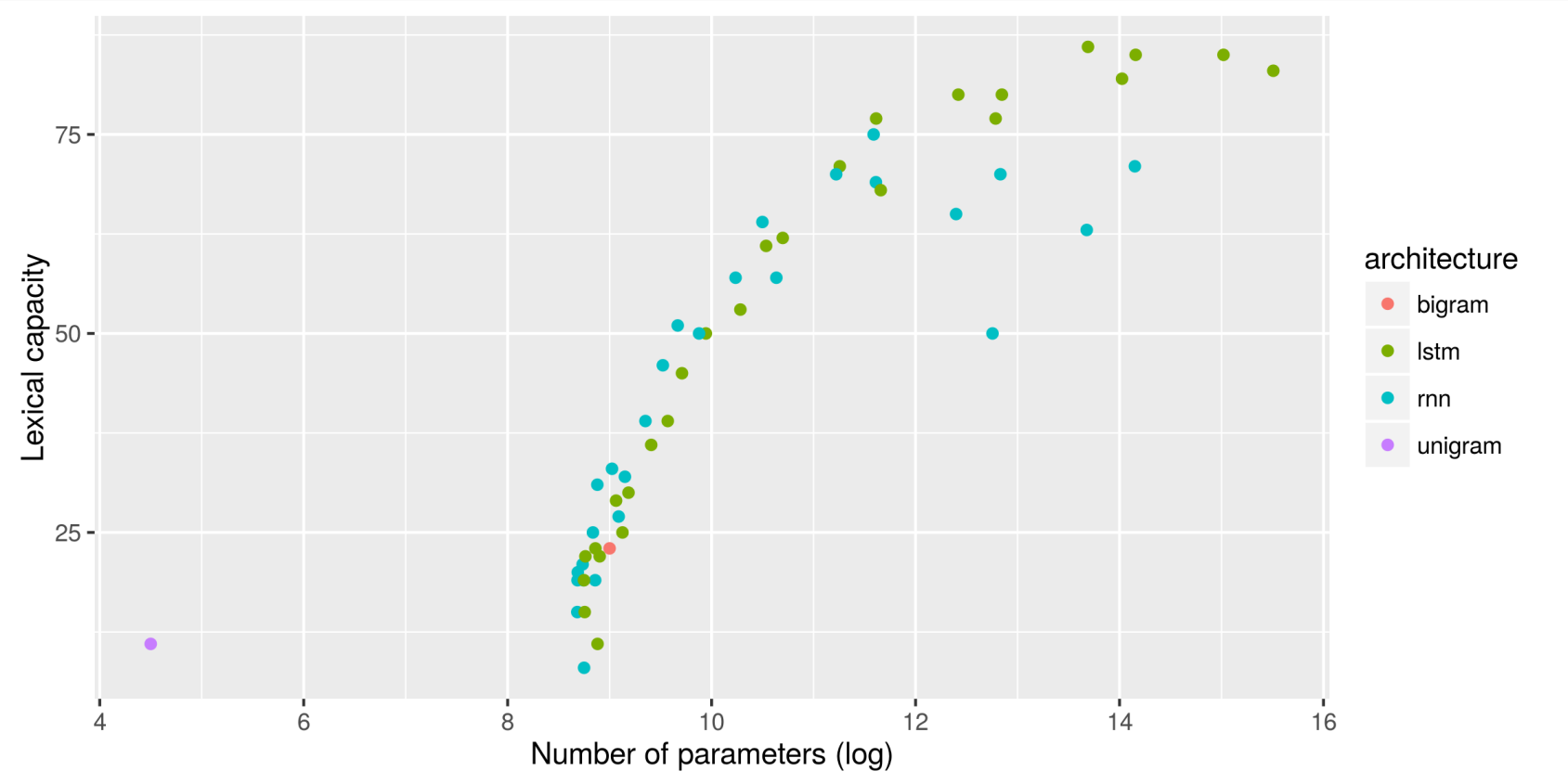
Nonwords generation with Wuggy[3]

Word	Match	Forced_Choice	Word_Freq	Dist	Word_Prob
tearing	fleaing	Right	6.69	3	7.81e-08
tearing	wooling	Wrong	6.69	4	7.81e-08
tearing	wresing	Right	6.69	4	7.81e-08
tearing	sureing	Right	6.69	4	7.81e-08
tearing	scabing	Wrong	6.69	3	7.81e-08
clear	strep	Right	234	5	9.50e-05
clear	psych	Right	234	5	9.50e-05
clear	splen	Right	234	4	9.50e-05
clear	stran	Right	234	4	9.50e-05

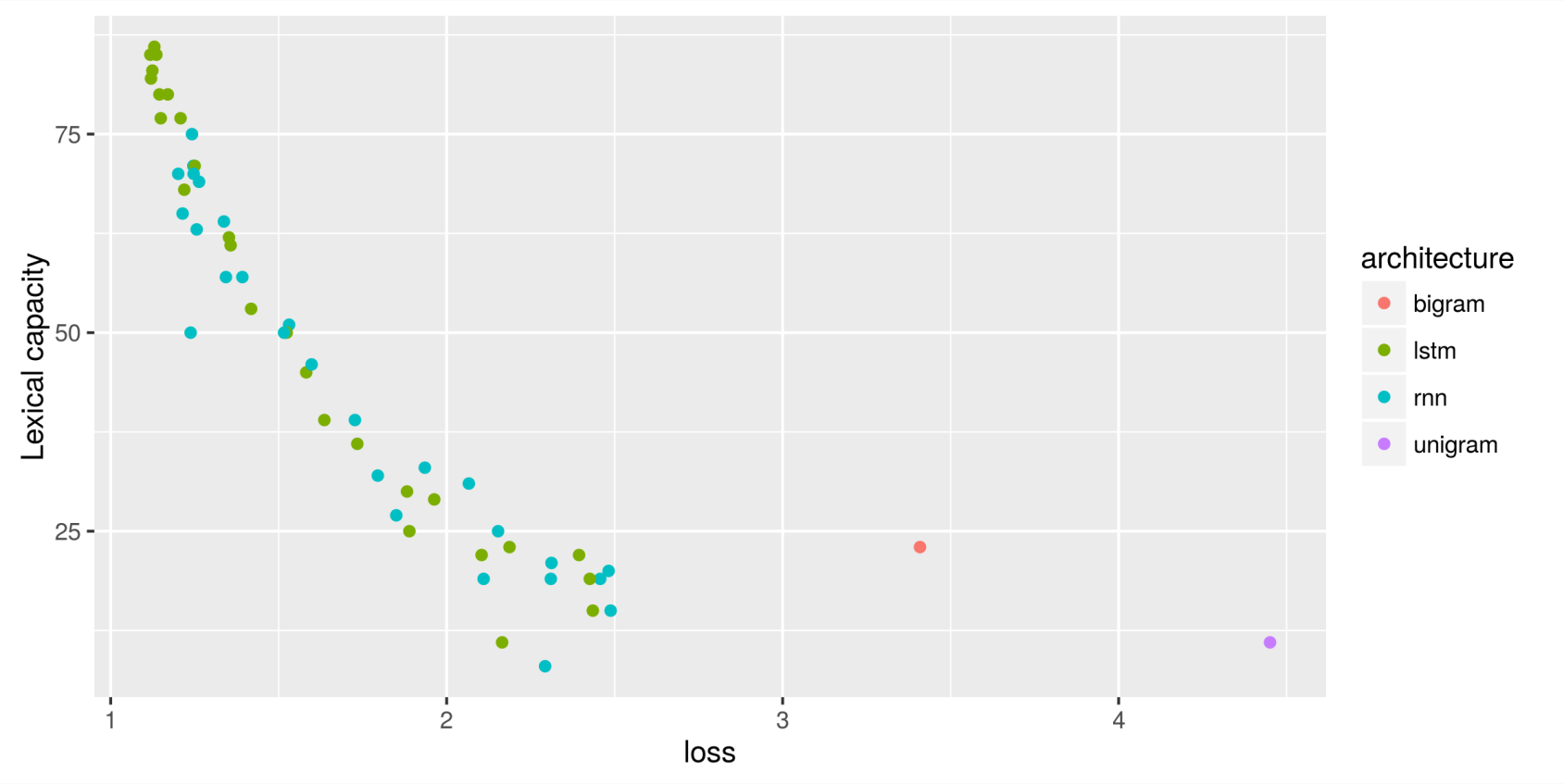
Raw data

architecture	units	layers	accuracy	lex_cap	n_params	loss
lstm	16	3	72	45	16470	1.582
lstm	256	1	88	77	356822	1.149
lstm	4	3	61	22	7350	2.104
lstm	512	2	92	85	3332566	1.136
rnn	32	2	73	46	13654	1.598
lstm	8	3	65	30	9750	1.882
lstm	32	3	80	61	37590	1.357
lstm	16	2	69	39	14326	1.636
lstm	256	2	93	86	882646	1.13

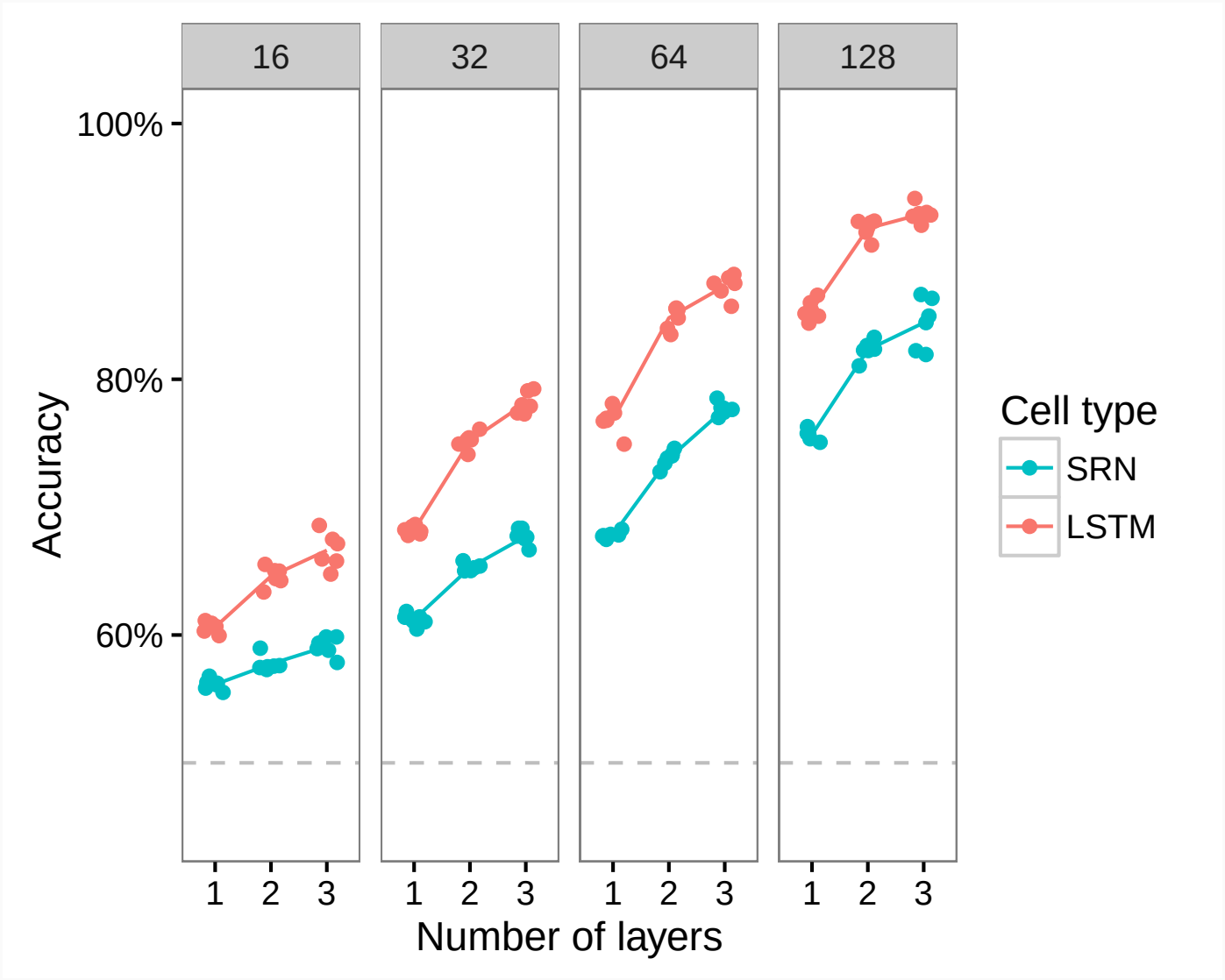
Baselines


Lexical capacity
75
50
25
4
6
8
10
12
14
16
Number of parameters (log)
architecture
bigram
lstm
rnn
unigram

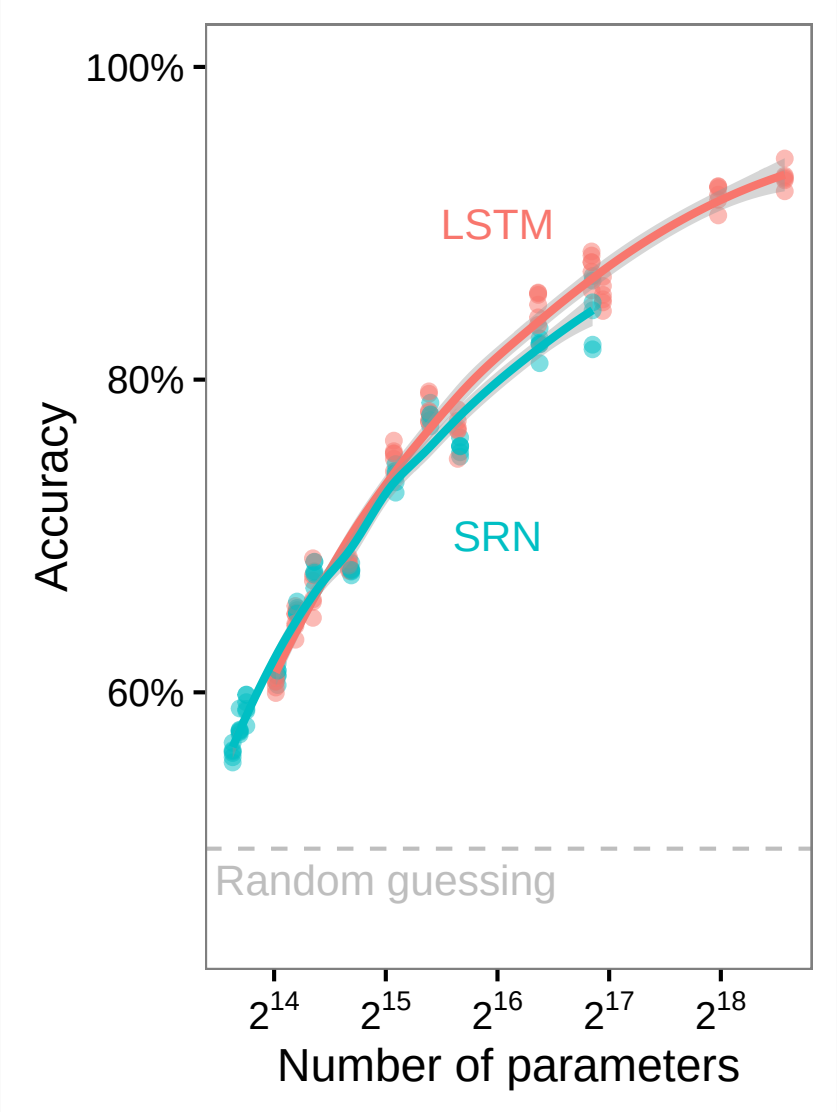
Baselines


Lexical capacity
75
50
25
1
2
3
4
loss
architecture
bigram
lstm
rnn
unigram

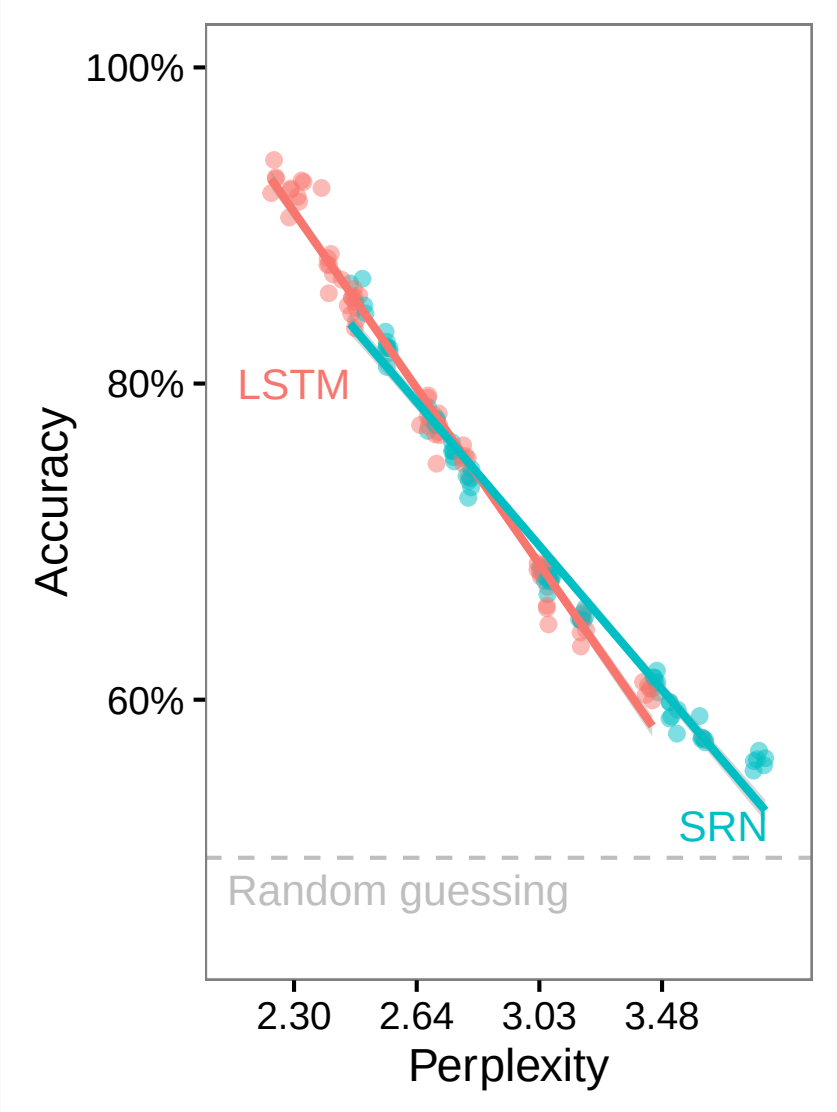
Breakdown


16
32
64
128
100%
80%
60%
Accuracy
1
2
3
Number of layers
Cell type
SRN
LSTM

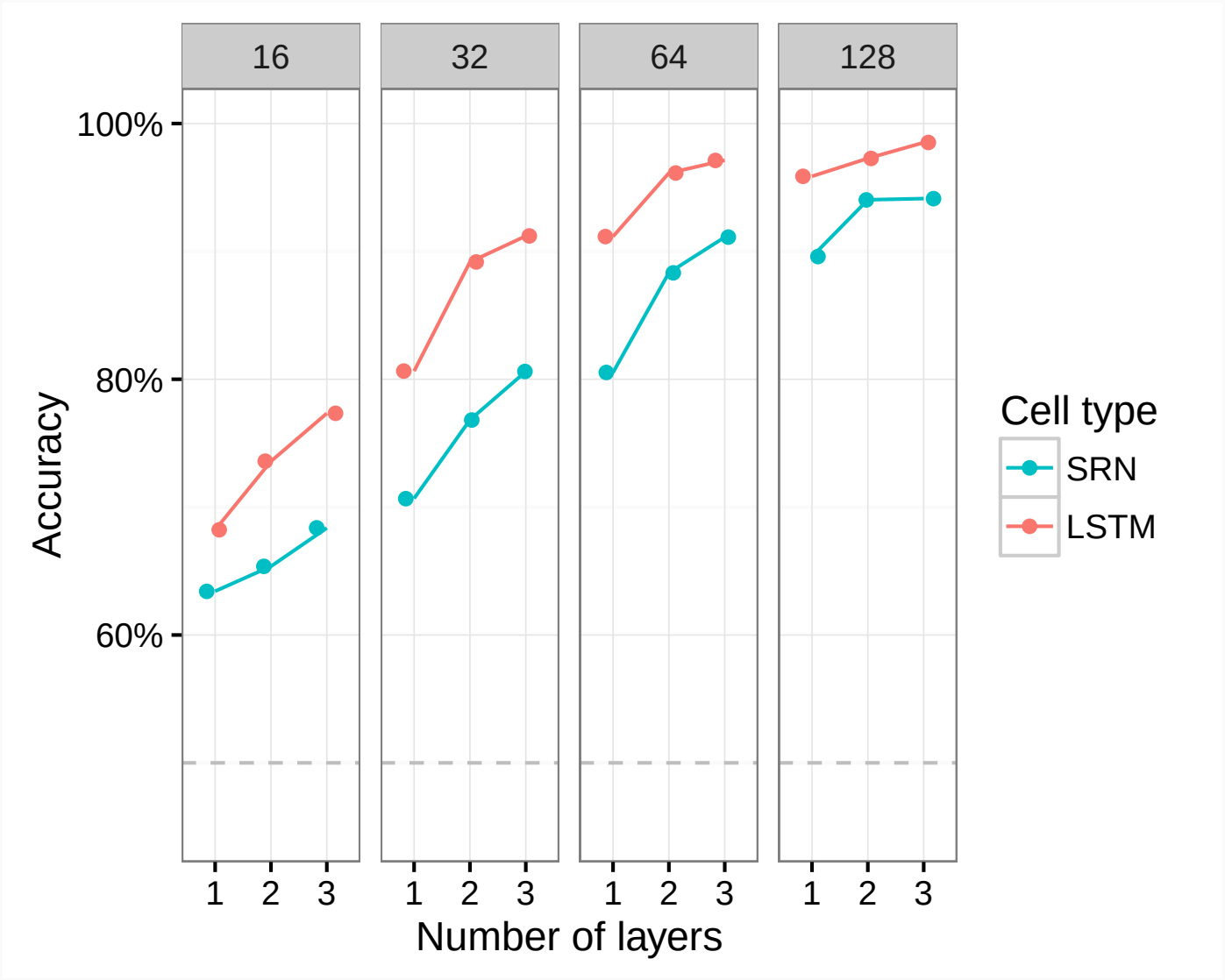
Effect of the number of parameters



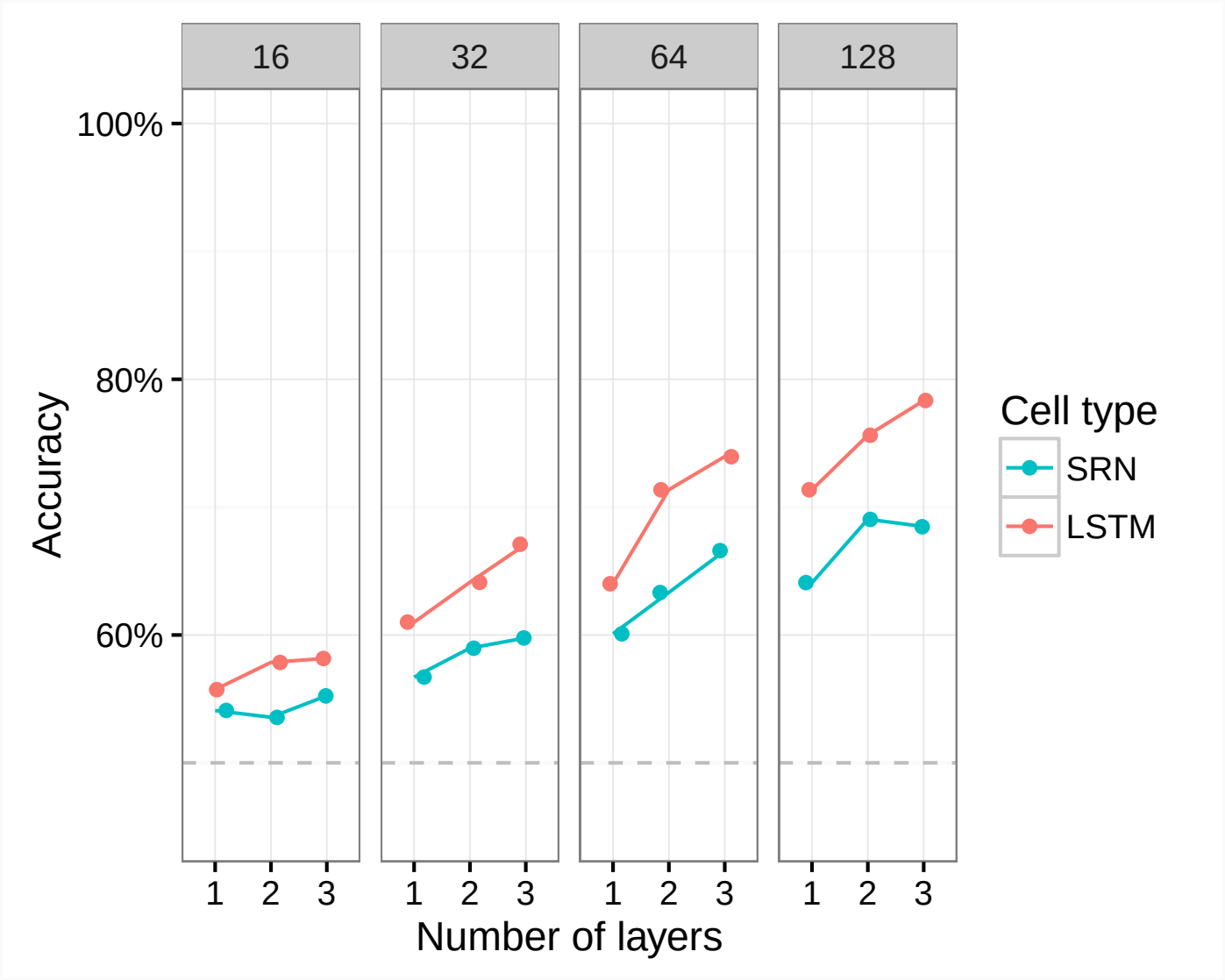
Accuracy vs Perplexity



Memory capacity

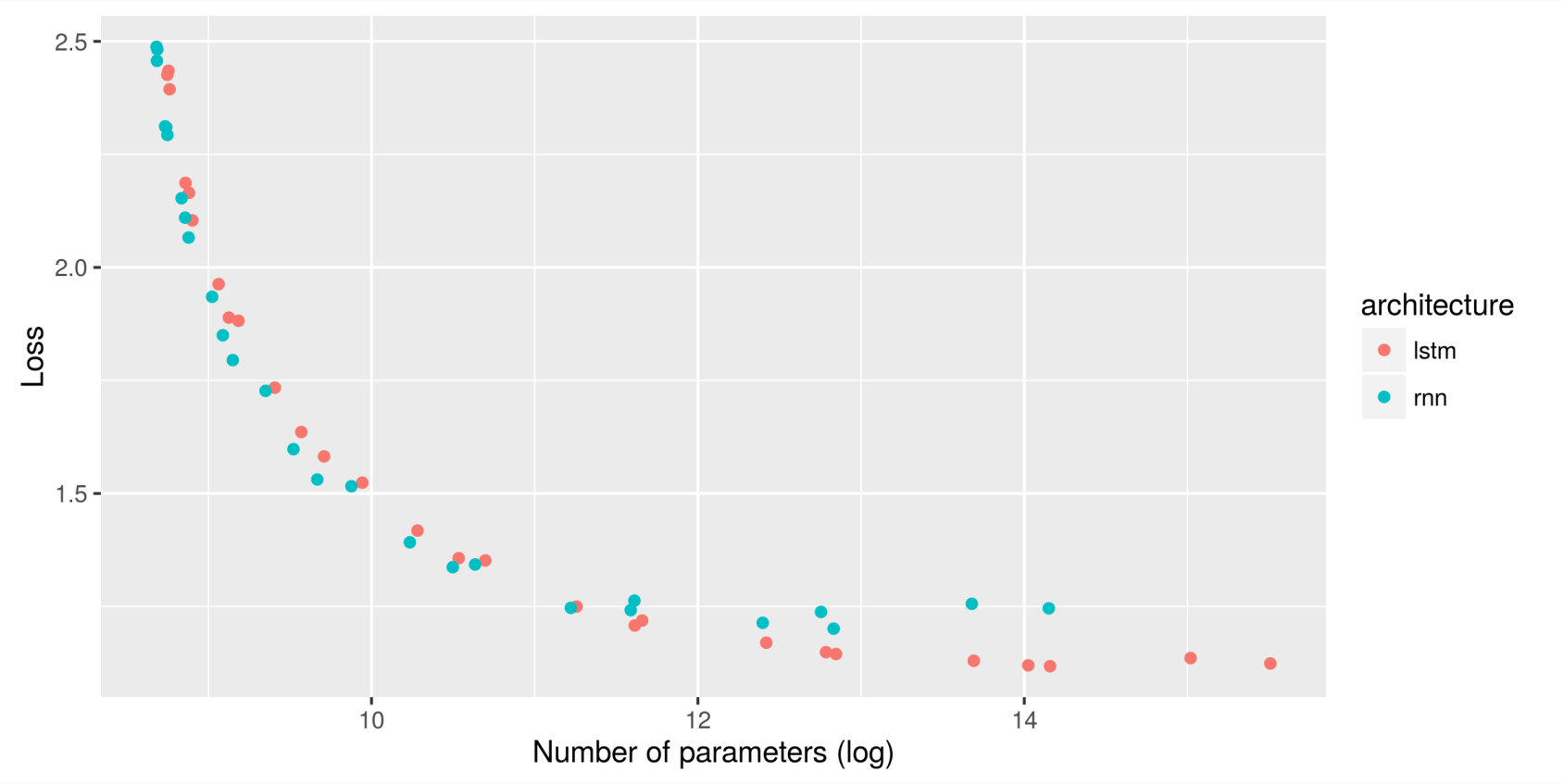


Generative capacity

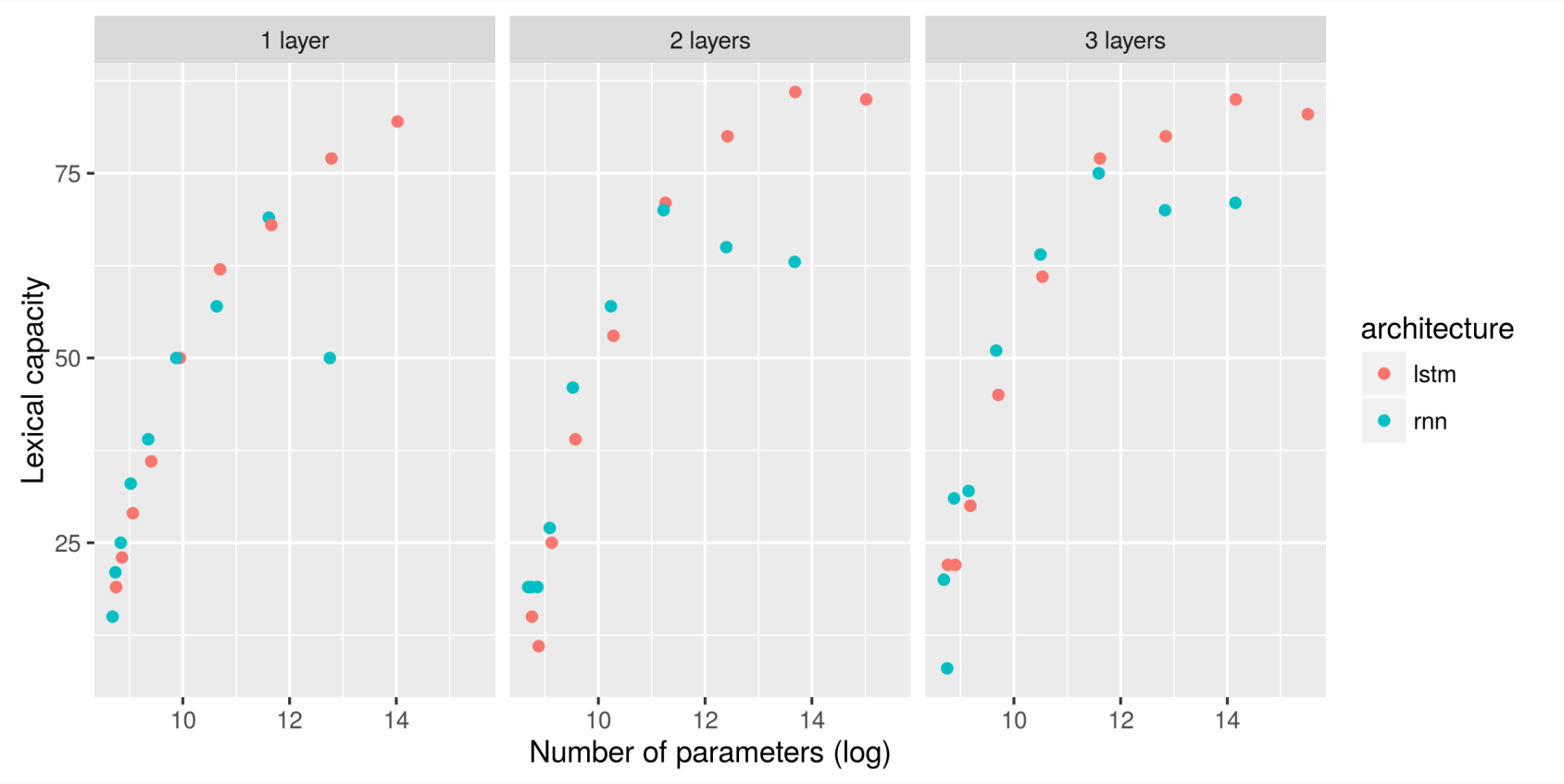


Questions?

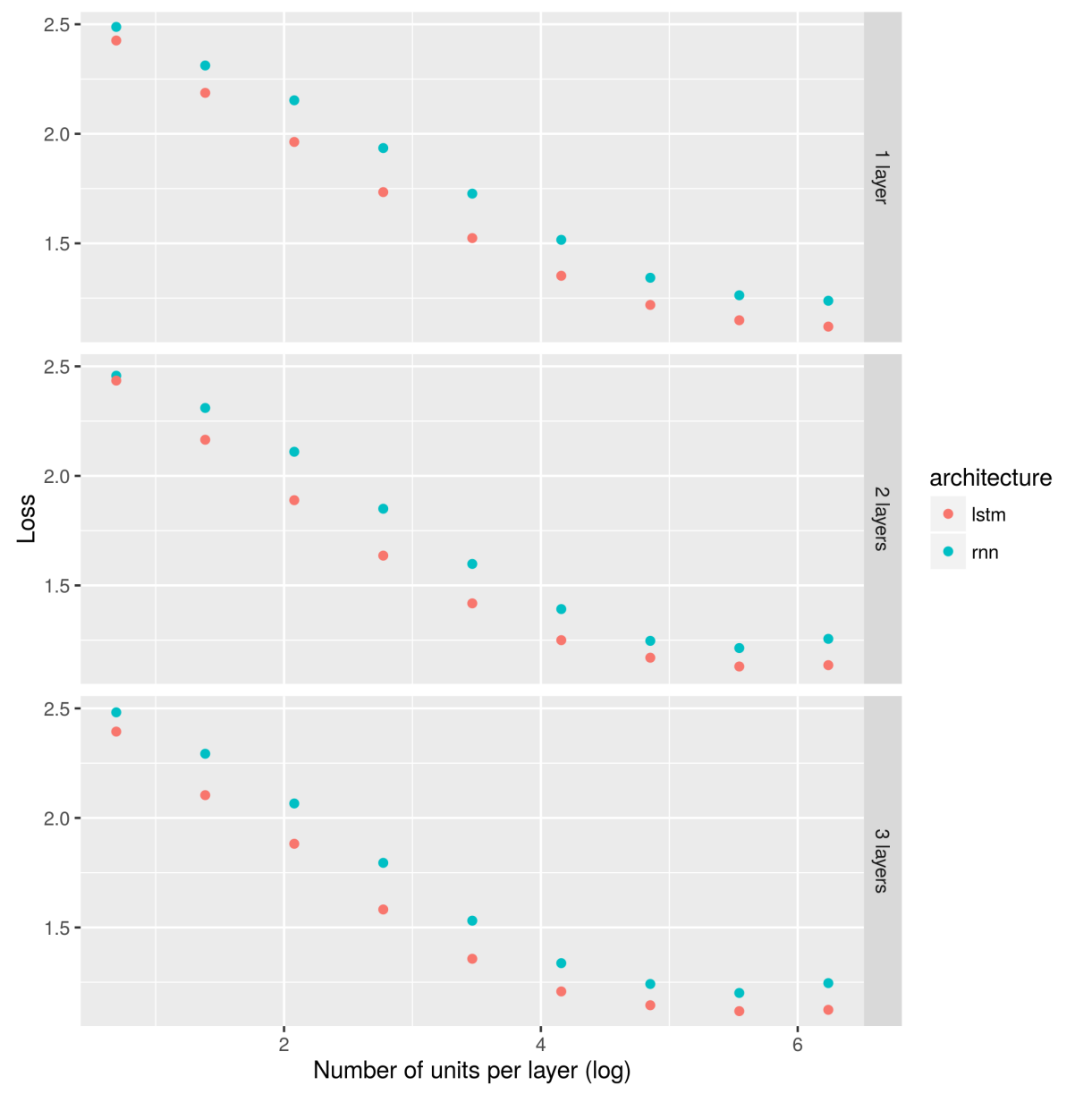
Effect of the number of parameters



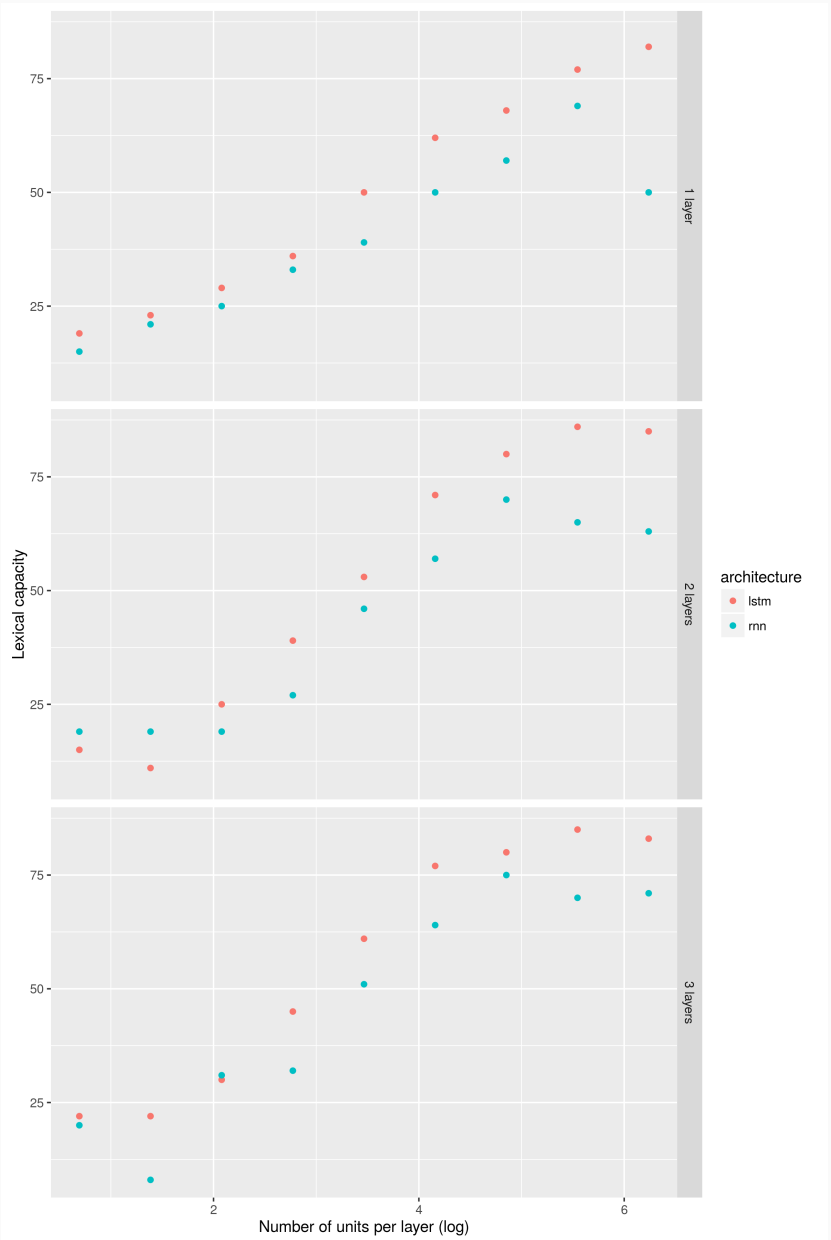
Effect of the number of parameters



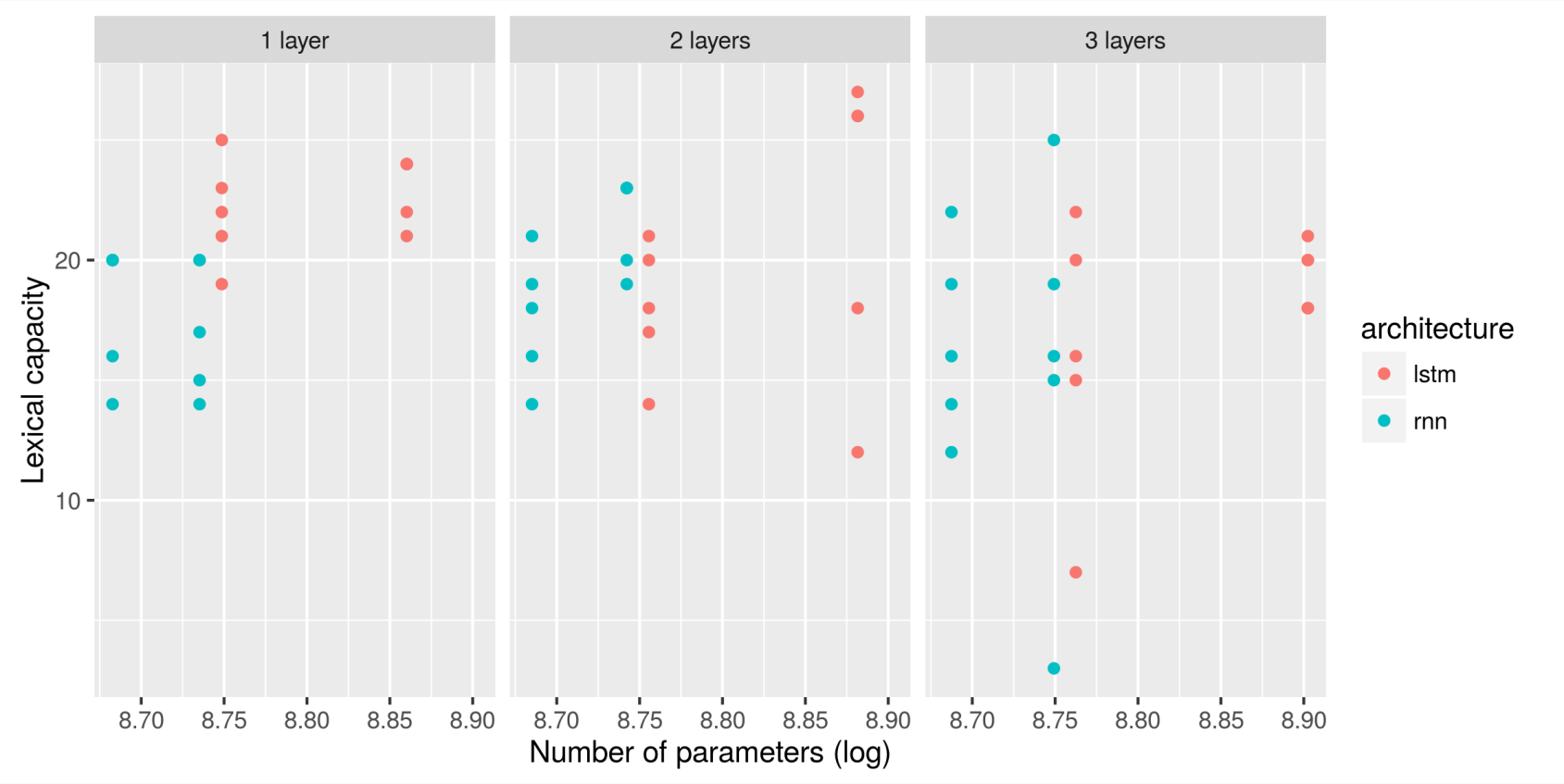
Effect of the number of units per layer



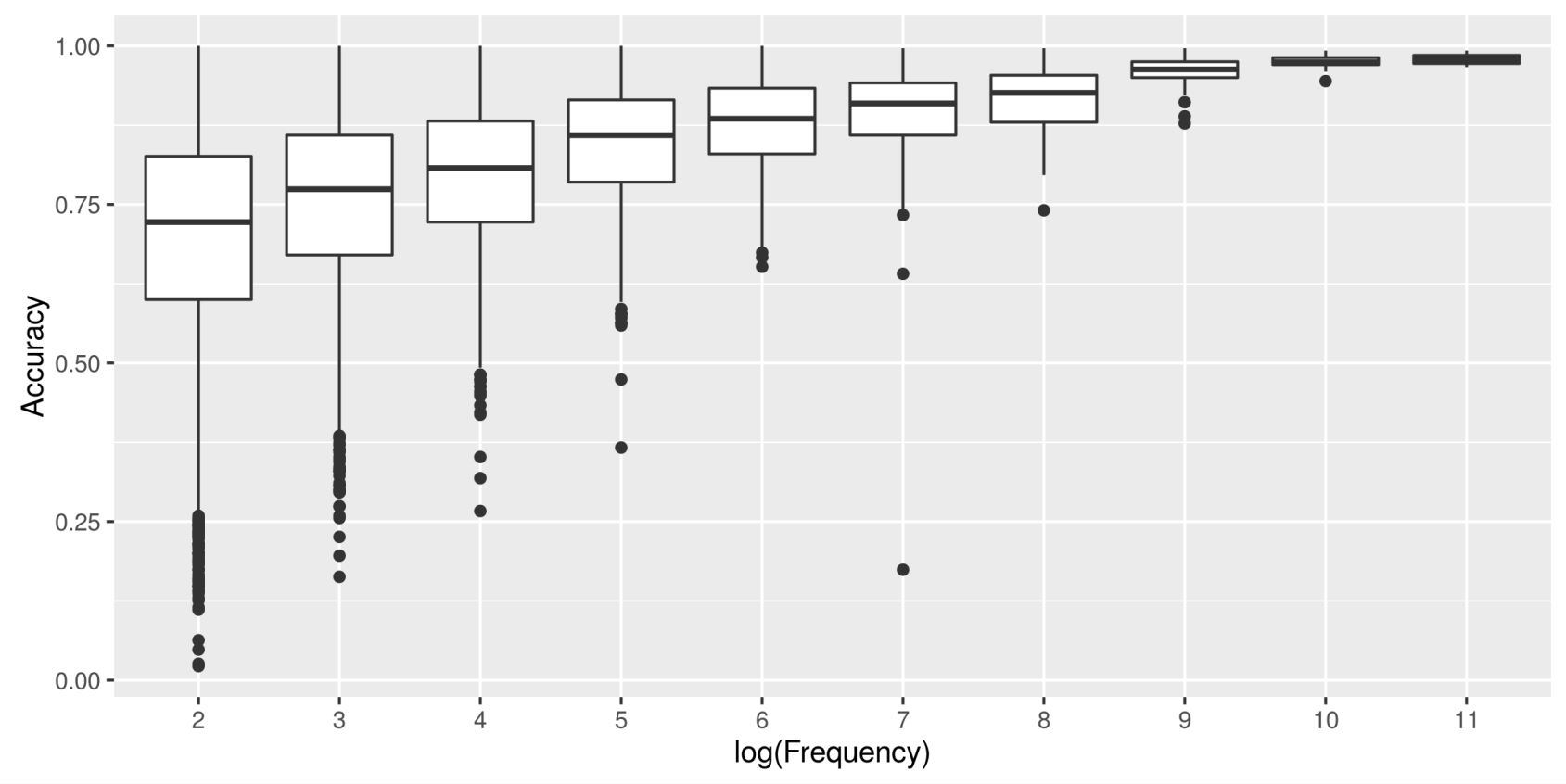
Effect of the number of units per layer



Noise


1 layer
2 layers
3 layers
Lexical capacity
20
10
8.70
8.75
8.80
8.85
8.90
Number of parameters (log)
architecture
lstm
rnn

Effect of word frequency



References I

A. Baddeley, H. Emslie, and I. Nimmo-Smith.
The spot-the-word test: A robust estimate of verbal intelligence based on lexical decision.
British Journal of Clinical Psychology, 32(1):55–65, 1993.

R. F.
The perceptron-a perceiving and recognizing automaton.
Report 85-460-1, Cornell Aeronautical Laboratory., 1957.

E. Keuleers and M. Brysbaert.
Wuggy: A multilingual pseudoword generator.
Behavior Research Methods, 42(3):627–633, 2010.

References II

K. P.K.

Early language acquisition: cracking the speech code.

Nature Reviews Neuroscience, 5:186–197, 2004.

Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler.

Aligning books and movies: Towards story-like visual explanations by watching movies and reading books.

In *arXiv preprint arXiv:1506.06724*, 2015.