# Structured Sparsification of Gated Recurrent Neural Networks

Ekaterina Lobacheva*[1] Nadezhda Chirkova*[1] Alexander Markovich[2] Dmitry Vetrov[1,3]

[1] Samsung-HSE Laboratory, Higher School of Economics;

[2] Higher School of Economics; [3] Samsung AI Center Moscow * Equal contribution

Moscow, Russia

## Overview

**Goal:** learn structured sparsity in gated RNNs (LSTM, GRU, …)

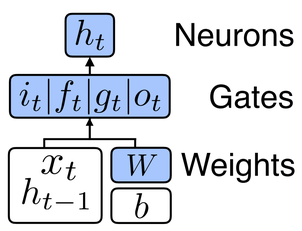
Structured sparsity in RNNs:

1. Learn sparse weight matrices [1] → reduce model size
2. Remove neurons by removing all the weights associated with these neurons [2] → faster prediction

**Our approach:**

add intermediate level of sparsification — make gates constant → simplify LSTM cell & reveal specifics of LSTM workflow

**3 sparsity levels:**

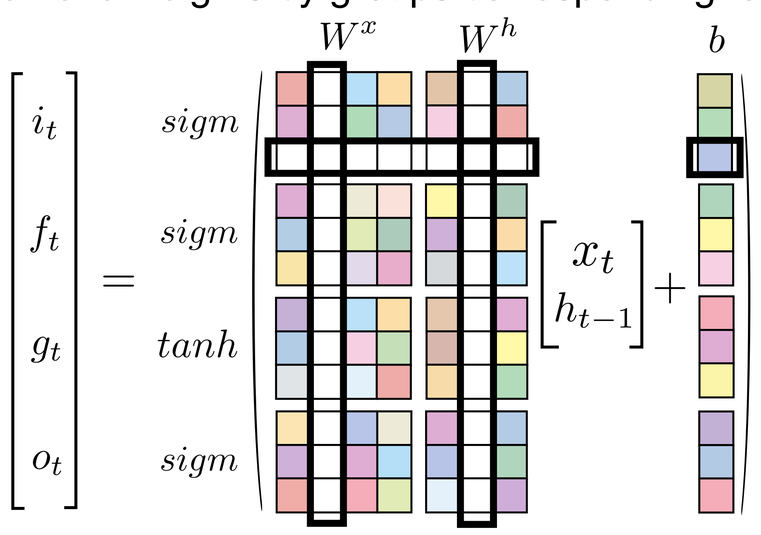



**Results:**

- Improved neuron-wise compression
- LSTM uses different gate structures to solve different tasks and to model long / short dependencies

## Main idea

Remove weights by groups corresponding to gates / neurons:




$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \qquad h_t = o_t \odot tanh(c_t)$$

zero column

no **outgoing** edges

⇩

remove neuron

zero row

no **ingoing** edges

⇩

constant gate

## Implementation in two sparsification frameworks

**Pruning [2]:**

apply group Lasso to the rows and columns of the LSTM weight matrix + apply Lasso to the individual weights → prune weights based on a threshold

**Bayesian sparsification [1], [3], [4]:**

modify computational graph — introduce group weights $\mathbf{z}$:

$$f_t = \sigma\left(\left(W_f^h(h_{t-1} \odot \mathbf{z^h}) + W_f^x(x_t \odot \mathbf{z^x})\right) \odot \mathbf{z^f} + b_f\right)$$

similarly for $i_t, o_t, g_t$

log-uniform prior over $\omega = \{W, z\}$: $p(\omega) = \prod_k \frac{1}{|\omega_k|}$ → learn approximate posterior over $\omega$: $q(\omega) = \prod_k \mathcal{N}(\mu_k, \sigma_k)$ → remove weights and group weights with $\sigma_k \gg \mu_k$

## Experiments: quantitative

### Text classification

IMDb / AGNews

Emb → LSTM → FC on $h_T$

### Text generation

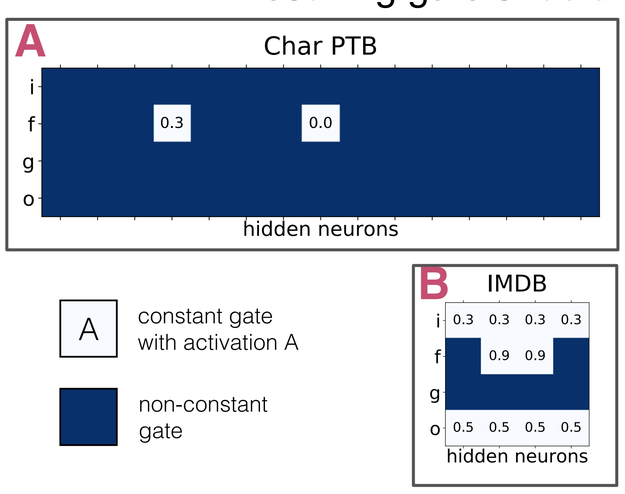
PTB char-level / word-level

LSTM → FC / Emb → 2 x LSTM → FC

| Task | Method | Quality | Compr. | Neurons | Gates |
|---|---|---|---|---|---|
| IMDb | Original | **84.1** | 1x | 128 | 512 |
| Accuracy | Bayes W+N [1]+[3] | 83.98 | 17874x | 5 | 12 |
| % | Bayes W+G+N | 83.98 | **19747x** | **4** | **6** |
| AGNews | Original | **90.6** | 1x | 512 | 2048 |
| Accuracy | Bayes W+N [1]+[3] | 88.55 | 645x | 17 | 62 |
| % | Bayes W+G+N | 88.41 | **647x** | **14** | **39** |
| Char PTB | Original | 1.454 | 1x | 1000 | 4000 |
| Bits-per | Bayes W+N [1]+[3] | 1.430 | **10.2x** | **390** | **1560** |
| -char | Bayes W+G+N | **1.425** | 9.8x | 404 | 1563 |
| Word PTB (small) Perplexity | Original | 114.41 | 1x | 200 – 200 | 800 – 800 |
| | Bayes W+N [1]+[3] | 104.81 | **11.65x** | 68 – 110 | 272 – 392 |
| | Bayes W+G+N | **104.45** | 11.44x | **52 – 108** | **197 – 349** |
| | Prun. W+N [2] | 106.25 | 1.44x | 72 – 123 | 288 – 492 |
| | Prun. W+G+N | **105.64** | **1.49x** | **64 – 115** | **193 – 442** |
| Word PTB (large) Perplexity | Original | 78.57 | 1x | 1500 – 1500 | 6000 – 6000 |
| | Prun. W+N [2] | **77.62** | 2.97x | 324 – 394 | 1296 – 1576 |
| | Prun. W+G+N | **77.82** | **3.22x** | **252 – 394** | **881 – 1418** |

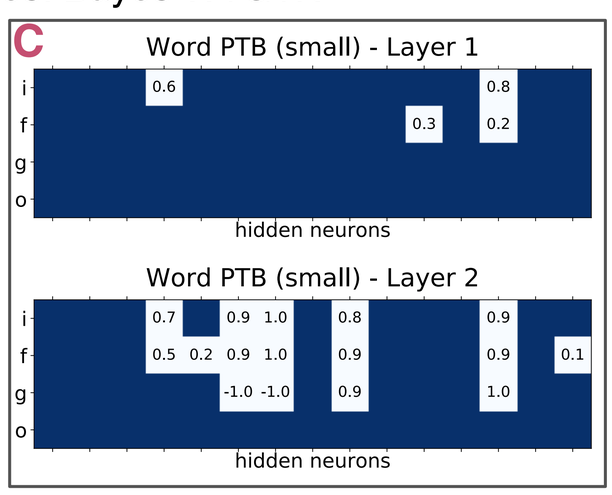
## Experiments: qualitative

Resulting gate structures: Bayes W+G+N

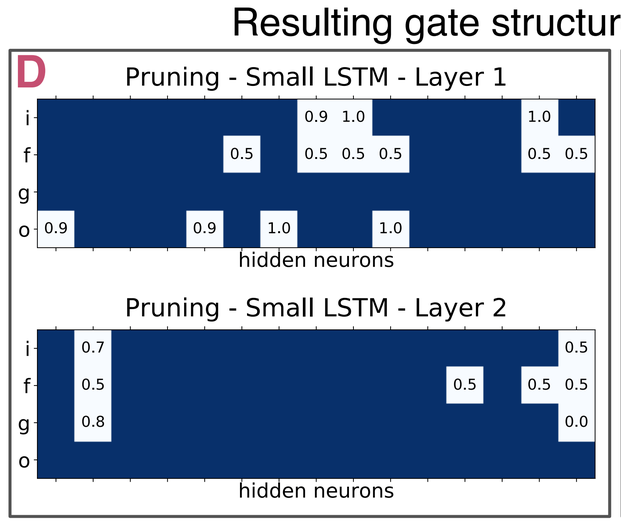


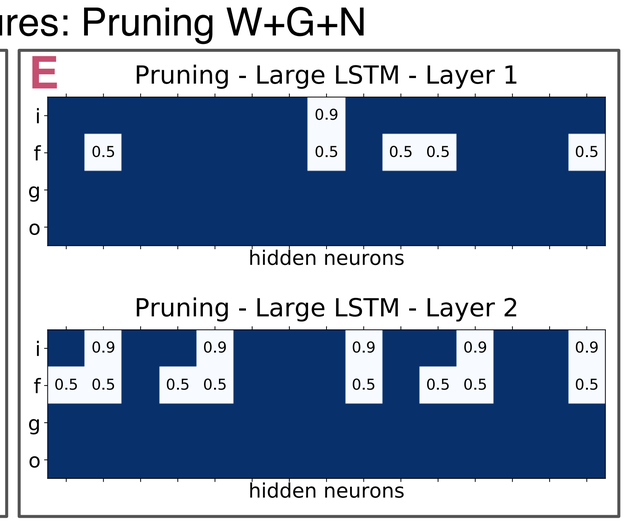







Resulting gate structures: Pruning W+G+N




A — constant gate with activation A

non-constant gate

**Resulting gate structures vary for different tasks: compare A, B, C**

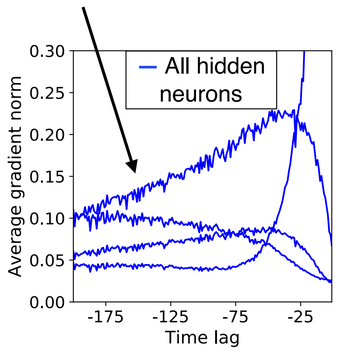
char-level language model (A): all gates non-constant

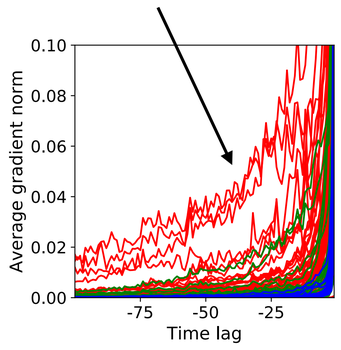
classification (B): non-constant g — long dependencies are highly important

word-level language model, 2nd layer (C): all gates non-constant — full LSTM units — long dependencies

word-level language model, 2nd layer (C): only non-constant o — vanilla recurrent units — short dependencies

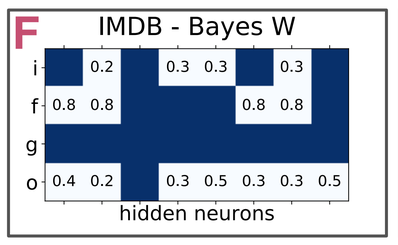
Averaged over validation sequences, norm of the gradients of hidden neurons w. r. t. LSTM input for different time-lag

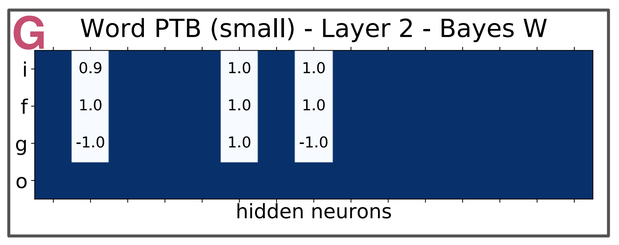



Hidden neurons:
- Only output gate non-constant
- All gates non-constant
- Other

Resulting gate structures: Bayes W




## Gate structure intrinsically exists in LSTM:

- for the same task, similar gate structures are obtained in different sparsification frameworks (compare **C and D**)
- characteristic patterns in gate structures are obtained even with unstructured sparsification (compare **B and F**; **C and G**), but with lower overall compression

## References

[1] Chirkova, N. et al. Bayesian compression for natural language processing. EMNLP 2018

[2] Wen, W. et al. Learning intrinsic sparse structures within long short-term memory. ICLR 2018

[3] Louizos, C. et al. Bayesian compression for deep learning. NeurIPS 2017

[4] Molchanov D. et al. Variational dropout sparsifies deep neural networks. ICML 2017