Jan. 17, 2020

CS 886 Deep Learning and NLP

Ming Li

CONTENT

Last two lectures: The bigger the better!

This lecture: The smaller the better!

04

Theory of Simplicity, ALBERT and SHA-RNN

LECTURE FOUR

Plan

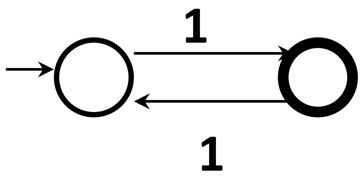
1. Why simpler?
2. ALBERT
3. SHA-RNN

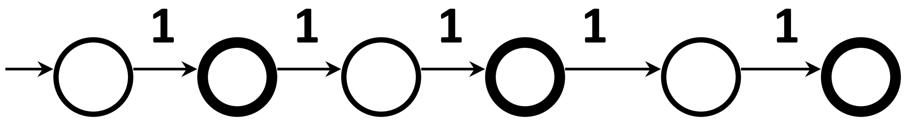
The Importance of being small

1. Occam’s Razor: Entities should not be multiplied beyond necessity.
2. I. Newton: Nature is pleased simplicity

One example: Inferring a DFA

- Given data that a DFA accepts: 1, 111, 11111, 1111111; and rejects: 11, 1111, 111111. What is it?





- There are actually infinitely many DFAs satisfying these data.
- The first DFA makes a nontrivial inductive inference,
- The 2nd does not. It “over fits” the data, can’t make further predictions.

History of Science

Maxwell's (1831-1879)'s equations say that in 1865:

- (a) An oscillating magnetic field gives rise to an oscillating electric field;
- (b) an oscillating electric field gives rise to an oscillating magnetic field.

Item (a) was known from M. Faraday's experiments. However (b) is a theoretical inference by Maxwell and his aesthetic appreciation of simplicity. The existence of such electromagnetic waves was demonstrated by the experiments of H. Hertz in 1888, 8 years after Maxwell's death, and this opened the new field of radio communication. Maxwell's theory is even relativistically invariant. This was long before Einstein’s special relativity. As a matter of fact, it is even likely that Maxwell's theory influenced Einstein’s 1905 paper on relativity which was actually titled `On the electrodynamics of moving bodies’.

Bayesian Inference

Bayes Formula:

$$P(H|D) = P(D|H)P(H)/P(D)$$

Take -log, maximize P(H|D) becomes minimize:

$$-\log P(D|H) - \log P(H) \quad \text{(modulo } \log P(D)\text{, constant).}$$

Where, by Shannon-Fano Theorem,

$-\log P(D|H)$ is the coding length of D given H.

$-\log P(H)$ is the coding length of model H

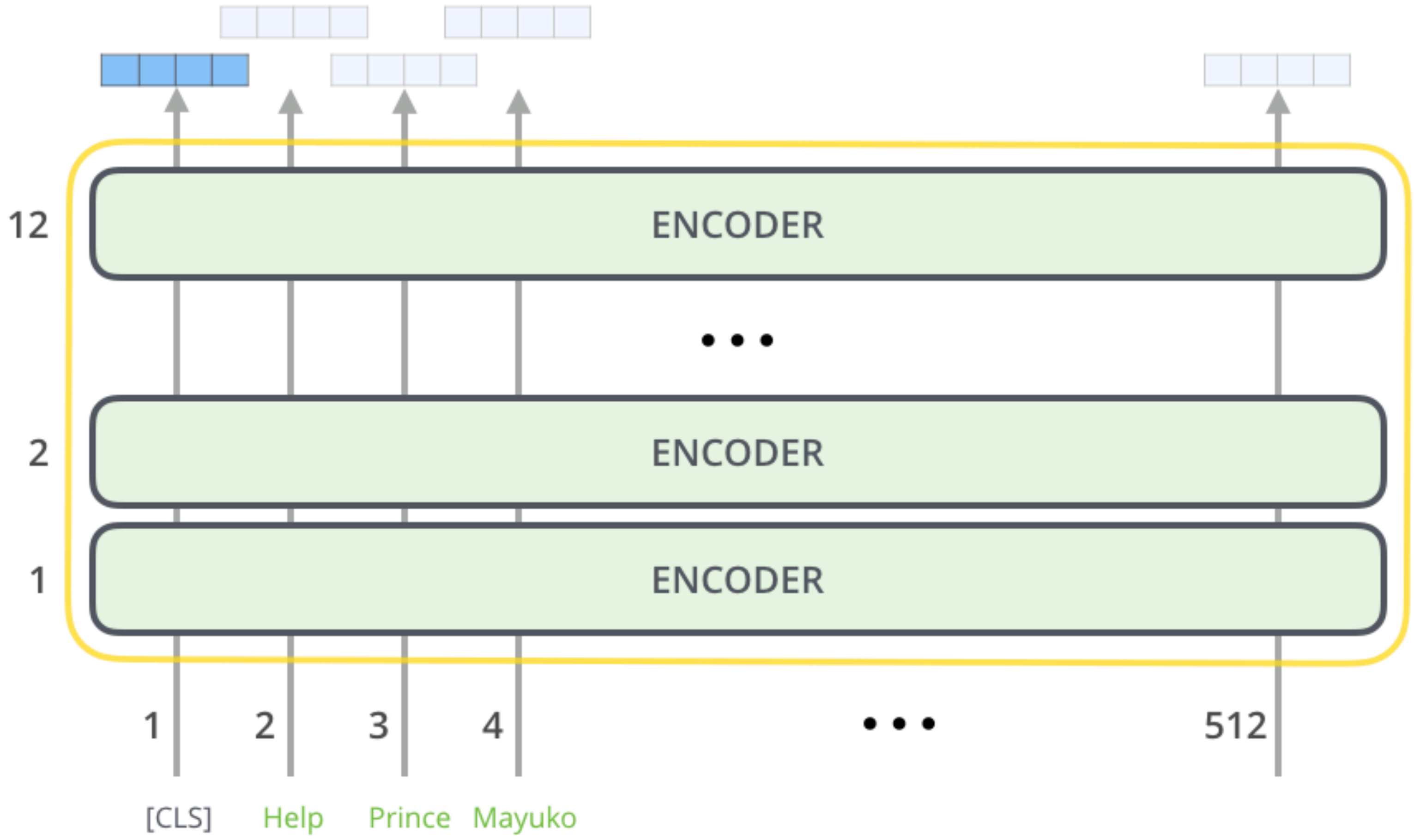
Thus, to maximize the probability is the same as minimizing the model length (and error description length).

PAC Learning theory / Statistical Inference

Given a set of data, if you have a model to fit the data, then the smaller the model is, the more likely it is to be correct. Such a statement can be proved formally, but it is not our focus here.

The key message I wish to deliver is: if you can do the same work with a smaller (neuron network) model, it will be most likely better.

We have studied Transformer, GPT-2, and especially BERT.



BERT

Input/output vector

size 768 or 1024

We have observed: The bigger model we get better results.

However, we have just learned the theory: the smaller the model is, the more likely we are close to ground truth.

So, what is the problem?

Could it be that “the ground truth model” is large?

But how much data have we used to train a human kid?

At 150 wpm (average speech speed), a human child of 10 year old has heard:

150 x 30 (min) x 5 (hours) x 365 (days) x 10 (years) ≈ 80M words

BERT used: 2500M words from Wikipedia, and 11,038 books (1000M words) = 3500M words, 43 times of human kid. At least, this shows the ground truth model is not that large. We should search for smaller models. Research project: what is the size of human language model? (Create a pseudo model to test BERT vs SHA-RNN)

ALBERT

1. Separate word embedding size (128 now) from hidden layer vector size (768). This saves Vx(768-128) weights.
2. Cross-layer parameter sharing: including attention part and feed forward network. I.e. all layers share same parameters. Doing this costs 1.5% loss of accuracy, but significantly saved parameters (so that ALBERT can add more layers)
3. Inter-sentence coherence loss: Replace "next sentence" prediction by "sentence order prediction".

4. Training data: Wikipedia: 2500M words; BookCorpus. Total 16GB text.

ALBERT Results

Model		Parameters	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg	Speedup
BERT	base	108M	90.5/83.3	80.3/77.3	84.1	91.7	68.3	82.1	17.7x
	large	334M	92.4/85.8	83.9/80.8	85.8	92.2	73.8	85.1	3.8x
	xlarge	1270M	86.3/77.9	73.8/70.5	80.5	87.8	39.7	76.7	1.0
ALBERT	base	12M	89.3/82.1	79.1/76.1	81.9	89.4	63.5	80.1	21.1x
	large	18M	90.9/84.1	82.1/79.0	83.8	90.6	68.4	82.4	6.5x
	xlarge	59M	93.0/86.5	85.9/83.1	85.4	91.9	73.9	85.5	2.4x
	xxlarge	233M	**94.1/88.3**	**88.1/85.1**	**88.0**	**95.2**	**82.3**	**88.7**	1.2x

Single Headed Attention RNN

1. Author's motivation: Alternative route of research?
2. My motivation: We should always look for simplicity.

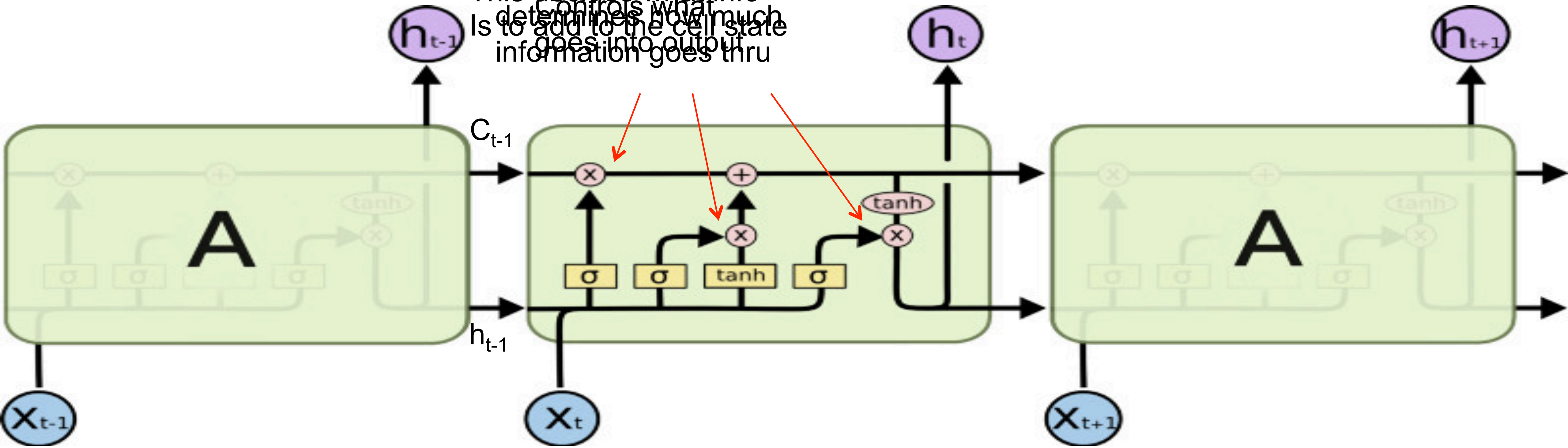
3. Here, we go back to the old approach of LSTM to raise sufficient doubt that Transformer is the only way.

LSTM

This sigmoid gate
determines how much
information goes thru

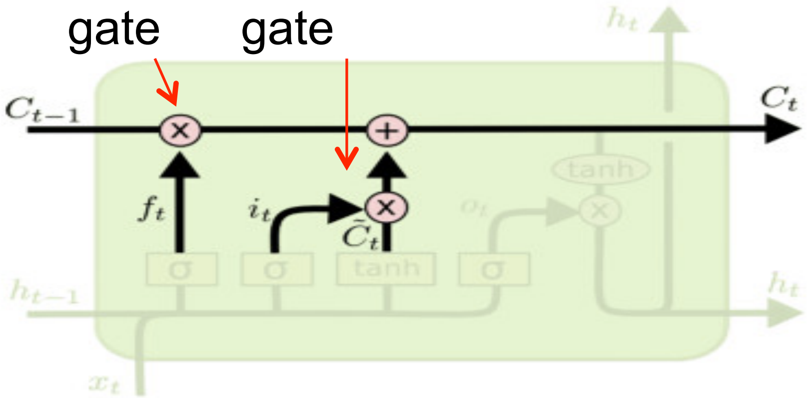
This decides what info
Is to add to the cell state

Output gate
Controls what
goes into output



Forget gate

input gate



The core idea is this cell state C_t, it is changed slowly, with only minor linear interactions. It is very easy for information to flow along it unchanged.

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Bits Per Character (BPC)

1. BPC is average cross-entropy
2. T is length of string. P_t is true distribution. n is (character) alphabet size. $P_i(x_j)$=1 iff i=j.

$$bpc(string) = \frac{1}{T}\sum_{t=1}^{T} H(P_t, \hat{P}_t) = -\frac{1}{T}\sum_{t=1}^{T}\sum_{c=1}^{n} P_t(c)\log_2 \hat{P}_t(c),$$

$$= -\frac{1}{T}\sum_{t=1}^{T}\log_2 \hat{P}_t(x_t).$$

Results of Single Headed Attention RNN

Enwik8:
100M characters.
90M train
5M validation
5M test

The Single Headed Attention RNN

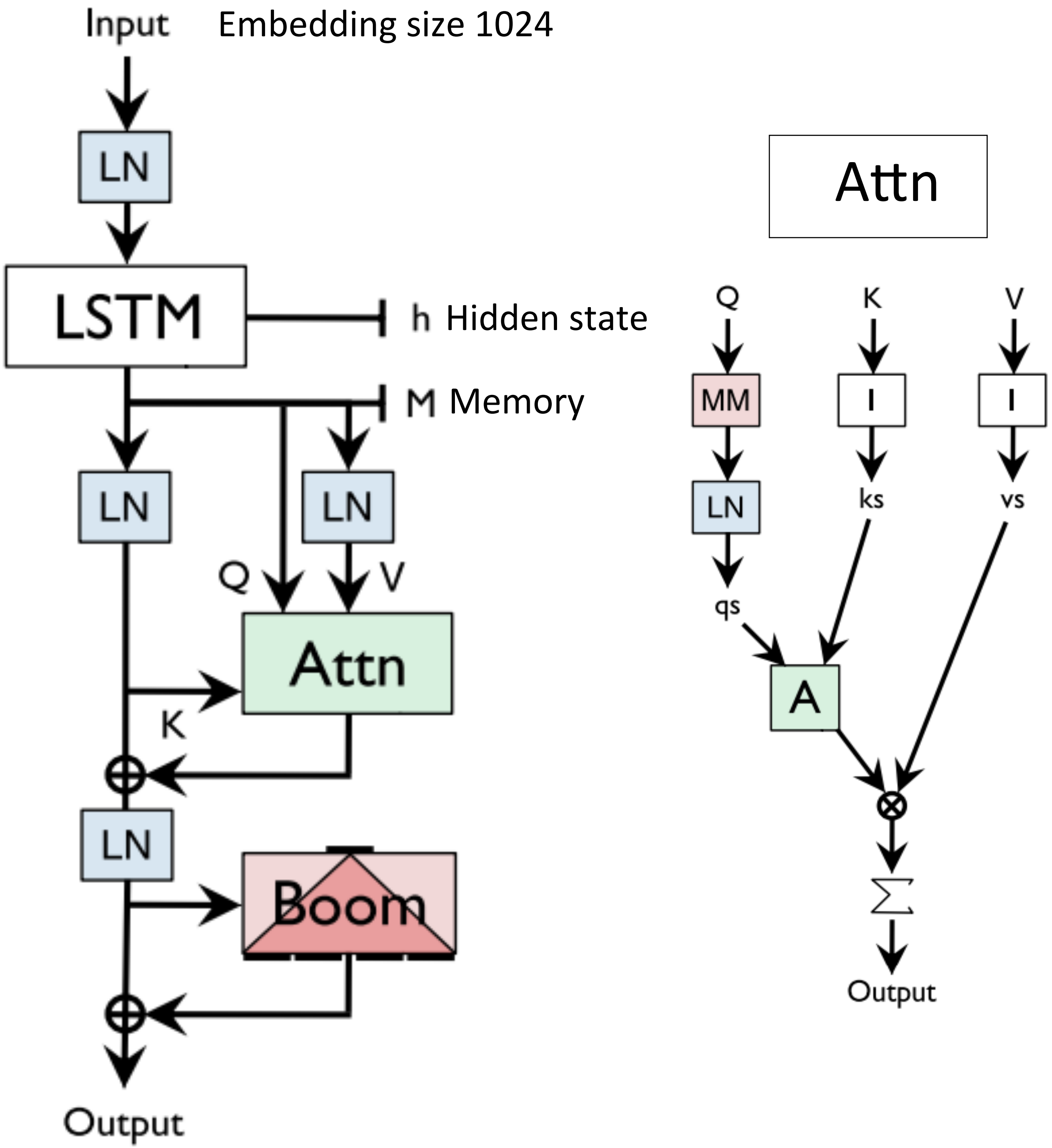
Model	Heads	Valid	Test	Params
Large RHN (Zilly et al., 2016)	0	–	1.27	46M
3 layer AWD-LSTM (Merity et al., 2018b)	0	–	1.232	47M
T12 (12 layer) (Al-Rfou et al., 2019)	24	–	1.11	44M
LSTM (Melis et al., 2019)	0	1.182	1.195	48M
Mogrifier LSTM (Melis et al., 2019)	0	1.135	1.146	48M
4 layer SHA-LSTM (h = 1024, no attention head)	0	1.312	1.330	51M
4 layer SHA-LSTM (h = 1024, single attention head)	1	1.100	1.076	52M
4 layer SHA-LSTM (h = 1024, attention head per layer)	4	1.096	1.068	54M
T64 (64 layer) (Al-Rfou et al., 2019)	128	–	1.06	235M
Transformer-XL (12 layer) (Dai et al., 2019)	160	–	1.06	41M
Transformer-XL (18 layer) (Dai et al., 2019)	160	–	1.03	88M
Adaptive Transformer (12 layer) (Sukhbaatar et al., 2019)	96	1.04	1.02	39M
Sparse Transformer (30 layer) (Child et al., 2019)	240	–	0.99	95M

Attention Head at Second last layer.

Table 1. Bits Per Character (BPC) on `enwik8`. The single attention SHA-LSTM has an attention head on the second last layer and had batch size 16 due to lower memory use. Directly comparing the head count for LSTM models and Transformer models obviously doesn't make sense but neither does comparing zero-headed LSTMs against bajillion headed models and then declaring an entire species dead. The hyper-parameters for the fully headed SHA-LSTM were used for the other SHA-LSTM experiments with zero tuning.

SHA RNN Architecture

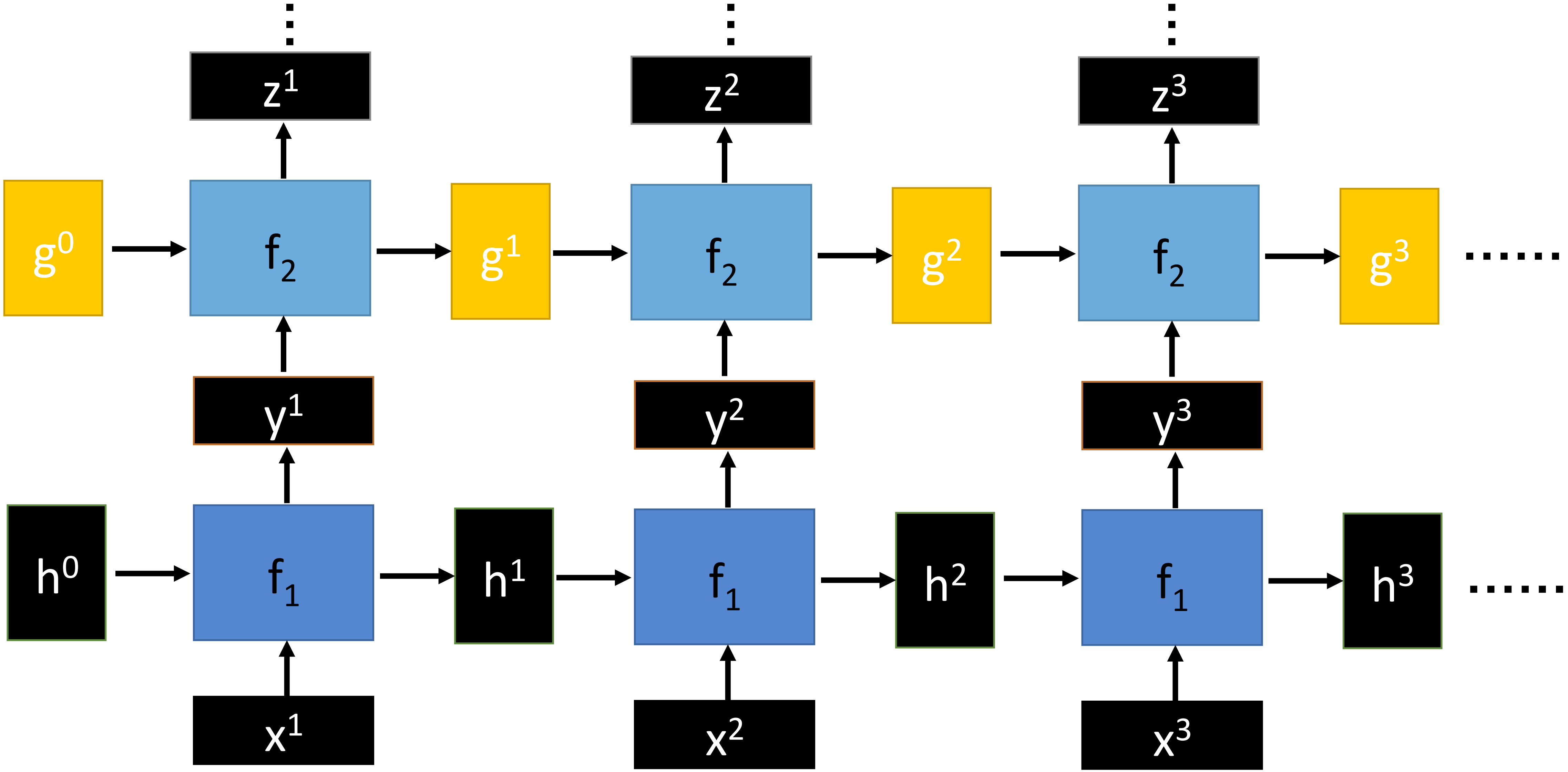
1. The Boom layer is a fully connected NN that maps v of size H to u of size NxH, then break u into N=4 vectors and sum them together to produce w of size H (1024).
2. The Attention mechanism is very similar to Attention we learned in Lecture 2, except softmax is replaced by sigmoid.



Deep RNN

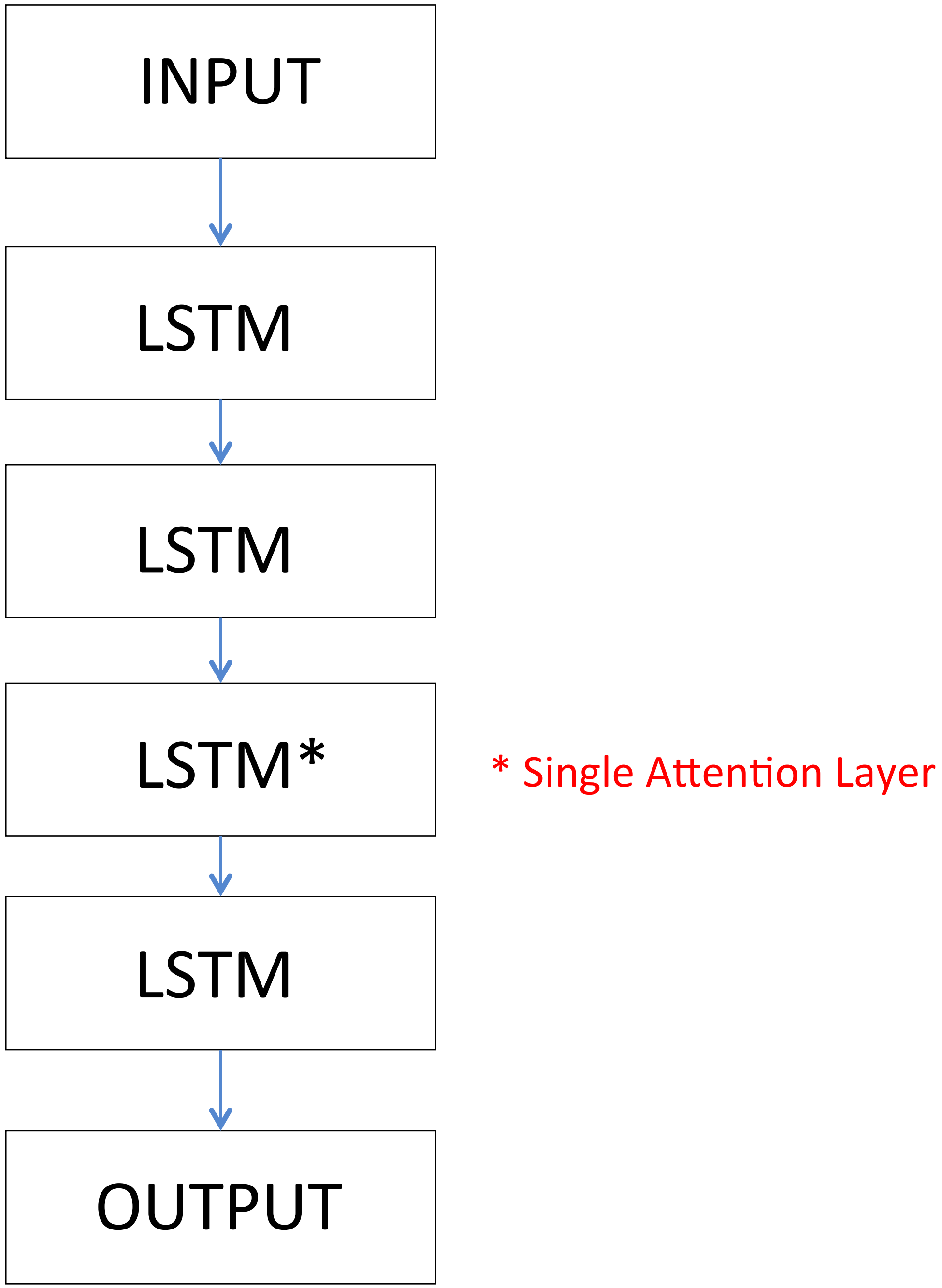
$h', y = f_1(h, x),\ g', z = f_2(g, y)$

…



4 layer SHA RNN

1. Each layer gives the next layer sequential output.
2. Name: Stacked LSTM or Deep LSTM.



Literature & Resources

Li and Vitanyi, An introduction to Kolmogorov complexity and its applications, 2019, 4th edition.

Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, R. Soricut, ALBERT: a lite BERT for self-supervised learning of language representations. 2019

S. Merity, Single hearded Attention RNN: Stop thinking with your head, 2020

Can somebody present Sukhbaatar et al 2019 Adaptive Transformer?

Can somebody present the Sparse Transformer by Child et, 2019.

https://twimlai.com/twiml-talk-325-single-headed-attention-rnn-stop-thinking-with-your-head-with-stephen-merity/