Applied Text Analytics (F20/F21AA)

Heriot Watt University , Dubai Campus

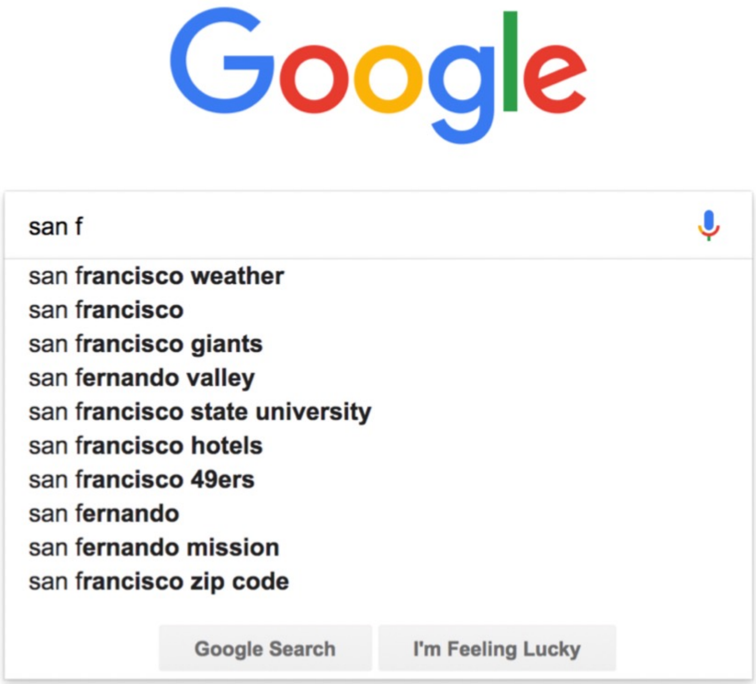
Dr. Radu Mihailescu

Associate Professor

RNN as a language model

Language Modeling

Language modelling is the task of predicting what word comes next.



Language Modeling

- Goal: compute the probability of a sentence or sequence of words:

$$P(W) = P(w_1,w_2,w_3,w_4,w_5\ldots w_n)$$

- Related task: probability of an upcoming word:

$P(w_5|w_1,w_2,w_3,w_4)$ I have a cat and it likes to drink ………

- A model that computes either of these:

$P(W)$ or $P(w_n|w_1,w_2\ldots w_{n-1})$ is called a **language model**.

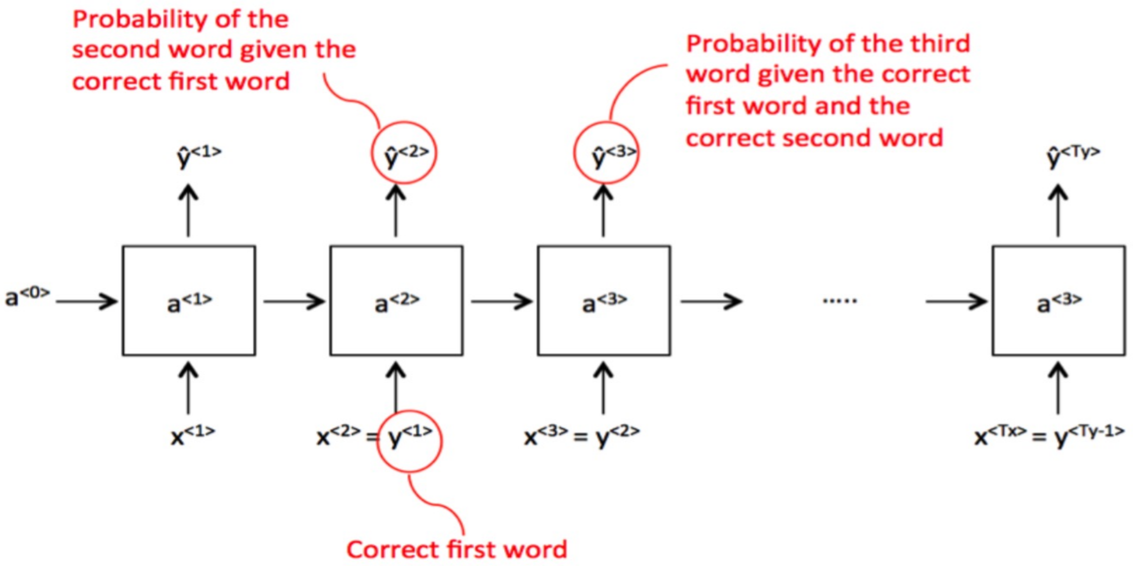
NLP Problems

Predicts the correct sentence, based on the probability of one sentence versus the other.

- Machine Translation:
  - P(**high** winds tonight) > P(**large** winds tonight)
- Spell Correction
  - The office is about fifteen **minuets** from my house
    - P(about fifteen **minutes** from) > P(about fifteen **minuets** from)
- Speech Recognition
  - P(I saw a van) >> P(eyes awe of an)

- \+ Summarization, question-answering, etc., etc.!!

Training RNN



$$\mathcal{L}(\hat{y}^{<t>}, y^{<t>}) = -\sum_i y_i^{<t>} \log \hat{y}_i^{<t>}$$

$$\mathcal{L} = \sum_t \mathcal{L}^{<t>}(\hat{y}^{<t>}, y^{<t>})$$

Using a large corpus of text.

- Tokenize the input sentences.
- Replace unknown word with <UNK> token.
- Model sentence end by the token token <EOS>.

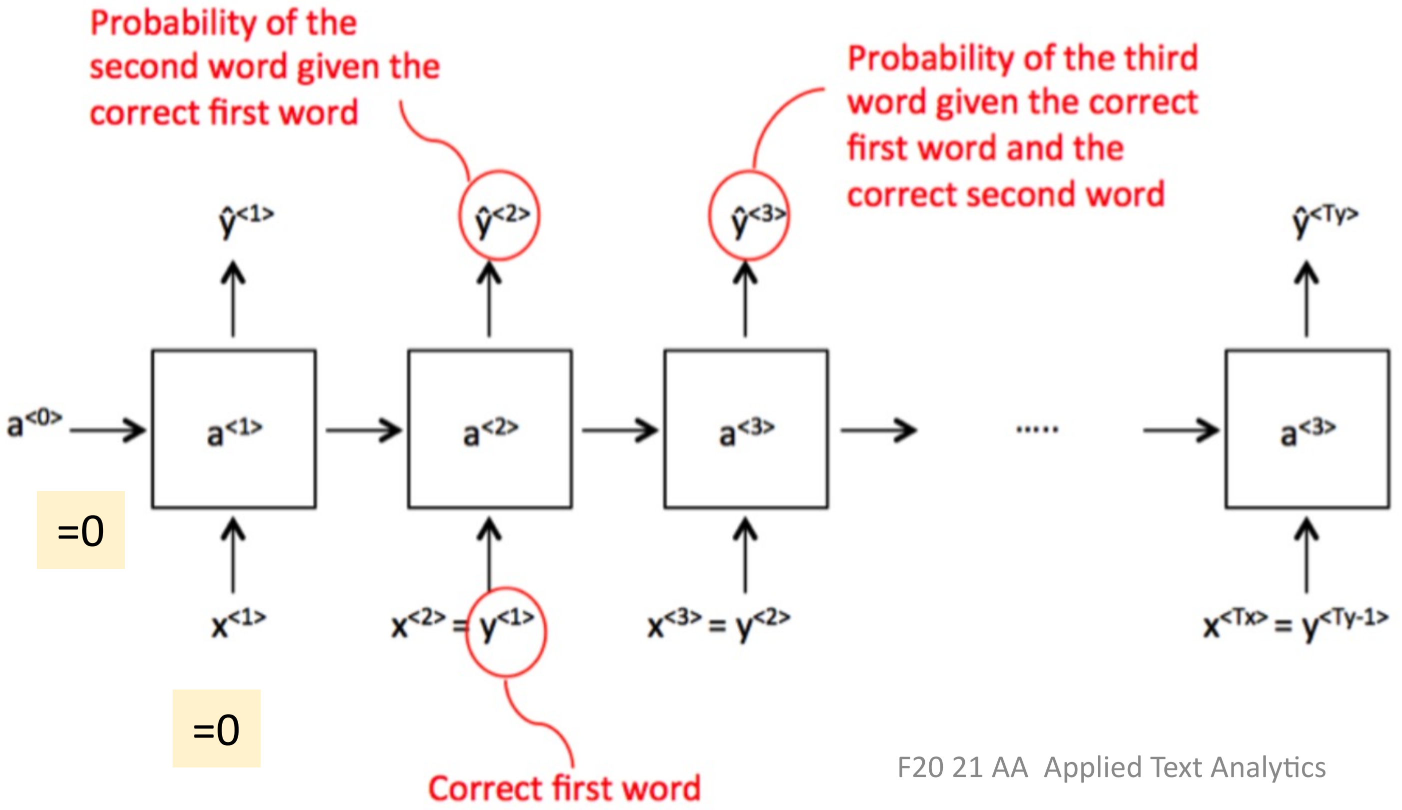
- Map each word to a one-hot vector of indices.
- Train RNN to model the chance of these different sequences.

RNN for Language Modelling

Given an input sentence: I have a dog and it likes to play

$y^{<1>}$ $y^{<2>}$ $y^{<3>}$ $y^{<4>}$ $y^{<5>}$ $y^{<6>}$ $y^{<7>}$ $y^{<8>}$ $y^{<9>}$

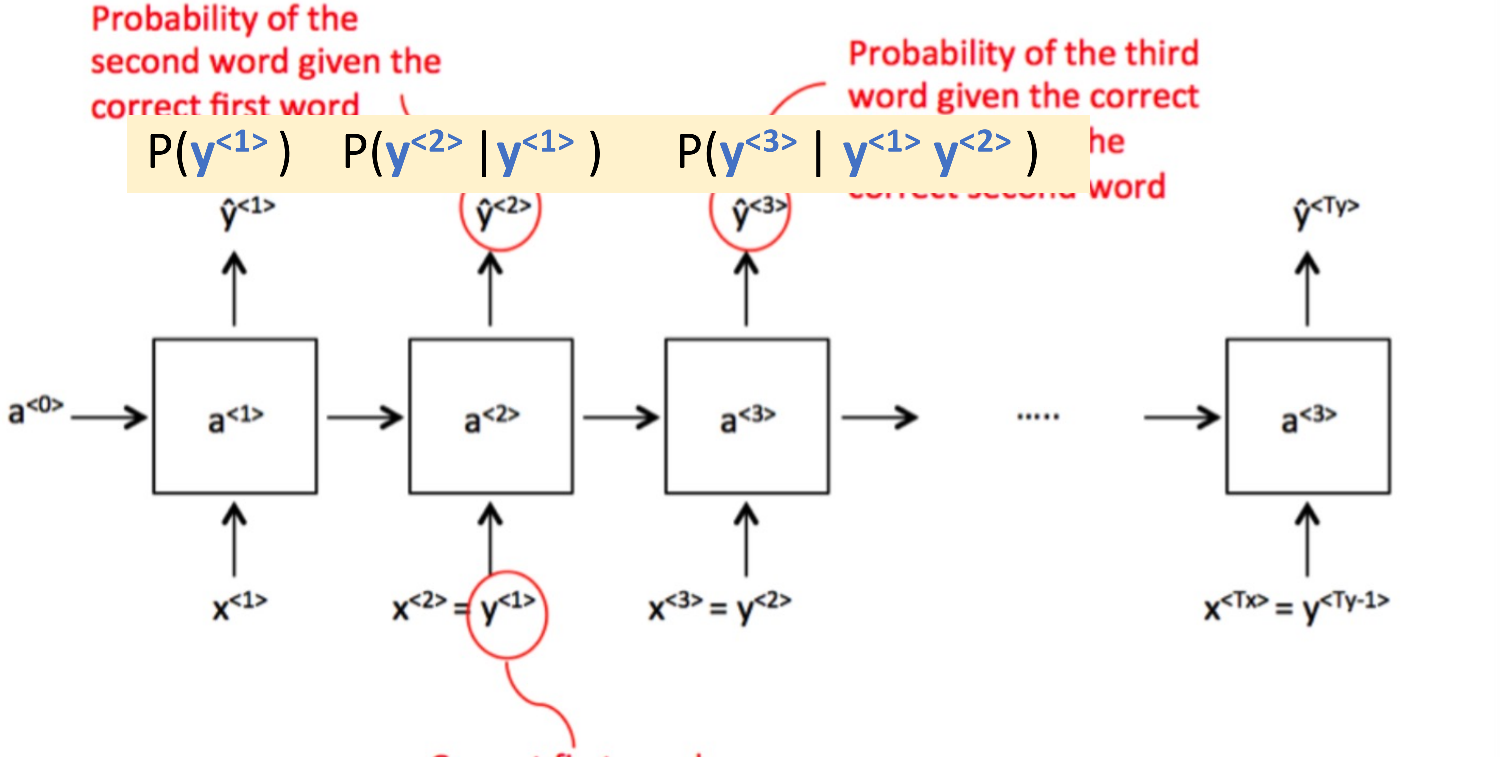
Calculate $P(y^{<1>}, y^{<2>}, y^{<3>}, y^{<4>}, y^{<5>}, y^{<6>}, y^{<7>}, y^{<8>}, y^{<9>})$



$x^{<t>} = y^{<t-1>}$

RNN for Language Modeling (after training you can calculate sentence probability)

$P(y^{<1>})$ $P(y^{<2>} \mid y^{<1>})$ $P(y^{<3>} \mid y^{<1>}\ y^{<2>})$



The Chain Rule in General

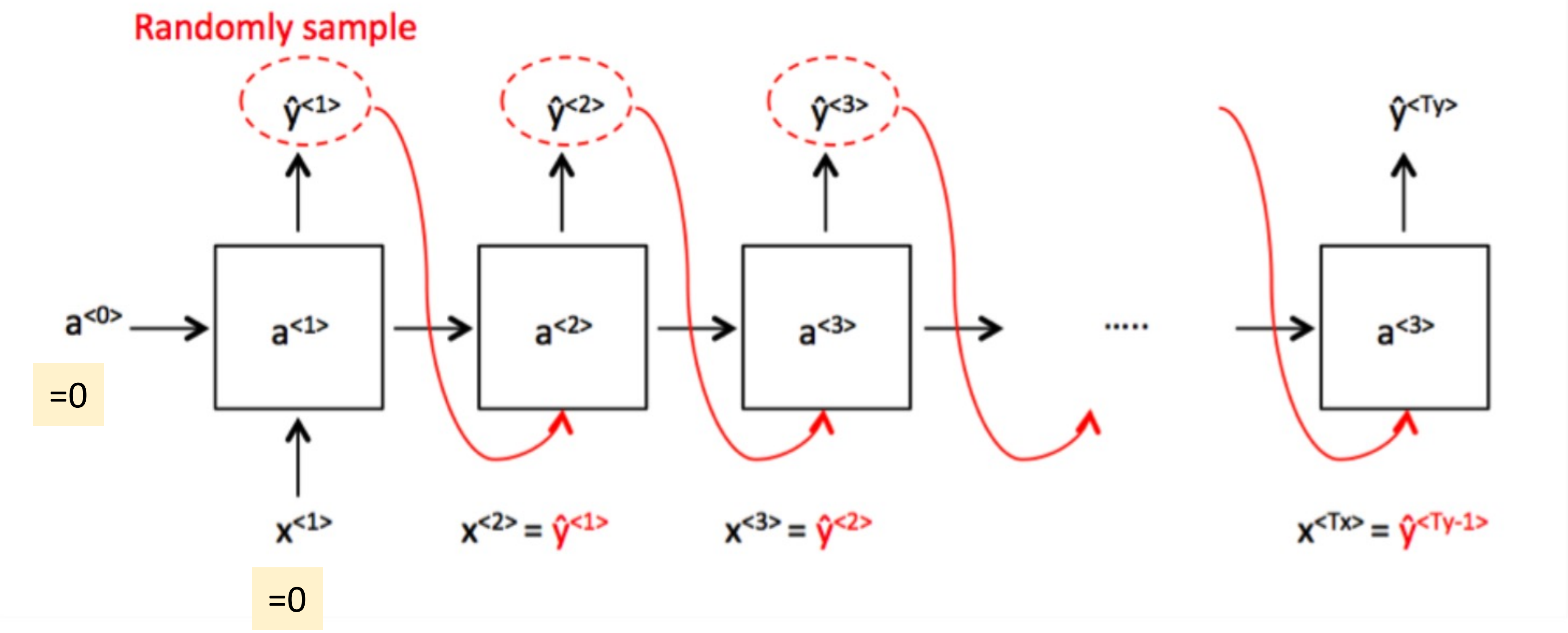
$P(y_1,y_2,y_3,\ldots,y_n) =$

$P(y_1)P(y_2 \mid y_1)P(y_3 \mid y_1,y_2)\ldots P(y_n \mid y_1,\ldots,y_{n-1})$

$$P(y^{<1>}, y^{<2>}, y^{<3>}) = P(y^{<1>}) * P(y^{<2>} \mid y^{<1>}) * P(y^{<3>} \mid y^{<1>}, y^{<2>})$$

Sequence Generation (Sampling from language model)

According to probability dist. generated from $\hat{y}^{<t>}$



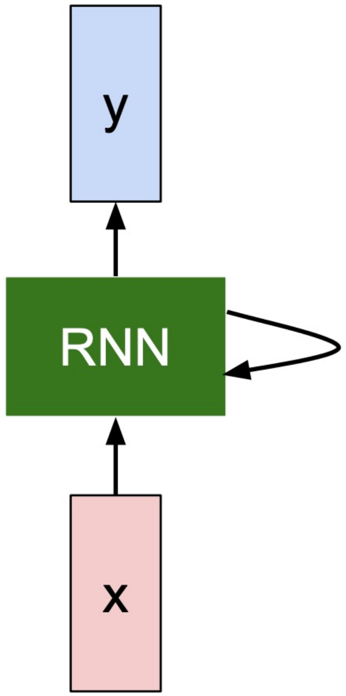
Sequence Generation (Sampling from language model)

THE SONNETS

by William Shakespeare

From fairest creatures we desire increase,
That thereby beauty's rose might never die,
But as the riper should by time decease,
His tender heir might bear his memory:
But thou, contracted to thine own bright eyes,
Feed'st thy light's flame with self-substantial fuel,
Making a famine where abundance lies,
Thyself thy foe, to thy sweet self too cruel:
Thou that art now the world's fresh ornament,
And only herald to the gaudy spring,
Within thine own bud buriest thy content,
And tender churl mak'st waste in niggarding:
    Pity the world, or else this glutton be,
    To eat the world's due, by the grave and thee.

When forty winters shall besiege thy brow,
And dig deep trenches in thy beauty's field,
Thy youth's proud livery so gazed on now,
Will be a tatter'd weed of small worth held:
Then being asked, where all thy beauty lies,
Where all the treasure of thy lusty days;
To say, within thine own deep sunken eyes,
Were an all-eating shame, and thriftless praise.
How much more praise deserv'd thy beauty's use,
If thou couldst answer 'This fair child of mine
Shall sum my count, and make my old excuse,'
Proving his beauty by succession thine!
    This were to be new made when thou art old,
    And see thy blood warm when thou feel'st it cold.



Sequence Generation

```
tyntd-iafhatawiaoihrdemot  lytdws  e ,tfti, astai f ogoh eoase rrranbyne 'nhthnee e
plia tklrgd t o idoe ns,smtt   h ne etie h,hregtrs nigtike,aoaenns lng
```

train more

```
"Tmont thithey" fomesscerliund
Keushey. Thom here
sheulke, anmerenith ol sivh I lalterthend Bleipile shuwy fil on aseterlome
coaniogennc Phe lism thond hon at. MeiDimorotion in ther thize."
```

train more

```
Aftair fall unsuch that the hall for Prince Velzonski's that me of
her hearly, and behs to so arwage fiving were to it beloge, pavu say falling misfort
how, and Gogition is so overelical and ofter.
```

train more

```
"Why do what that day," replied Natasha, and wishing to himself the fact the
princess, Princess Mary was easier, fed in had oftened him.
Pierre aking his soul came to the packs and drove up his father-in-law women.
```

Sequence Generation

```c
static void do_command(struct seq_file *m, void *v)
{
  int column = 32 << (cmd[2] & 0x80);
  if (state)
    cmd = (int)(int_state ^ (in_8(&ch->ch_flags) & Cmd) ? 2 : 1);
  else
    seq = 1;
  for (i = 0; i < 16; i++) {
    if (k & (1 << 1))
      pipe = (in_use & UMXTHREAD_UNCCA) +
        ((count & 0x00000000fffffff8) & 0x000000f) << 8;
    if (count == 0)
      sub(pid, ppc_md.kexec_handle, 0x20000000);
    pipe_set_bytes(i, 0);
  }
  /* Free our user pages pointer to place camera if all dash */
  subsystem_info = &of_changes[PAGE_SIZE];
  rek_controls(offset, idx, &soffset);
  /* Now we want to deliberately put it to device */
  control_check_polarity(&context, val, 0);
  for (i = 0; i < COUNTER; i++)
    seq_puts(s, "policy ");
}
```

Generated C code

```c
#include <asm/io.h>
#include <asm/prom.h>
#include <asm/e820.h>
#include <asm/system_info.h>
#include <asm/setew.h>
#include <asm/pgproto.h>

#define REG_PG    vesa_slot_addr_pack
#define PFM_NOCOMP  AFSR(0, load)
#define STACK_DDR(type)     (func)

#define SWAP_ALLOCATE(nr)     (e)
#define emulate_sigs()  arch_get_unaligned_child()
#define access_rw(TST)  asm volatile("movd %%esp, %0, %3" : : "r" (0));   \
  if (__type & DO_READ)

static void stat_PC_SEC __read_mostly offsetof(struct seq_argsqueue, \
          pC>[1]);

static void
os_prefix(unsigned long sys)
{
#ifdef CONFIG_PREEMPT
  PUT_PARAM_RAID(2, sel) = get_state_state();
  set_pid_sum((unsigned long)state, current_state_str(),
           (unsigned long)-1->lr_full; low;
}
```

Character level RNNs

- Instead of words use character level language model
- Vocabulary [a, b,…...z,………0,…9,………..A,………..Z]

Advantages:
Will never encounter an unknown word.

Disadvantages:
computational cost to train such a network as they deal with much longer sequences
not so good at capturing long-range dependencies, meaning how the earlier part of the sentence affects the later part.

Sequence Generation (Character level RNNs)

Examples: **Deep Learning for Natural Language Processing (NLP) – using RNNs & CNNs** [Kdnuggets News Feb 2019]

You can Subscribe to this !!!

Links to articles & code

- Generate fake Linux functions
- Composing Jazz Music
- Create a new Super Mario level

(Search for updates …..)

Sequence Generation (Character level RNNs)

Examples: **Deep Learning for Natural Language Processing (NLP) – using RNNs & CNNs** [Kdnuggets News Feb 2019]

You can Subscribe to this !!!

Links to articles & code

- Generate fake Linux functions
- Composing Jazz Music
- Create a new Super Mario level

(Search for updates …..)

Beethoven's unfinished Tenth Symphony completed by artificial intelligence…. September 21

Vanishing/Exploding gradients

- A general problem in deep NN
- Derivative decreases/grows exponentially as a function of the layer
- Vanishing gradient: (more difficult)
  - network has a difficult time propagating back to affect the weights of earlier layers.
  - Basic RNNs are not good at capturing these long-term dependencies.
  - Use GRU (Gated recurrent NN)
- Exploding gradients:
  - Can be solved by “gradient clipping”

Resources

Textbooks:

Speech and Language Processing (3rd ed. draft) -Chapter 9
Dan Jurafsky and James H. Martin

https://web.stanford.edu/~jurafsky/slp3/

Hands-On Machine Learning with Scikit-Learn and TensorFlow by Aurélien Géron (O'Reilly).

Chapter 15 Ebook

Video Lectures Stanford NLP & DL lecture 6 (RNN and Language models)

- Lecture 3(NN) Lecture 4 (Backpropagation)

Interesting Reads:

Sequence Models (Based on Andrew NG DL course)

Deep Learning for NLP (KDnuggets)

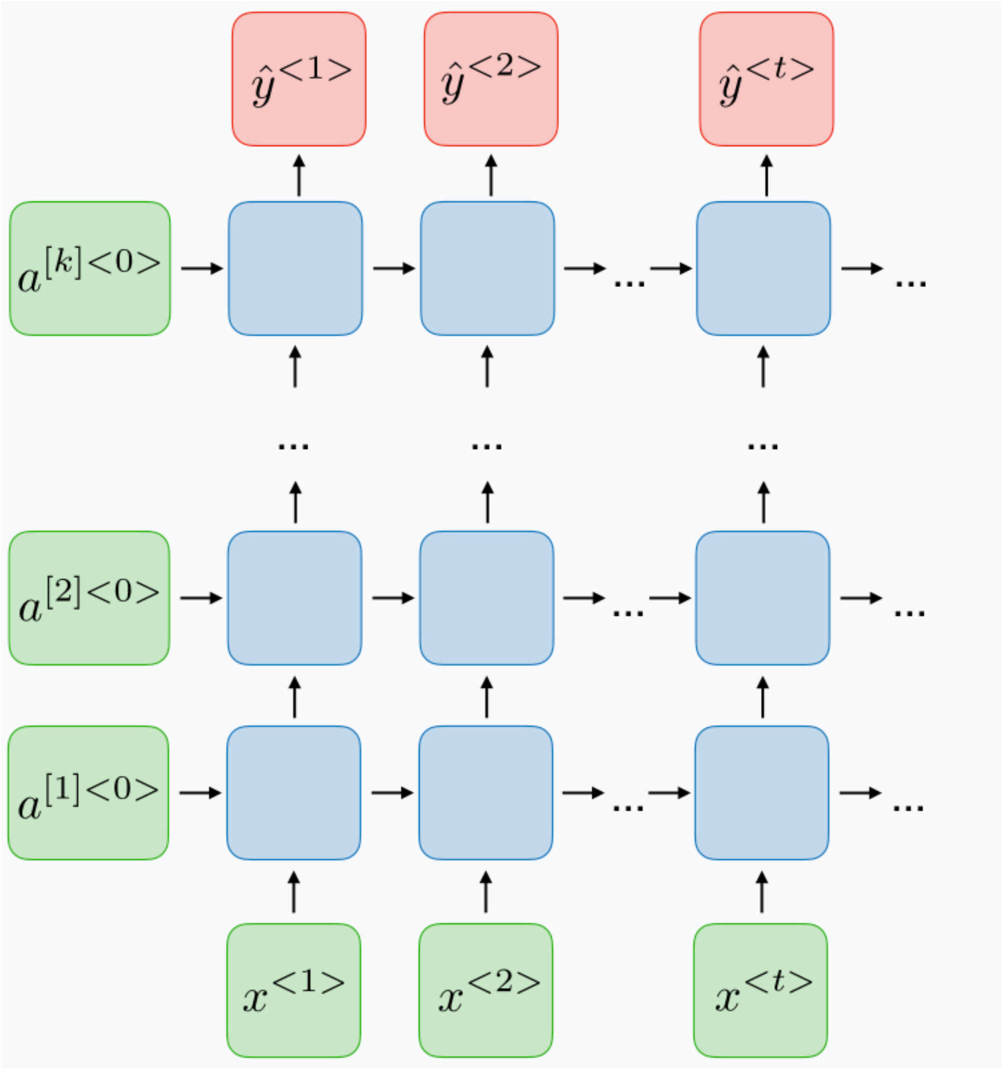
Next

- We discussed RNNs as sequence models
  - problem they face vanishing/exploding gradients
- Solutions widely used: LSTM and GRU cells.
- Lab week 9: We will show how to implement RNNs using TensorFlow (lab) & use for text classification
- Next lecture: RNN more fancy models & applications

Extra for this week’s lab

Deep RNN

- Multi-layer RNNs are also called *stacked RNNs*.
- These networks can learn complex functions
- Generally no more than 3-5 layers because of the temporal dimension
- The lower RNNs should compute lower-level features and the higher RNNs should compute higher-level features.

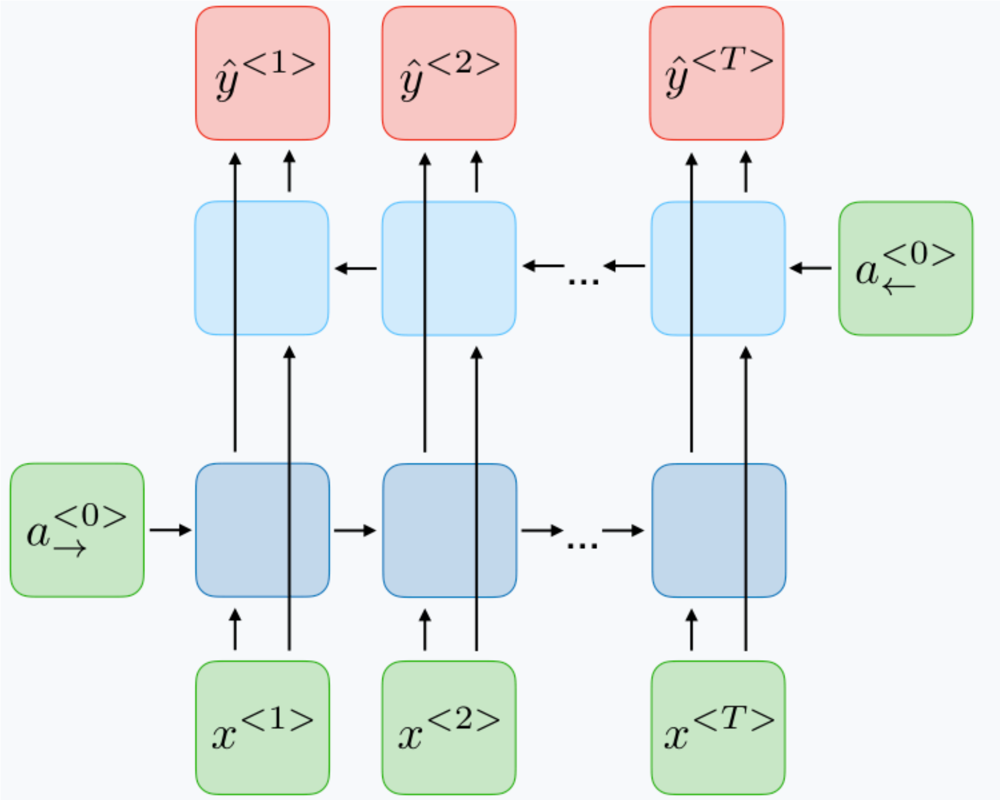


Bi-Directional RNNs (BRNN)

- Bidirectional Recurrent Neural Networks (BRNN) allow hidden states to receive information from both past and future states. (early & late sequence)

- Could consider new hidden state as a concatenation between forward and backward states
- Can also work for GRU and LSTM (besides Vanilla RNNs)

- Can only work if the entire sequence is present before predicting (doesn’t work for real time speech application)

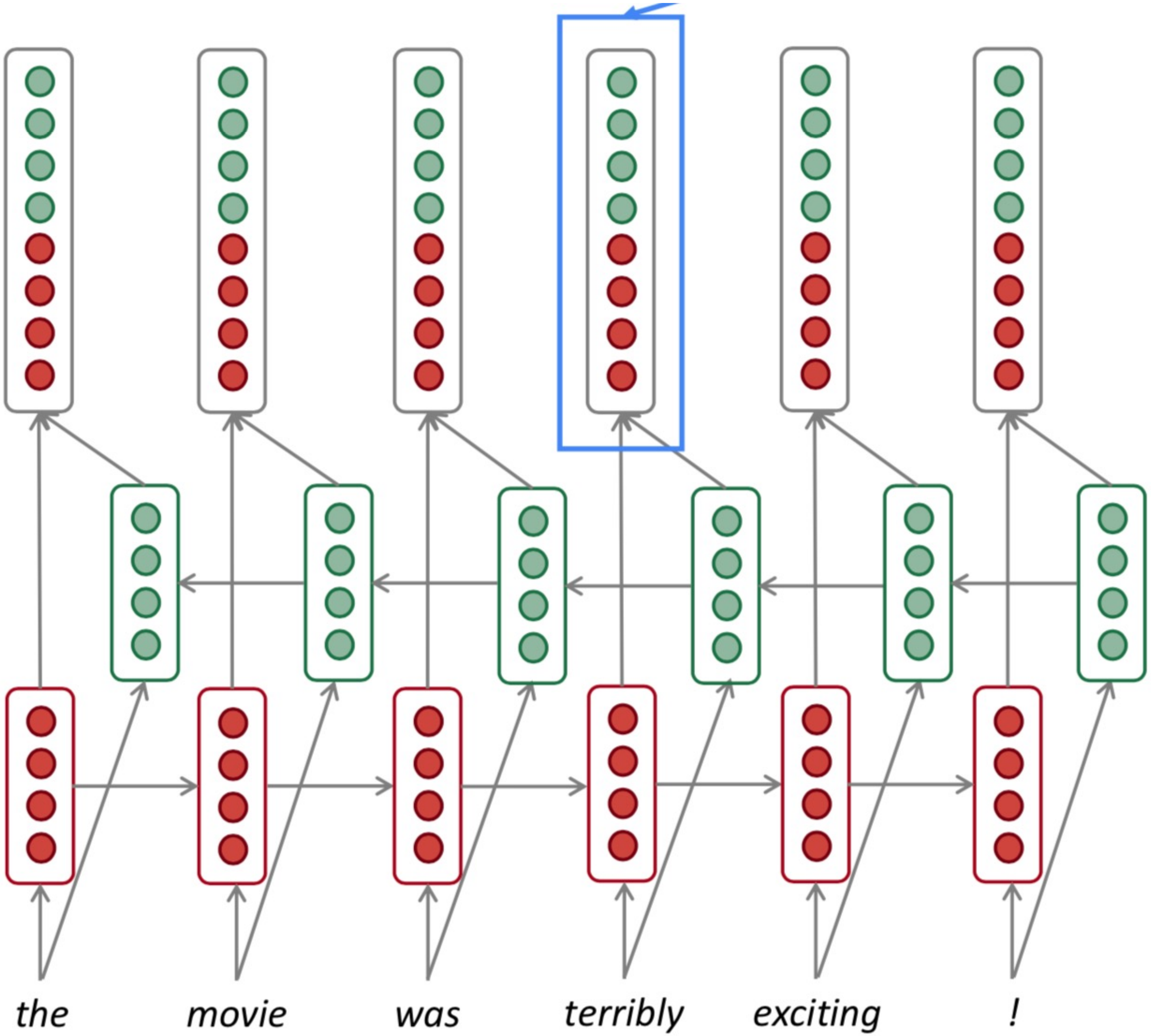


BRNN

- Concatenating forward & backward states

Backward RNN

Forward RNN



Bi- Directional RNNs (BRNN)

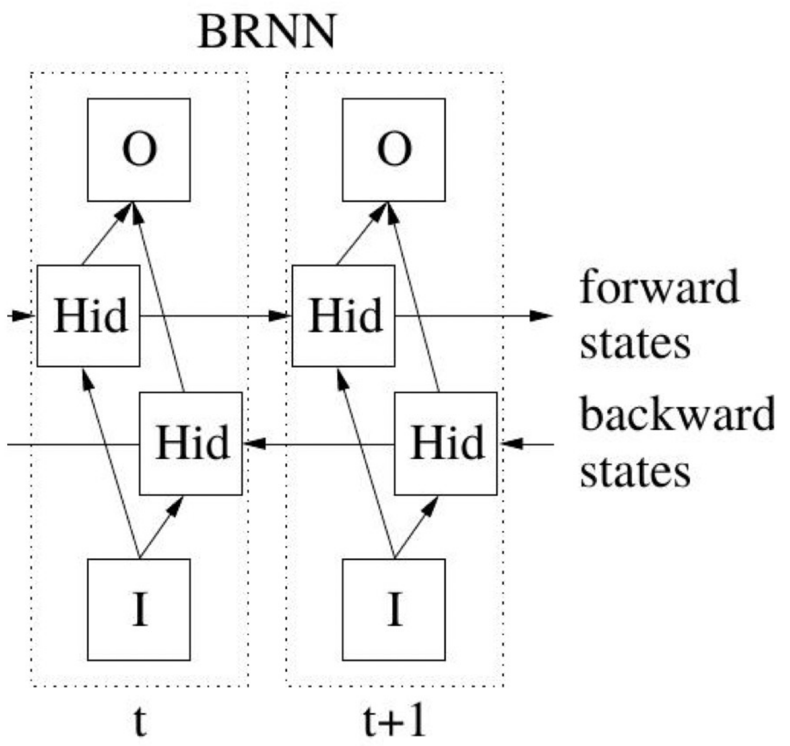
for $t = 1$ to T **do**
  Do forward pass for the forward hidden layer, storing activations at each timestep
for $t = T$ to 1 **do**
  Do forward pass for the backward hidden layer, storing activations at each timestep
for $t = 1$ to T **do**
  Do forward pass for the output layer, using the stored activations from both hidden layers

Algorithm 3.1: BRNN Forward Pass

Similarly, the backward pass proceeds as for a standard RNN trained with BPTT, except that all the output layer δ terms are calculated first, then fed back to the two hidden layers in opposite directions:

for $t = T$ to 1 **do**
  Do BPTT backward pass for the output layer only, storing δ terms at each timestep
for $t = T$ to 1 **do**
  Do BPTT backward pass for the forward hidden layer, using the stored δ terms from the output layer
for $t = 1$ to T **do**
  Do BPTT backward pass for the backward hidden layer, using the stored δ terms from the output layer

Algorithm 3.2: BRNN Backward Pass



Thank you!