Word Embedding & RNN Applications

Representing words and one-hot encoding

X: Rama Conquered Ravana to install the virtue of dharma

A	1	Rama	Ravana
:		0	0
:		0	0
Conquered	329	0	0
:		0	0
:		0	0
Install	4521	:	:
:		:	:
:		:	:
Rama	7689	1 -7689	:
:		:	1-7900
Ravana	7900	:	:
:		0	0
ZZZ	10000	0	0

Featurized Representation: Word Embeddings

	Man **5391**	**Woman** **9853**	**King** **4914**	**Queen** **7157**	**Apple** **456**	**Orange** **6257**
Gender	1	1	0.95	0.97	0.00	0.01
Royal	0.01	0.02	0.93	0.95	-0.01	0
Age	0.03	0.02	0.7	0.69	0.03	0.02
Food	0.04	0.01	0.02	0.01	0.95	0.97
Size						
Cost						
Alive						

If we have 300 such properties and 10000 Words then it will be a 300x10000 Matrix and is denoted as Embedding Matrix (E) and O_{man} is One hot Vector for Man Word and e_{man} can be embedding vector for Man word.

I Want A glass of orange __Juice
I want a glass of Apple

Transfer Learning and Word Embeddings

- Learn Word Embeddings from Large Text Corpus (1-100 Billion Words)

- Pre-trained embeddings are available online

- Transfer Embedding to a new task with smaller training set

- Continue to finetune word embeddings with new data

Analogies using Word Vectors

As Man-> Woman King->?

As INR->India Dollar->?

As Delhi->India Kathmandu->?

e_{man}-e_{woman} ≈ e_{king}-e_{w}

As Tall->taller Big->?

As Man->Woman Boy->?

……

Find a word w : Maximize similarity(e_{w} , e_{king}-e_{man}+e_{woman})

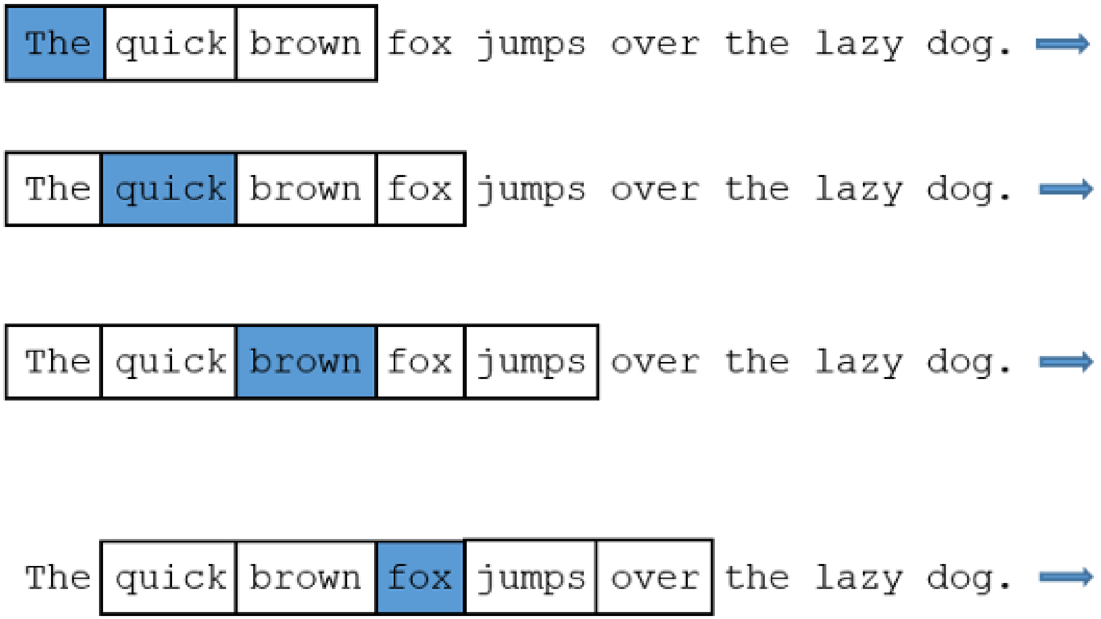
Cosine similarity

Sim(u,v)= $u^{T}v$ / ||u|| ||v||

Skip Gram Model

Source Text

Training Samples



The quick brown fox jumps over the lazy dog. ⟹

(the, quick)
(the, brown)

The quick brown fox jumps over the lazy dog. ⟹

(quick, the)
(quick, brown)
(quick, fox)

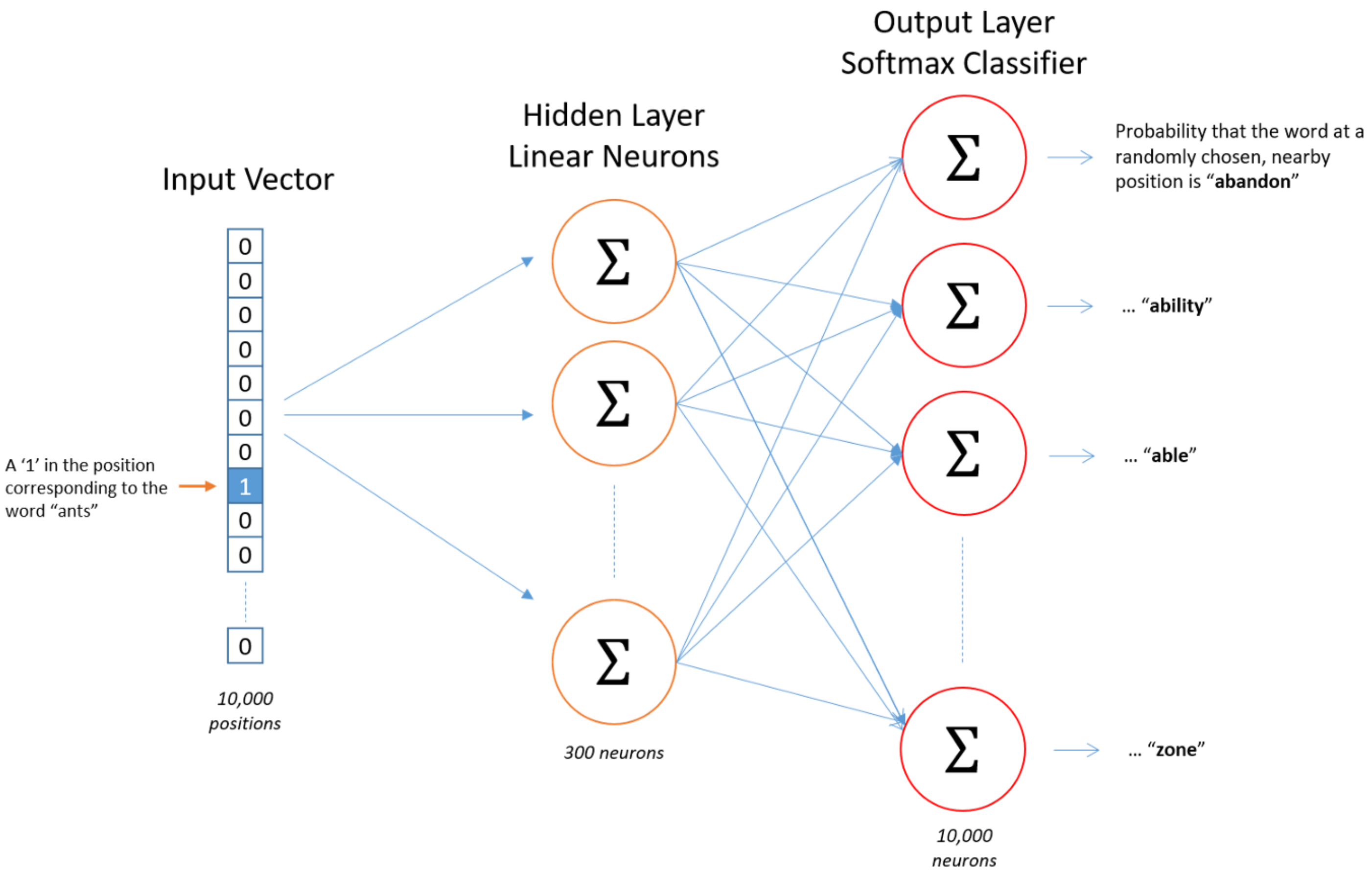
The quick brown fox jumps over the lazy dog. ⟹

(brown, the)
(brown, quick)
(brown, fox)
(brown, jumps)

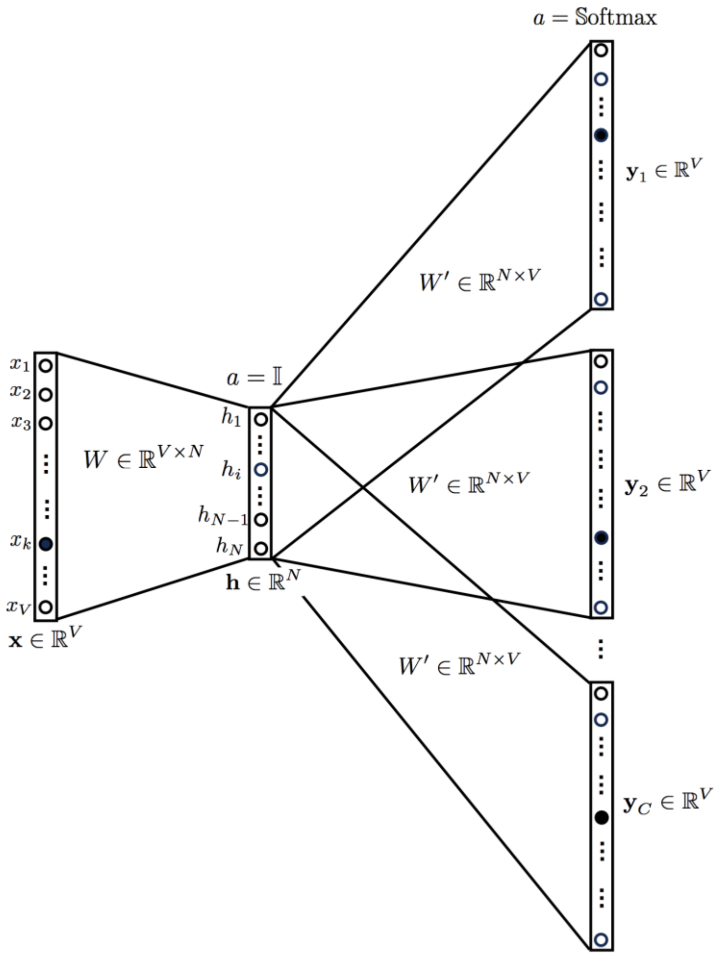
The quick brown fox jumps over the lazy dog. ⟹

(fox, quick)
(fox, brown)
(fox, jumps)
(fox, over)

Skip Gram Model



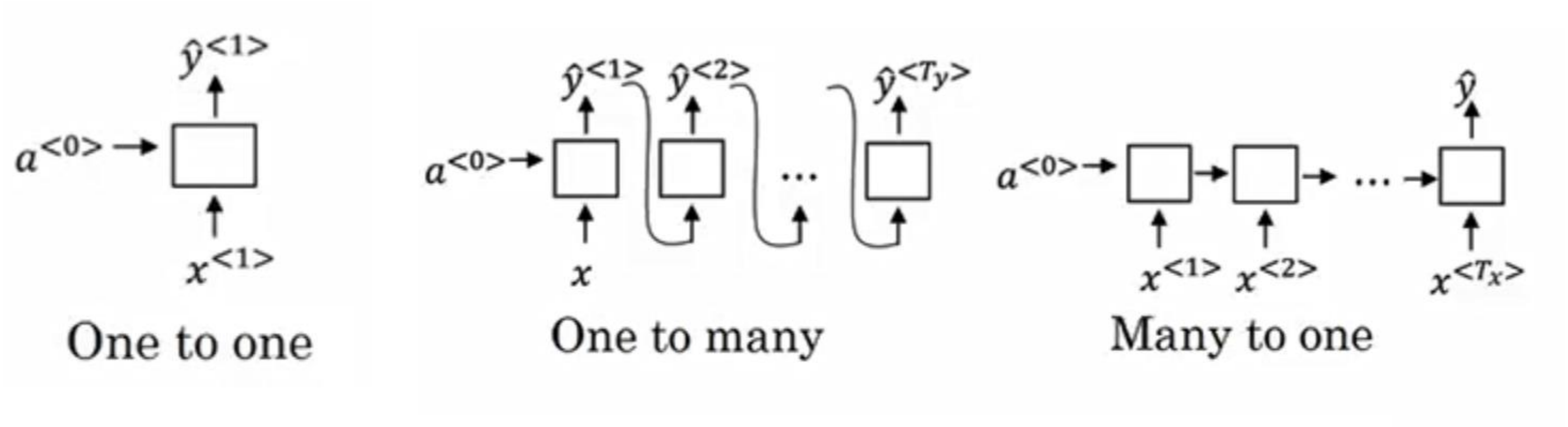
Skip Gram Model


$a = \text{Softmax}$
$\mathbf{y}_1 \in \mathbb{R}^V$
$W' \in \mathbb{R}^{N \times V}$
x_1
x_2
x_3
x_k
x_V
$\mathbf{x} \in \mathbb{R}^V$
$W \in \mathbb{R}^{V \times N}$
$a = \mathbb{I}$
h_1
h_i
h_{N-1}
h_N
$\mathbf{h} \in \mathbb{R}^N$
$W' \in \mathbb{R}^{N \times V}$
$\mathbf{y}_2 \in \mathbb{R}^V$
$W' \in \mathbb{R}^{N \times V}$
$\mathbf{y}_C \in \mathbb{R}^V$

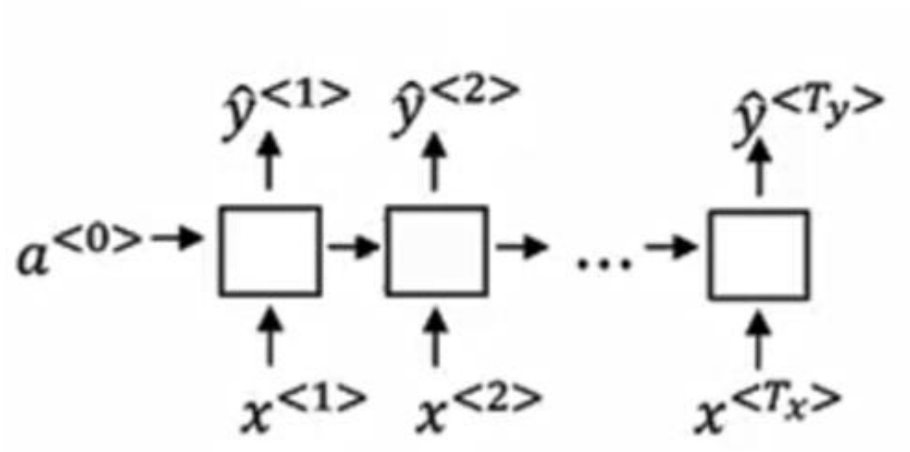
Removing Biases in NLP

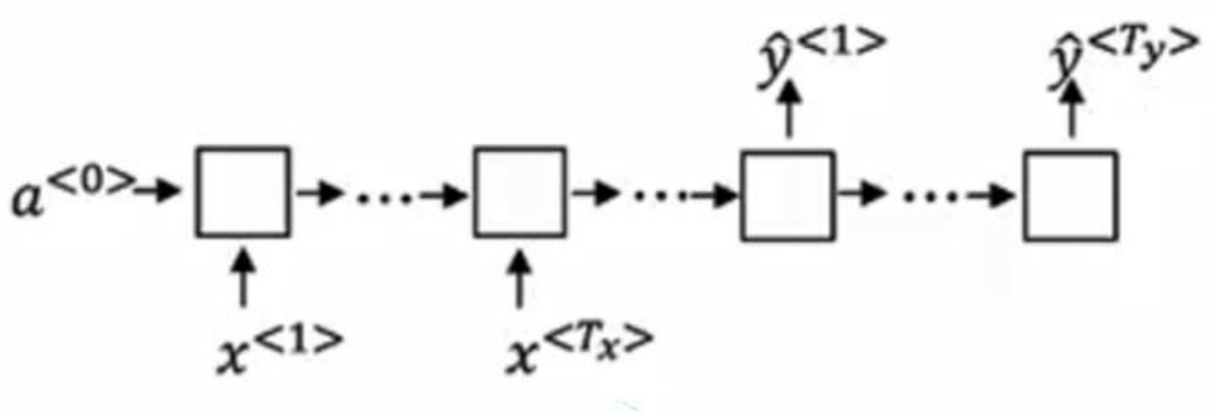
- It related to Gender and ethnicity biases and we need to be very careful about this
- Man: Computer_Programmer as Women Homemaker
- Father: Doctor Mother: Nurse

- Biases will be picked from the text it has been trained upon
- First step is to identify Bias Direction e.g. male to female
- Next is to Neutralize the bias for all the non-definitional word for example Father, Mother, He, She are definitional word for Gender and should not get changes due to this. However, Non-definitional word like soldier, doctor, Manager, Programmer etc should be neutralized for bias
- Last step is to equalize pairs like niece, nephew; grandmother, grandfather and they should be equidistant from words like babysitter etc.



Different Type of RNN Architectures


$\hat{y}^{<1>}$ $\hat{y}^{<2>}$ $\hat{y}^{<T_y>}$
$a^{<0>}$
$x^{<1>}$ $x^{<2>}$ $x^{<T_x>}$



Many to Many

Many to Many

Word Level Language Model

Train your Language model on a large data.

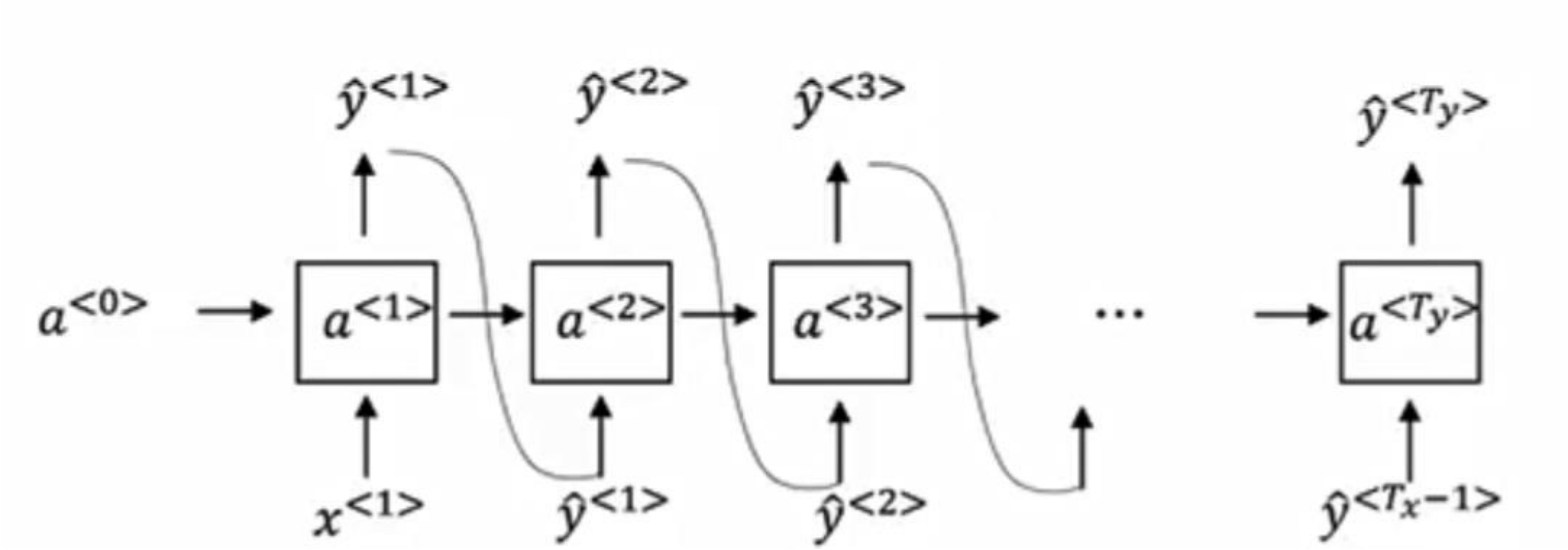
Then You can build on it different kinds of NLP applications as discussed.

Also, You will be able to generate new sentences or paragraphs etc as per the requirements of your application.

Difference between Word Level and Character Level Language Model

- Word Level Language models are more common due to their better performance as of now.
- Word level language models have <EOS> and <UNK> also as tokens in the corpus.
- Character level corpus is very small as compared to word level corpus
- In most cases word level corpus are of size 30-50k but in some cases can be upto 1 million
- In character level language model it becomes hard to predict and relate the relationship between far off characters as the distance becomes very large as compared to word language models.

Word level and character level language model



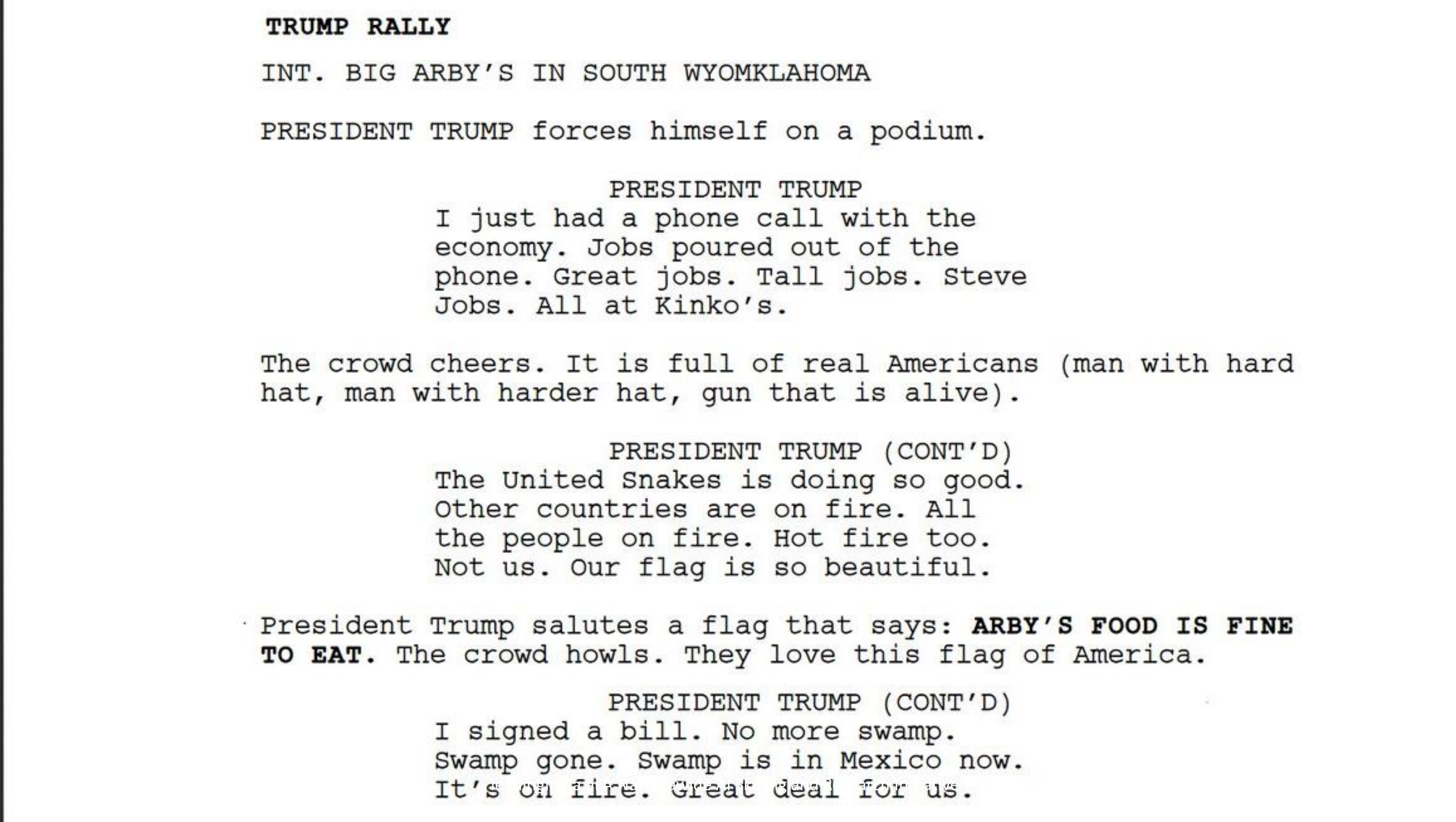
Sampling a novel sequence

Once you have a trained model on a corpus you can also have a RNN that can sample new sequences for you.

In that case you initialize with a zero and your first output gives a probability in terms of softmax function of the size of the no of categories equal to the size of your corpus.

You Choose a random word as the first output and then that word acts as the input for the second input and so on.

If you get a <UNK> then you can reject that token and continue with the next guess. It can go on until you get a <EOS> token.



TRUMP RALLY

INT. BIG ARBY’S IN SOUTH WYOMKLAHOMA

PRESIDENT TRUMP forces himself on a podium.

PRESIDENT TRUMP
I just had a phone call with the economy. Jobs poured out of the phone. Great jobs. Tall jobs. Steve Jobs. All at Kinko’s.

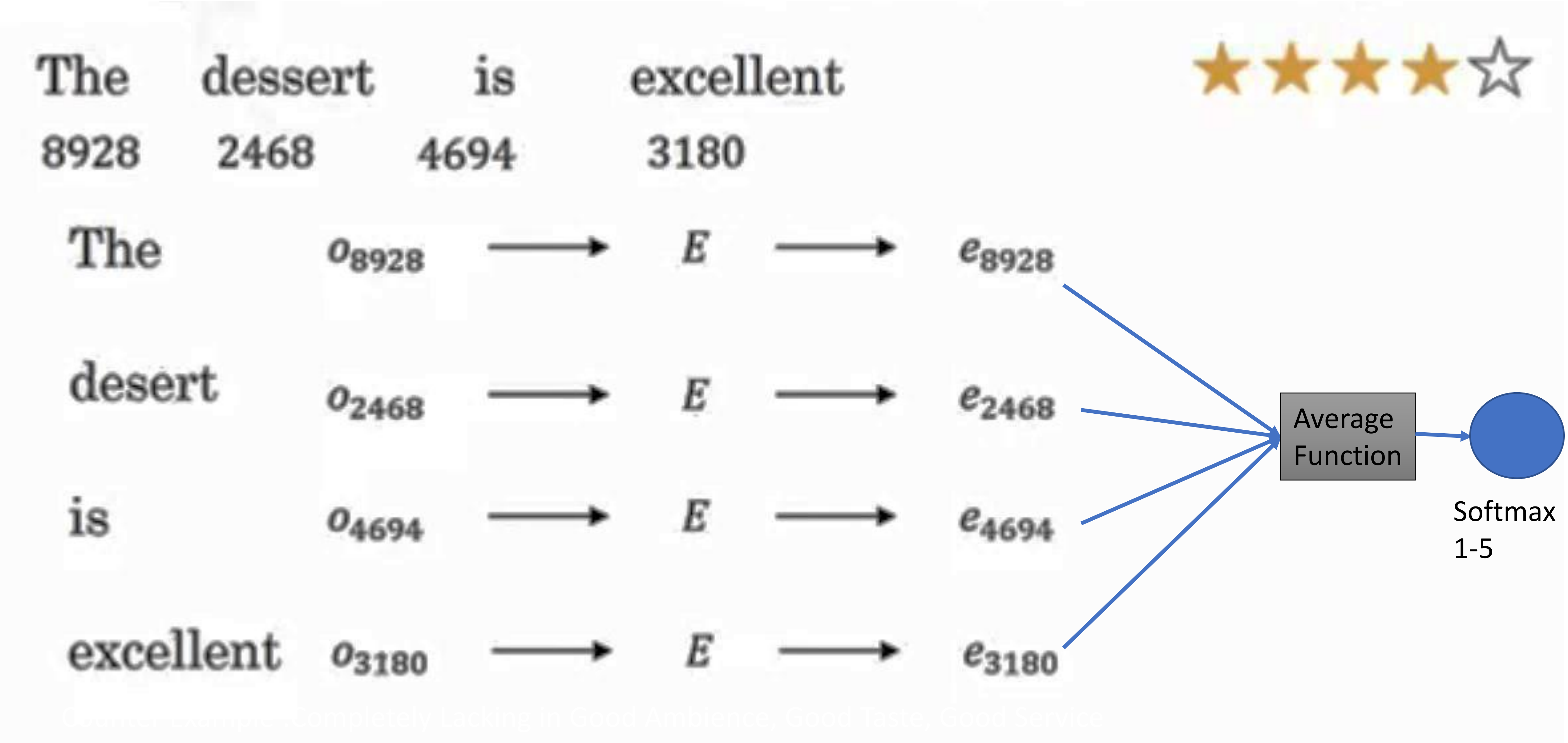
The crowd cheers. It is full of real Americans (man with hard hat, man with harder hat, gun that is alive).

PRESIDENT TRUMP (CONT’D)
The United Snakes is doing so good. Other countries are on fire. All the people on fire. Hot fire too. Not us. Our flag is so beautiful.

President Trump salutes a flag that says: **ARBY’S FOOD IS FINE TO EAT.** The crowd howls. They love this flag of America.

PRESIDENT TRUMP (CONT’D)
I signed a bill. No more swamp. Swamp gone. Swamp is in Mexico now. It’s on fire. Great deal for us.

Simple Sentiment Classification Model



The	dessert	is	excellent
8928	2468	4694	3180

★★★★☆

The $o_{8928} \longrightarrow E \longrightarrow e_{8928}$

desert $o_{2468} \longrightarrow E \longrightarrow e_{2468}$

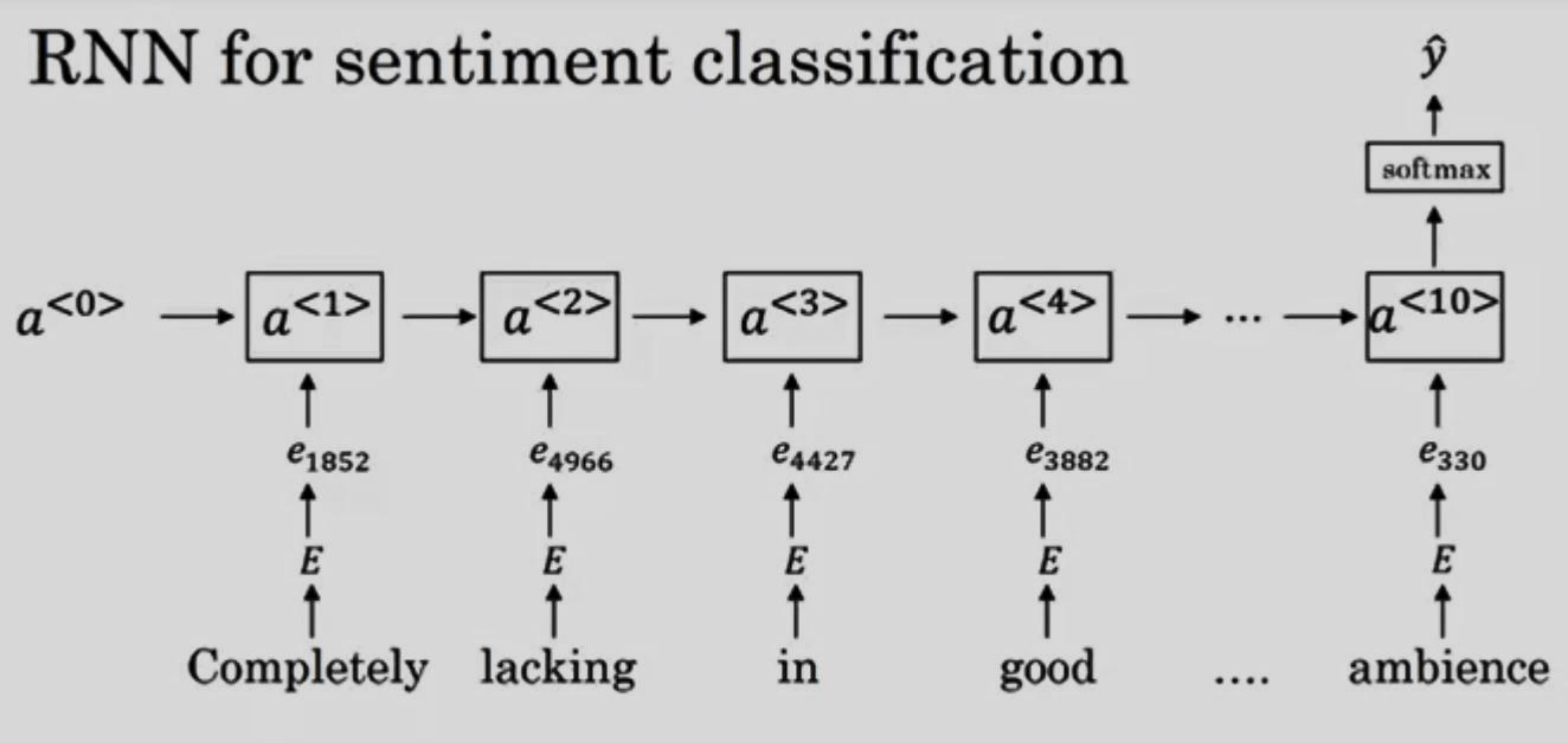
is $o_{4694} \longrightarrow E \longrightarrow e_{4694}$

excellent $o_{3180} \longrightarrow E \longrightarrow e_{3180}$

Average Function

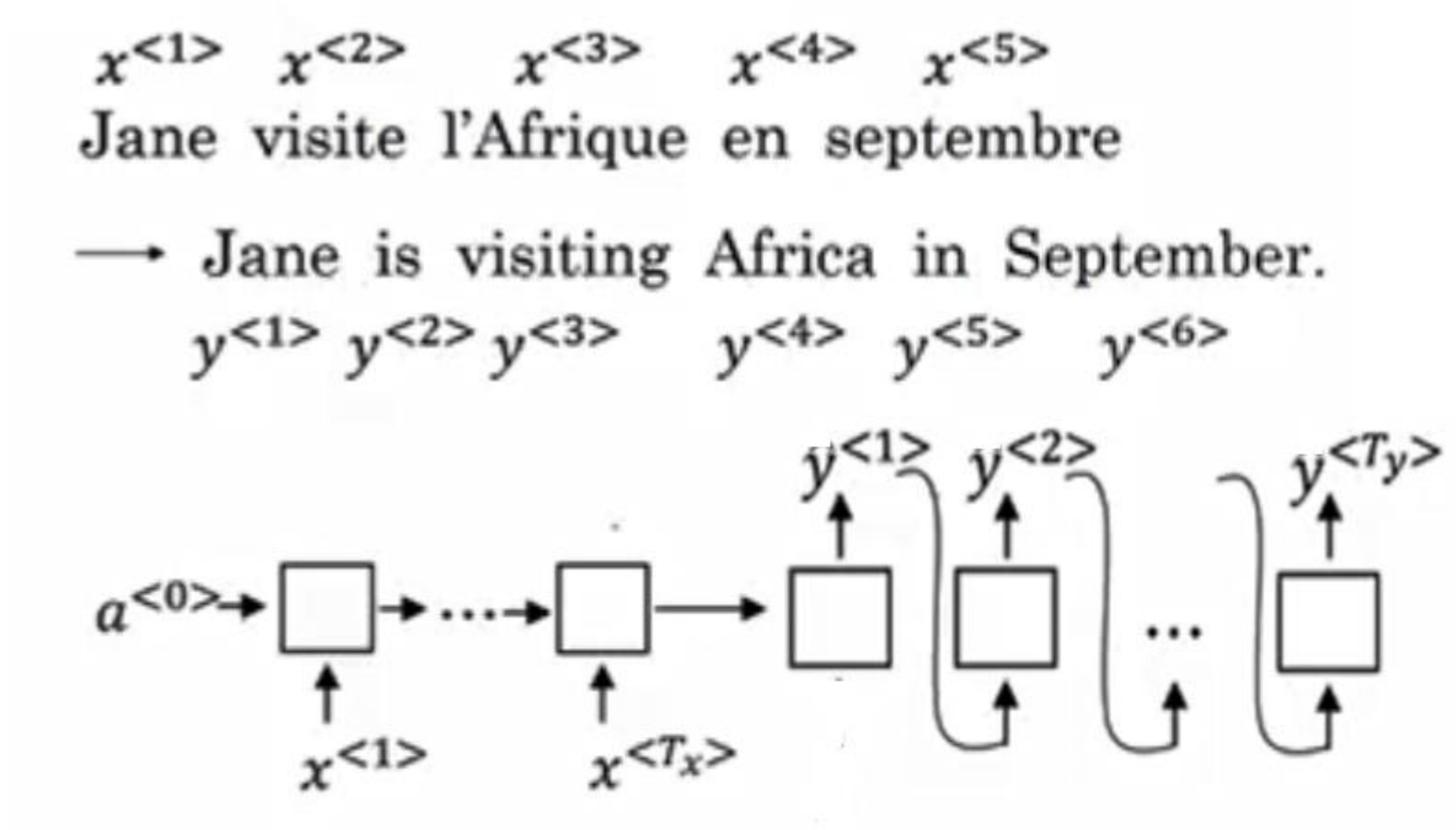
Softmax
1-5

Counter example: Completely Lacking in Good Ambience, Good Taste, Good Service



RNN for sentiment classification

Sequence to Sequence Model



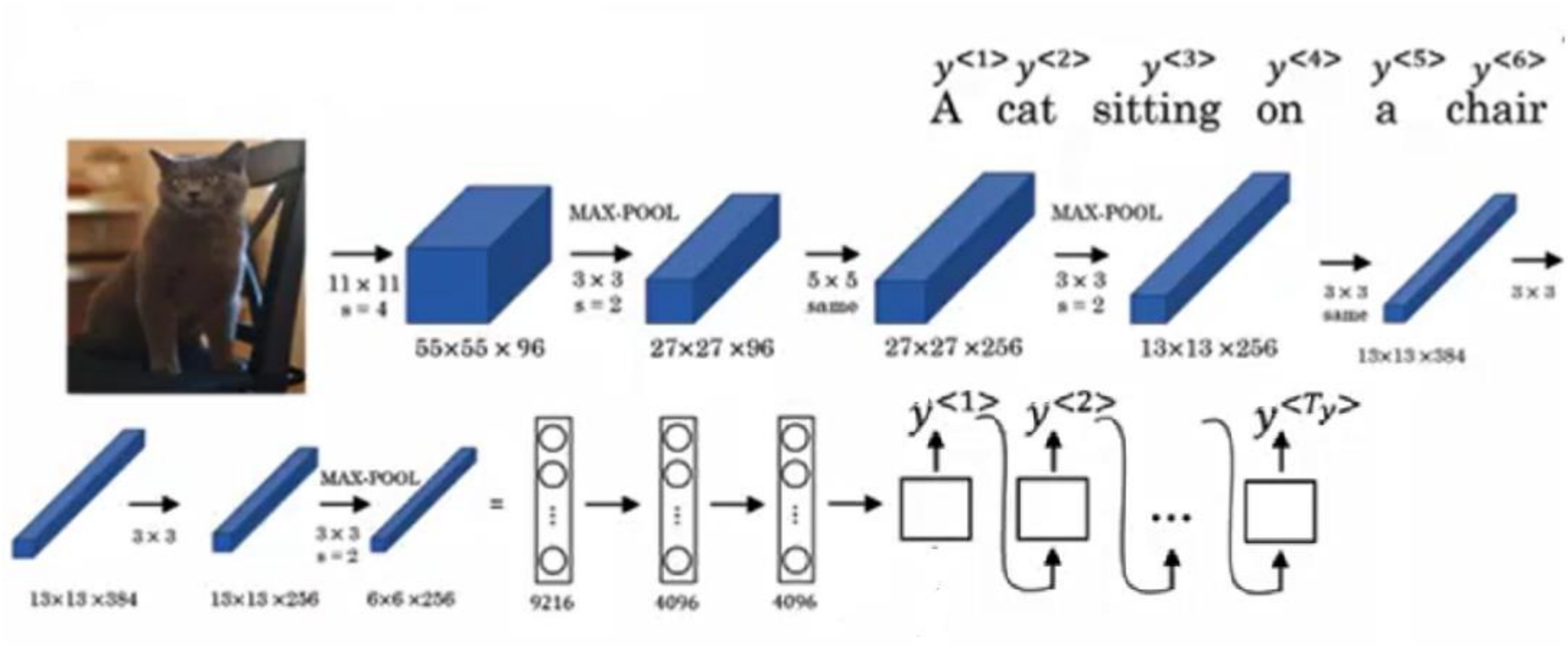
$x^{<1>}$ $x^{<2>}$ $x^{<3>}$ $x^{<4>}$ $x^{<5>}$

Jane visite l'Afrique en septembre

→ Jane is visiting Africa in September.

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

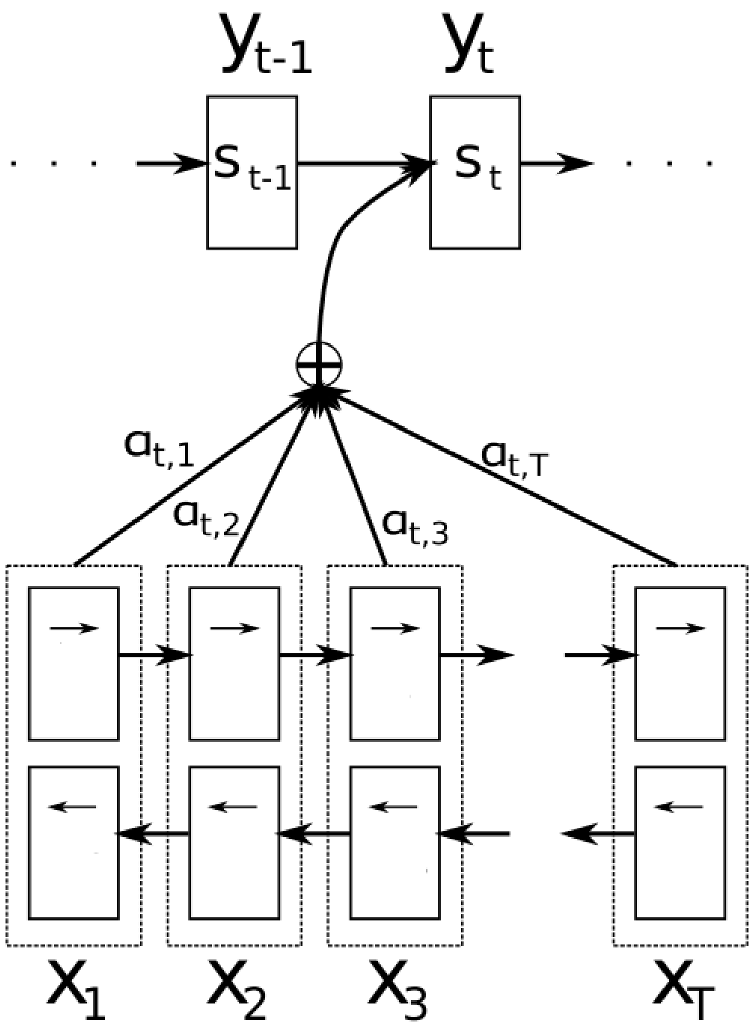
Image captioning model



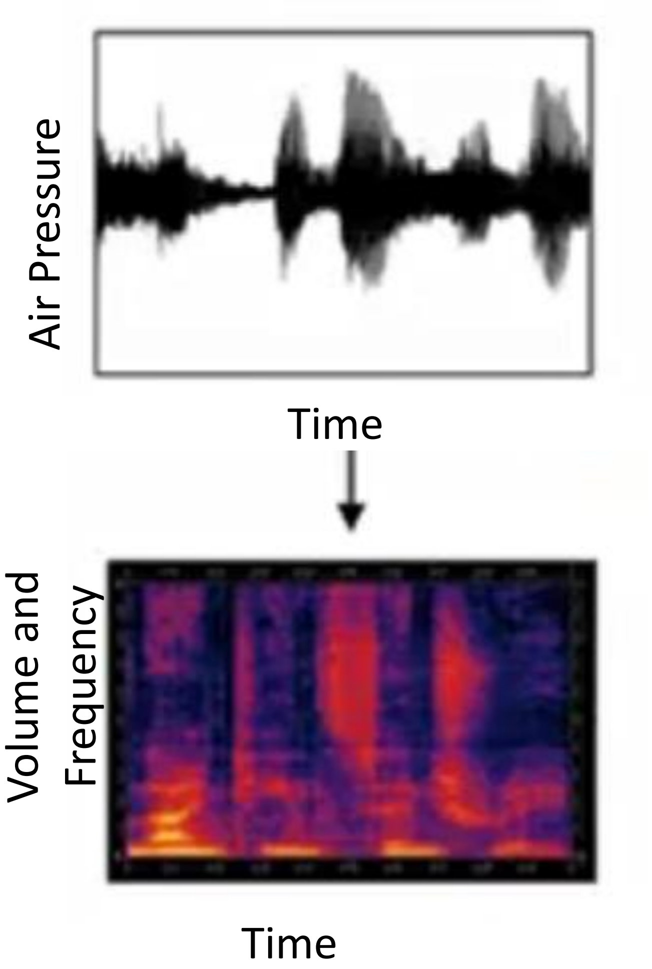
Problem of long sequences: Attention model

- When the sequences grow beyond a certain length (20 or more), then the bleu score goes down considerably and generally does not has good mapping with the goodness of the translation.
- To handle this recently Researchers came out with a new model called attention mechanism.
- It basically tells that how much attention needs to paid to each word of the source sequence for every position of the translated sequence.
- Total of attention values for any particular word should be equal to 1, so we can think of it as a softmax classifier with a small neural network for determining the probabilities of each attention value vector.

Attention Model



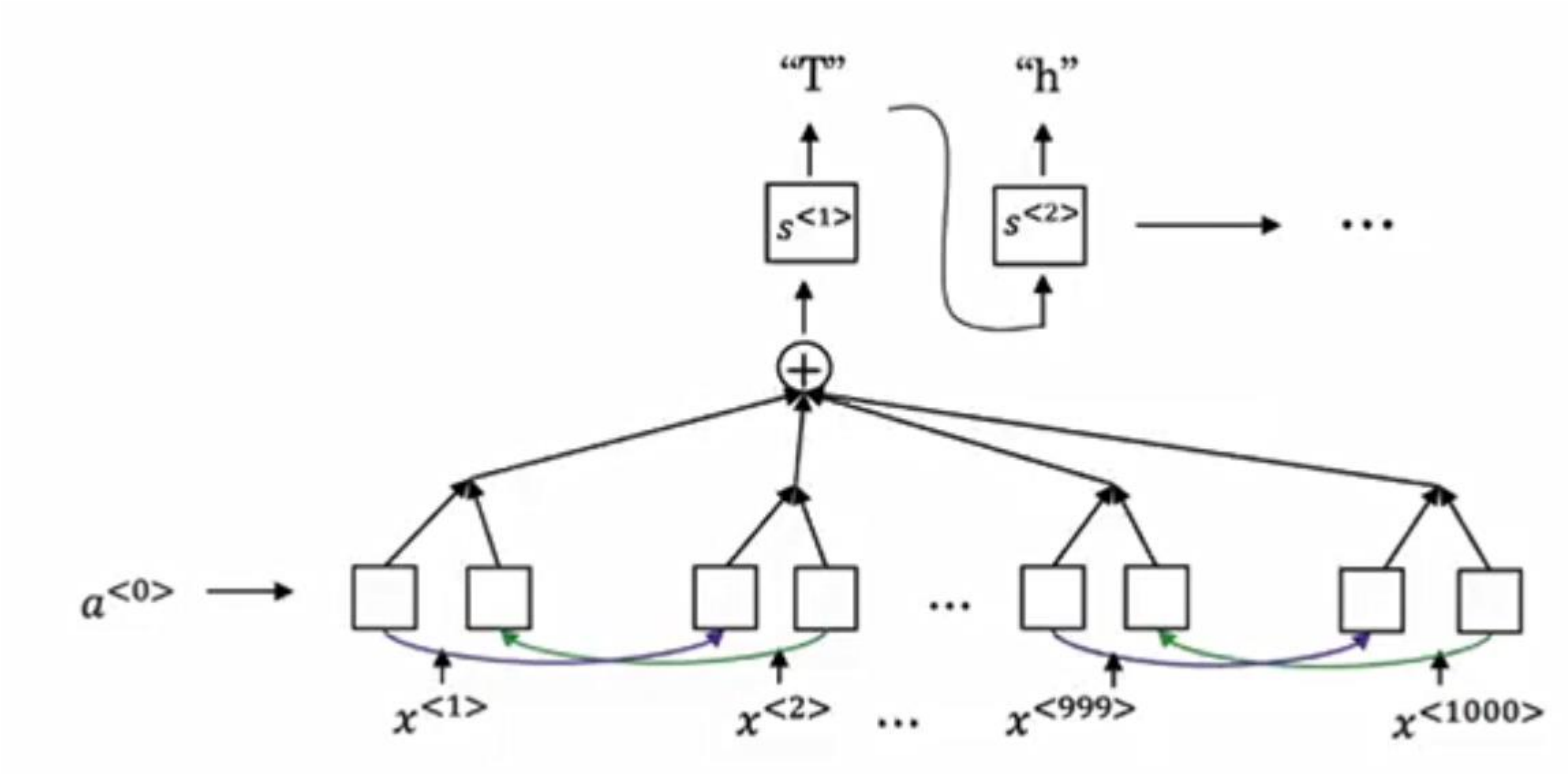
Speech Recognition



Previously we used to have Hand-engineered features consisting of different phonemes

Thousands of hours of speech/audio data is used to train the data depending upon the application

Attention model for speech Systems


"T"
"h"
$s^{<1>}$
$s^{<2>}$
...
$a^{<0>}$
...
$x^{<1>}$
$x^{<2>}$
...
$x^{<999>}$
$x^{<1000>}$