Introduction to Recurrent Neural Networks

STAT5241 Section 2

Statistical Machine Learning

Xiaofei Shi

Sequences

- Words, Letters

> *50 years ago, the fathers of artificial intelligence convinced everybody that logic was the key to intelligence. Somehow we had to get computers to do logical reasoning. The alternative approach, which they thought was crazy, was to forget logic and try and understand how networks of brain cells learn things. Curiously, two people who rejected the logic based approach to AI were Turing and Von Neumann. If either of them had lived I think things would have turned out differently... now neural networks are everywhere and the crazy approach is winning.*

Geoff Hinton

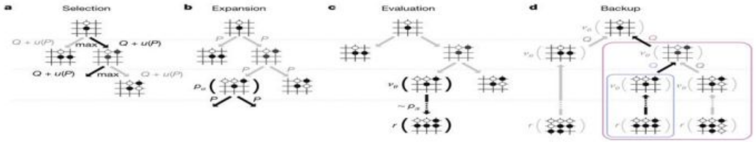
- Speech

- Images, Videos

- Programs

```
while (*d++ = *s++);
```

- Sequential Decision Making (RL)



Classical Models for Sequence Predictions

- Sequence prediction was classically handled as a structured prediction task
  - Most were built on conditional independence assumptions
  - Other such as DAGGER were based on supervisory signals and auxiliary information

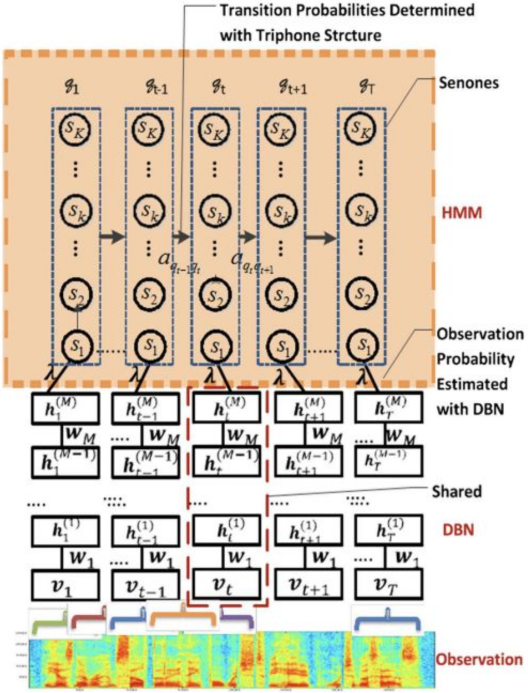
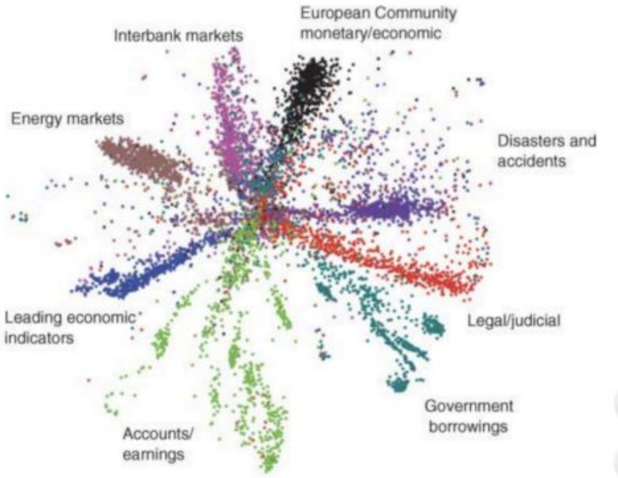


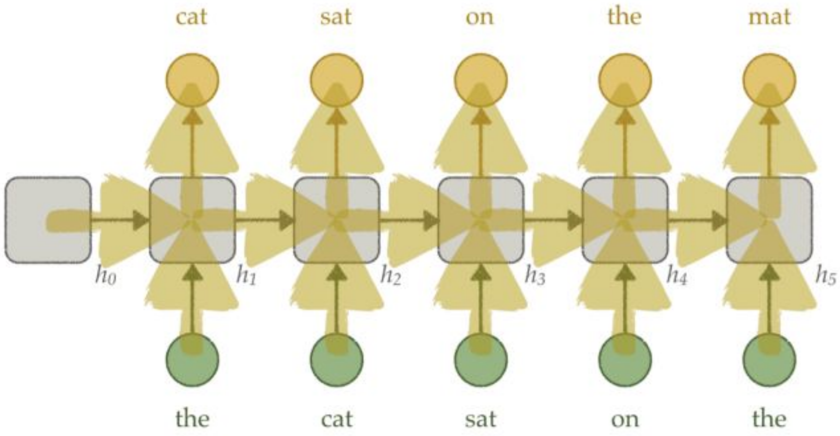
Figure credit: Li Deng

Key Ingredients:

Neural Embeddings



Recurrent Language Models



Hinton, G., Salakhutdinov, R. "Reducing the Dimensionality of Data with Neural Networks." *Science (2006)*

Mikolov, T., et al. "Recurrent neural network based language model." *Interspeech (2010)*

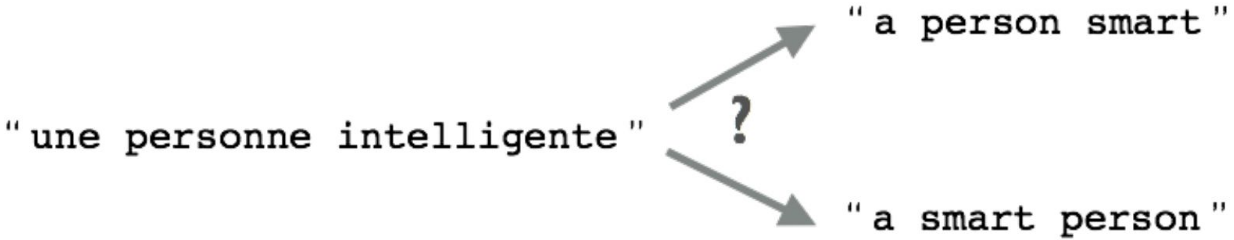
Columbia University

Language Models

- A language model is a probabilistic model that assigns probabilities to any sequence of words

$$p(w_1, \dots, w_T)$$

  - ➢ language modeling is the task of learning a language model that assigns high probabilities to well formed sentences
  - ➢ plays a crucial role in speech recognition and machine translation systems



COLUMBIA UNIVERSITY

Language Models

- An assumption frequently made is the n^{th} order Markov assumption

$$p(w_1, \ldots, w_T) = \prod_{t=1}^{T} p(w_t \mid w_{t-(n-1)}, \ldots, w_{t-1})$$

- ➢ the t^{th} word was generated based only on the n−1 previous words
- ➢ we will refer to $w_{t-(n-1)}, \ldots, w_{t-1}$ as the context

Columbia University

$$P(w_1, w_2, \ldots, w_{T-1}, w_T) = \prod_{t=1}^{T} P(w_t | w_{t-1}, w_{t-2}, \ldots, w_1)$$

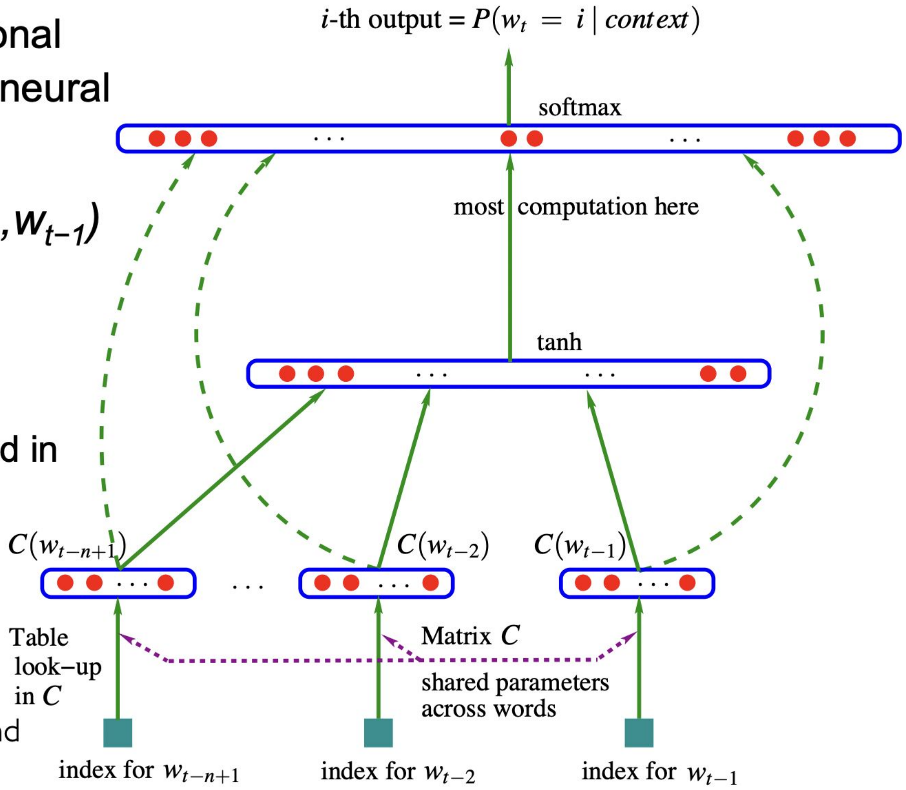
the	cat	sat	on	the	mat	$P(w_1)$
the	**cat**	sat	on	the	mat	$P(w_2 \| w_1)$
the	cat	**sat**	on	the	mat	$P(w_3 \| w_2, w_1)$
the	cat	sat	**on**	the	mat	$P(w_4 \| w_3, w_2, w_1)$
the	cat	sat	on	**the**	mat	$P(w_5 \| w_4, w_3, w_2, w_1)$
the	cat	sat	on	the	**mat**	$P(w_6 \| w_5, w_4, w_3, w_2, w_1)$

Neural Language Models

- Model the conditional distributions with a neural network:

$p(w_t \mid w_{t-(n-1)}, \ldots, w_{t-1})$

  - ➢ learn word representations to allow transfer to n-grams not observed in training corpus



C is a continuous representation of words, usually a lookup table

Bengio, Ducharme, Vincent and Jauvin, 2003

Neural Language Models

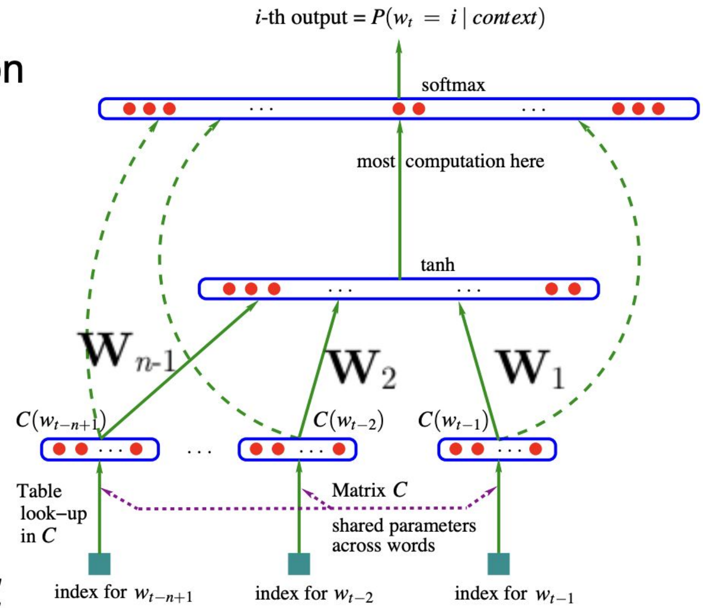
- We know how to propagate gradients in such a network

  - we know how to compute the gradient for the linear activation of the hidden layer $\nabla_{\mathbf{a}(\mathbf{x})} l$

  - let's note the submatrix connecting w_{t-i} and the hidden layer as $\boldsymbol{W_i}$

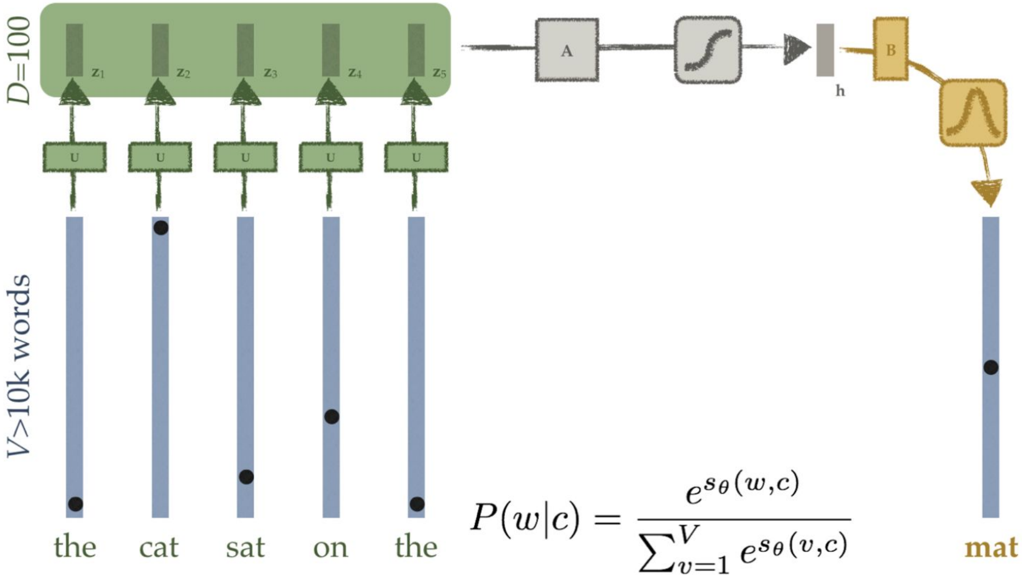
- The gradient wrt $C(w)$ for any w is

$$\nabla_{C(w)} l = \sum_{i=1}^{n-1} 1_{(w_{t-i}=w)} \, \mathbf{W}_i^\top \, \nabla_{\mathbf{a}(\mathbf{x})} l$$



COLUMBIA UNIVERSITY

$$p(w_t|w_1,\ldots,w_{t-1}) = p_\theta(w_t|f_\theta(w_1,\ldots,w_{t-1}))$$



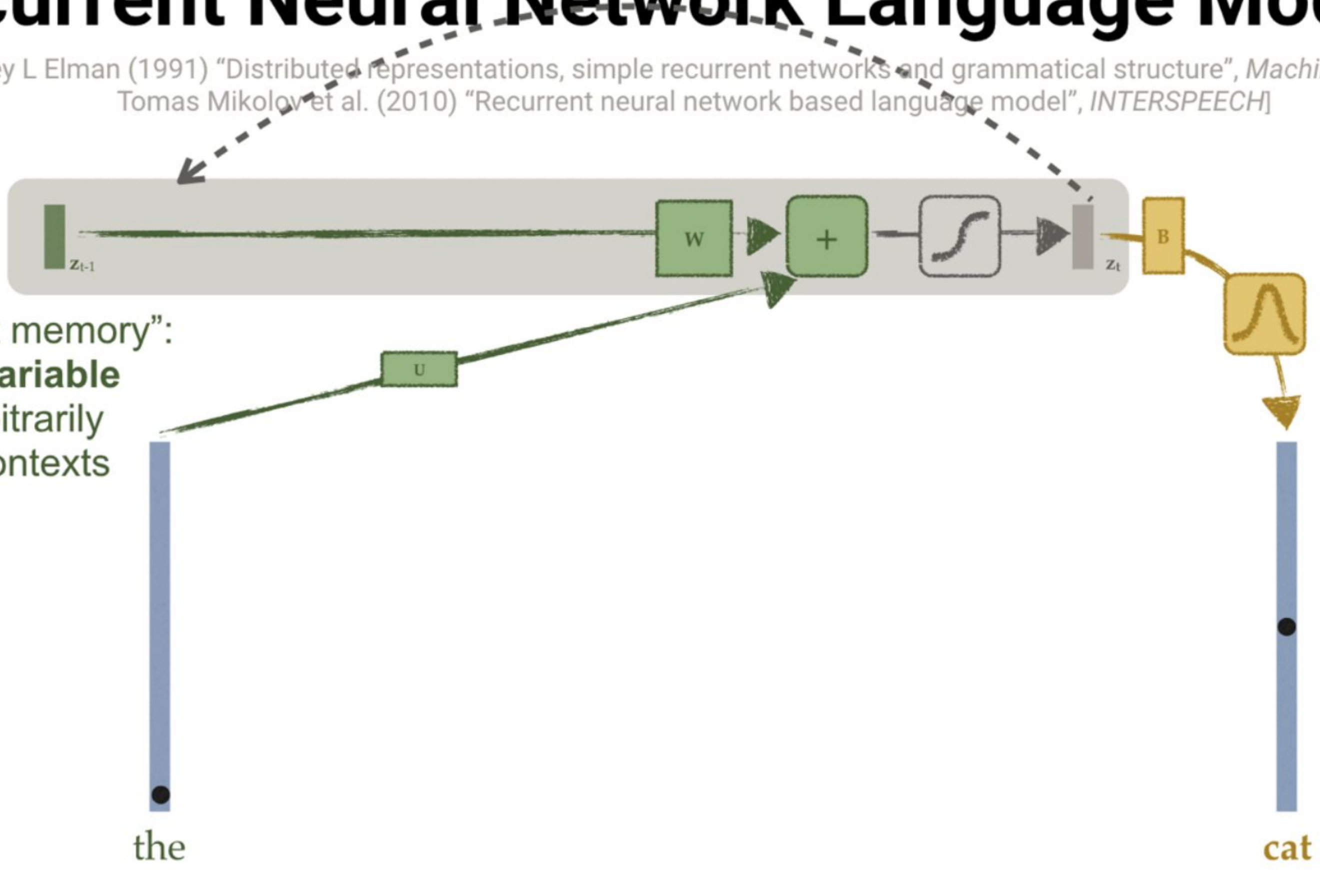
$$P(w|c) = \frac{e^{s_\theta(w,c)}}{\sum_{v=1}^{V} e^{s_\theta(v,c)}}$$

Bengio, Y. et al., “A Neural Probabilistic Language Model”, *JMLR (2001, 2003)*
Mnih, A., Hinton, G., “Three new graphical models for statistical language modeling”, *ICML 2007*

Slide Credit: Piotr Mirowski

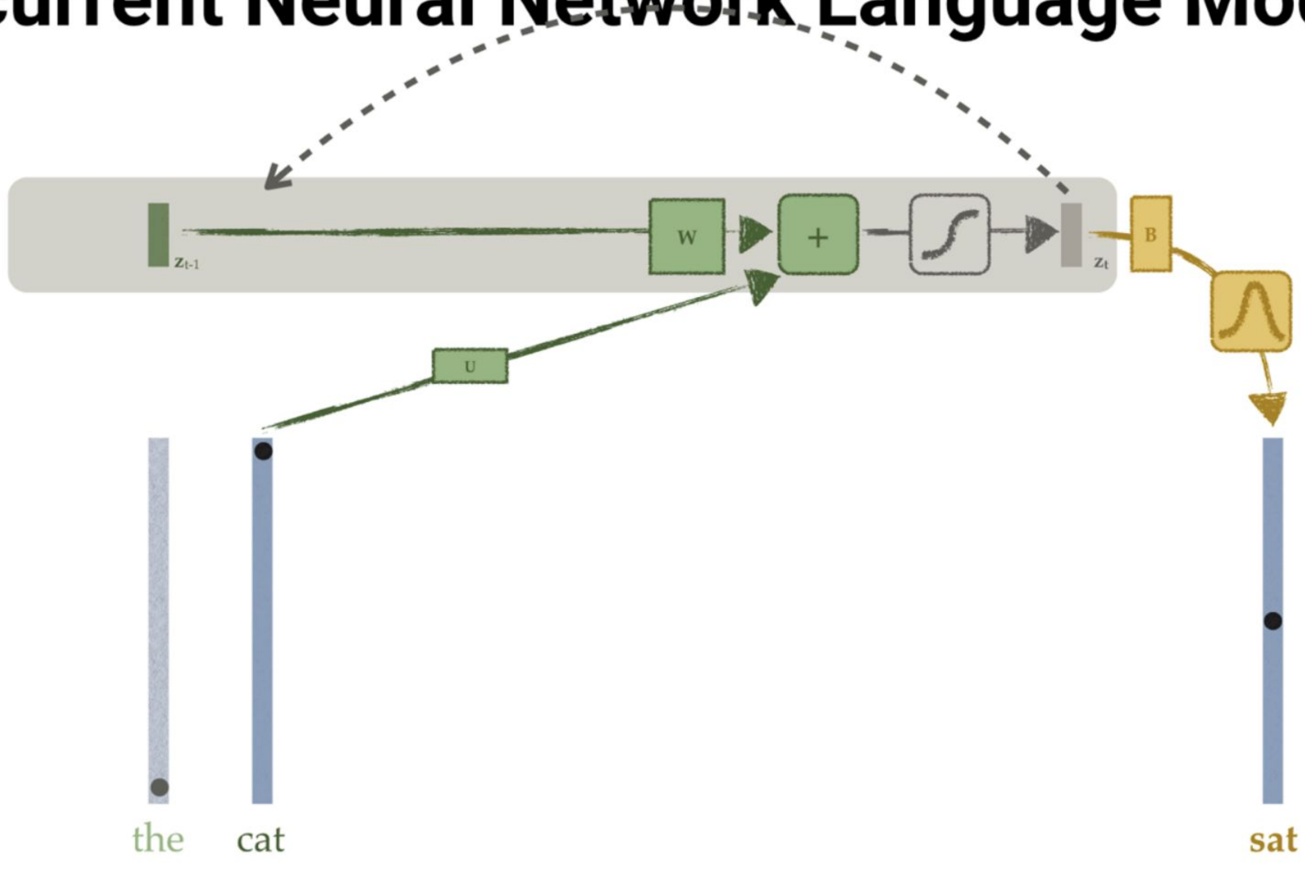
Recurrent Neural Network Language Models

[Jeffrey L Elman (1991) “Distributed representations, simple recurrent networks and grammatical structure”, *Machine Learning*; Tomas Mikolov et al. (2010) “Recurrent neural network based language model”, *INTERSPEECH*]

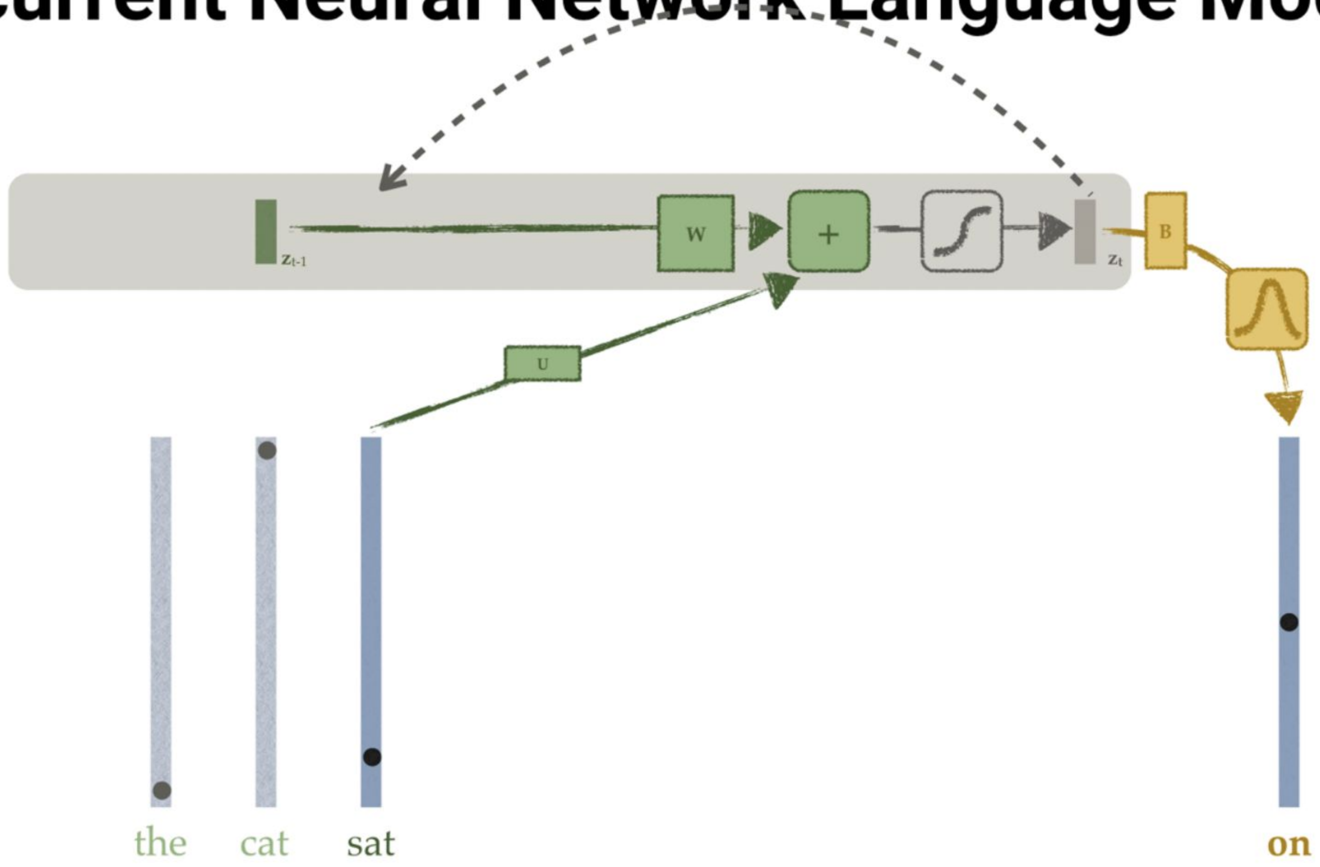


“persistent memory”:
state variable
for arbitrarily
long contexts

Recurrent Neural Network Language Models

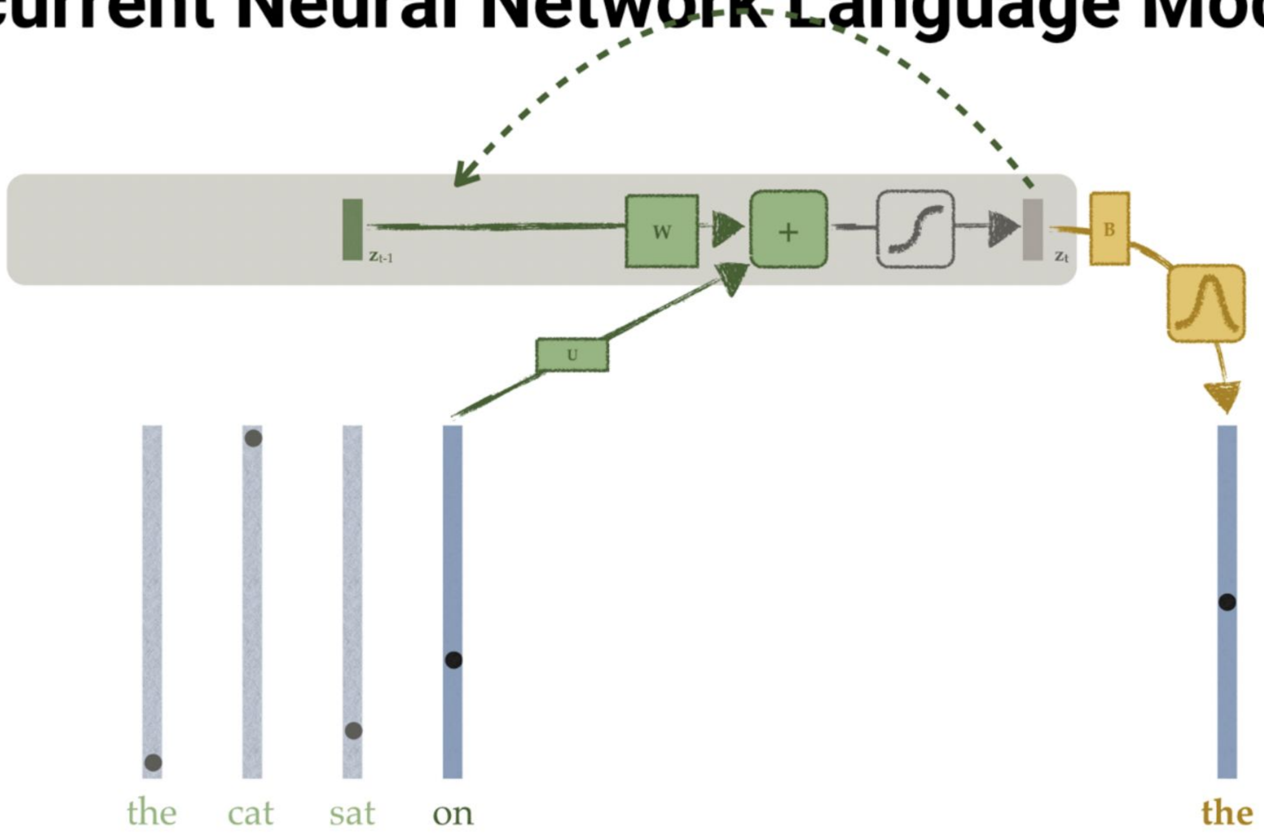


Recurrent Neural Network Language Models



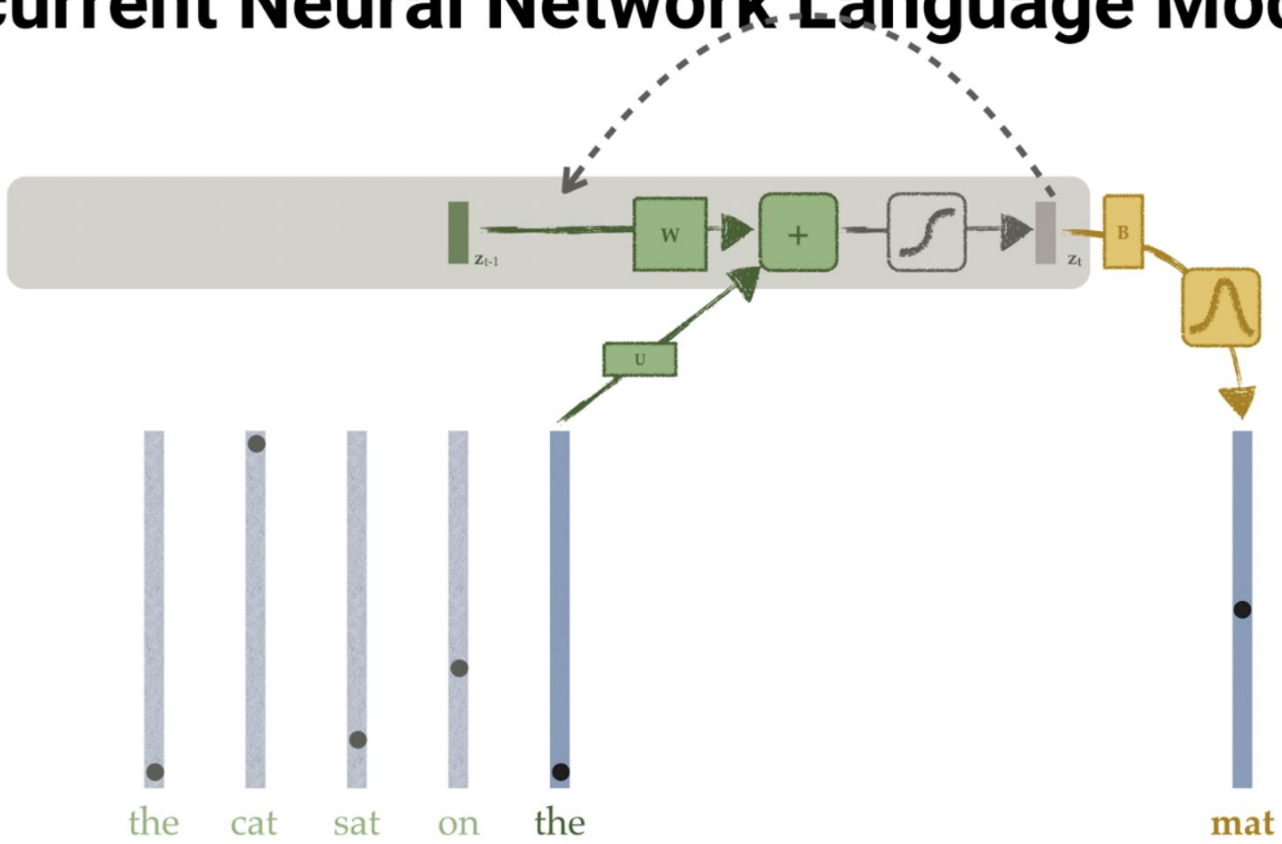
COLUMBIA UNIVERSITY

Recurrent Neural Network Language Models



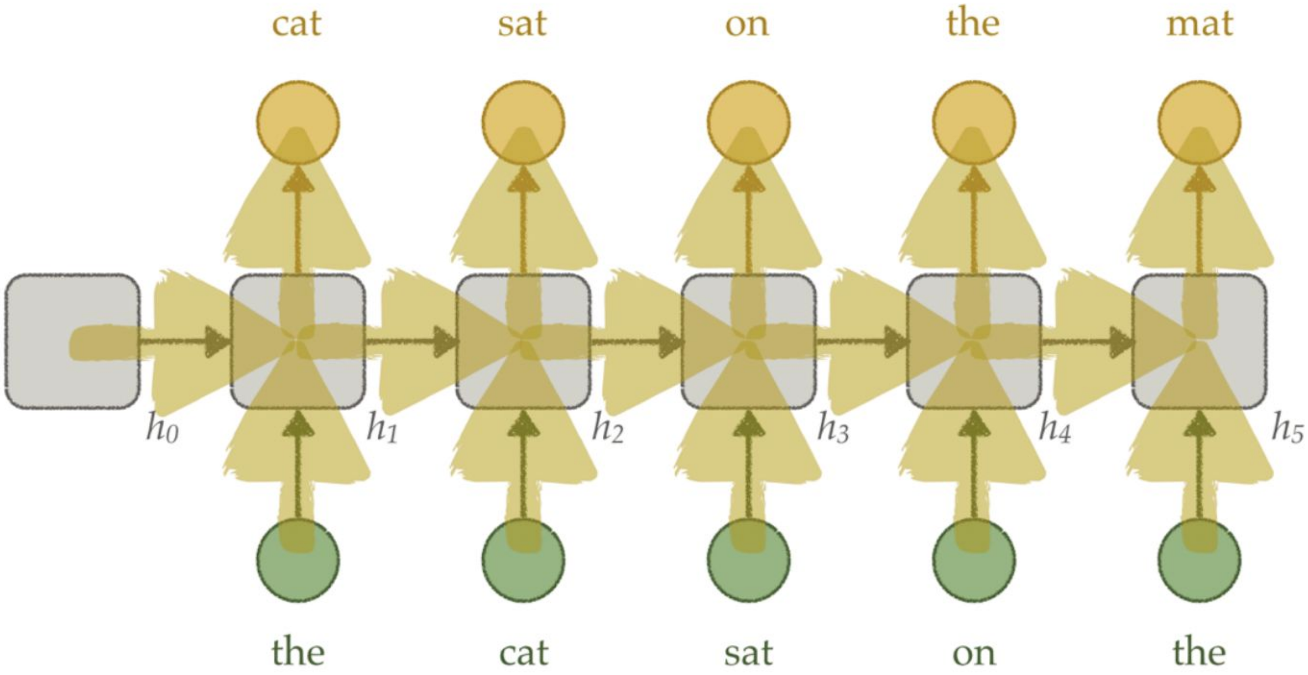
COLUMBIA UNIVERSITY

Recurrent Neural Network Language Models



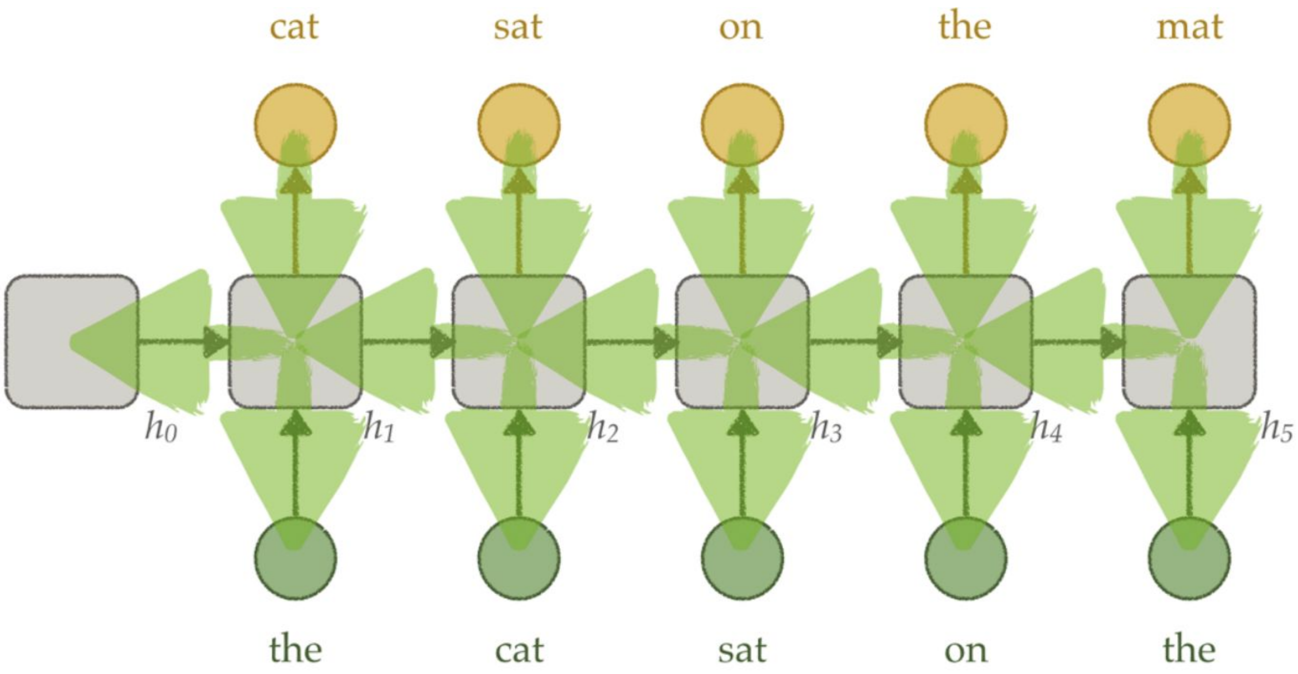
COLUMBIA UNIVERSITY

Recurrent neural networks: forward pass

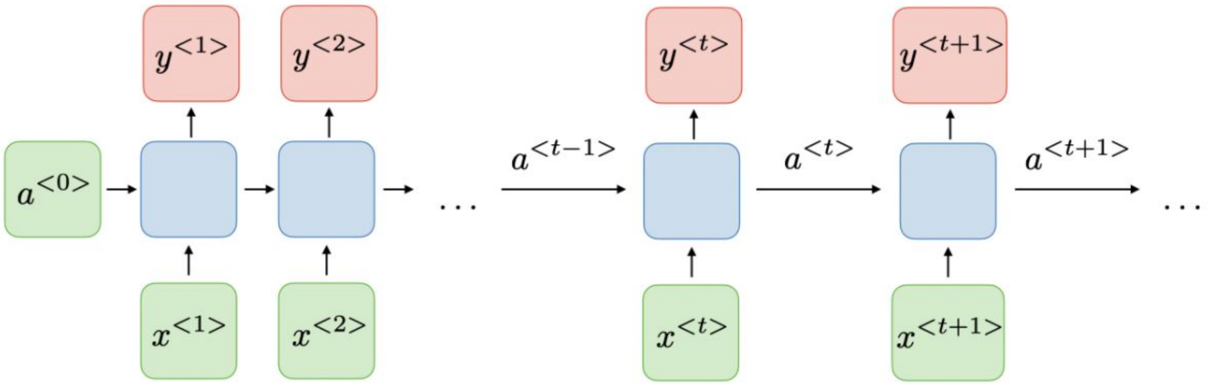


Columbia University

Recurrent neural networks: backward updates



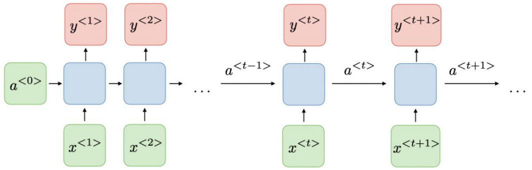
RNN: architecture



For each timestep t, the activation $a^{<t>}$ and the output $y^{<t>}$ are expressed as follows:

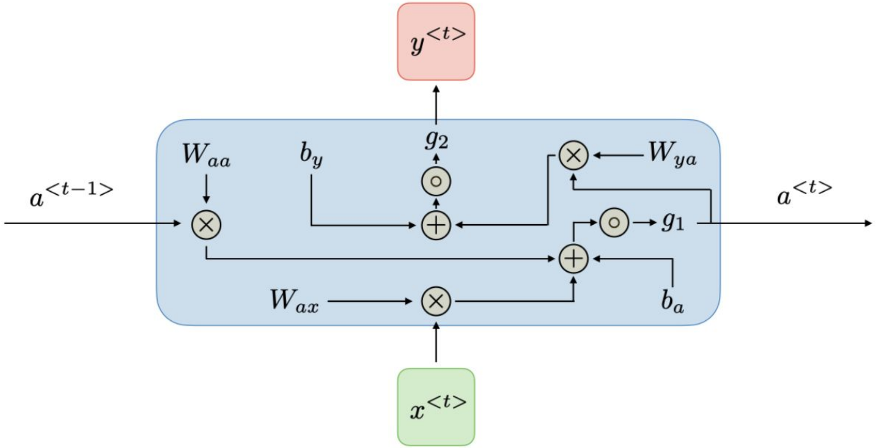
$$\boxed{a^{<t>} = g_1(W_{aa}a^{<t-1>} + W_{ax}x^{<t>} + b_a)} \quad \text{and} \quad \boxed{y^{<t>} = g_2(W_{ya}a^{<t>} + b_y)}$$

RNN: architecture



For each timestep t, the activation $a^{<t>}$ and the output $y^{<t>}$ are expressed as follows:

$$\boxed{a^{<t>} = g_1(W_{aa}a^{<t-1>} + W_{ax}x^{<t>} + b_a)} \quad \text{and} \quad \boxed{y^{<t>} = g_2(W_{ya}a^{<t>} + b_y)}$$



Columbia University

Summary

Advantages:

- Possibility of processing input of any length
- Model size not increasing with size of input
- Computation takes into account historical information
- Weights are shared across time

Disadvantages :

- Computation being slow
- Difficulty of accessing information from a long time ago
- Cannot consider any future input for the current state

References

- Christopher Bishop: Pattern Recognition and Machine Learning, Chapter 5
- Ziv Bar-Joseph, Tom Mitchell, Pradeep Ravikumar and Aarti Singh: CMU 10-701
- Ryan Tibshirani: CMU 10-725
- Ruslan Salakhutdinov: CMU 10-703
- https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks

Columbia University