Deep Learning for Computer Vision

Attention Models in Vision: An Introduction

Vineeth N Balasubramanian

Department of Computer Science and Engineering
Indian Institute of Technology, Hyderabad

Review

Question

What do you think will happen if you train a model on normal videos and do inference on a reversed video?

Review

Question

What do you think will happen if you train a model on normal videos and do inference on a reversed video?

Depends on the application/task. May work for certain tasks, say differentiating walking versus jumping, but may not for recognizing a tennis forehand.

Review

Question

What do you think will happen if you train a model on normal videos and do inference on a reversed video?

Depends on the application/task. May work for certain tasks, say differentiating walking versus jumping, but may not for recognizing a tennis forehand.

An interesting problem in this context: Finding the arrow of time, see Wei et al, Learning and Using the Arrow of Time, CVPR 2018

Review

- RNNs can be used to efficiently model sequential data
- RNNs use Backpropagation through time (BPTT) approach as training method
- RNNs suffer from vanishing & exploding gradients problems
- Gradient clipping can be used to control exploding gradient
- LSTM/ GRU units use gates to help mitigate the vanishing gradients problem

Review

- RNNs can be used to efficiently model sequential data
- RNNs use Backpropagation through time (BPTT) approach as training method
- RNNs suffer from vanishing & exploding gradients problems
- Gradient clipping can be used to control exploding gradient
- LSTM/ GRU units use gates to help mitigate the vanishing gradients problem

But...is this modeling sufficient?

RNN Tasks

Image Captioning

A woman is throwing a frisbee
in a park

RNN Tasks

Image Captioning

A woman is throwing a frisbee in a park

Neural Machine Translation

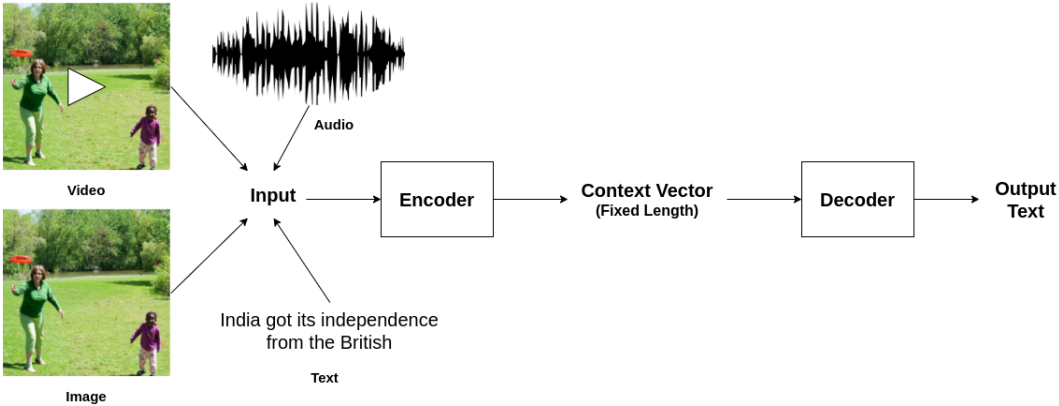
India got its independence from the British

भारत को अंग्रेजों से आजादी मिली

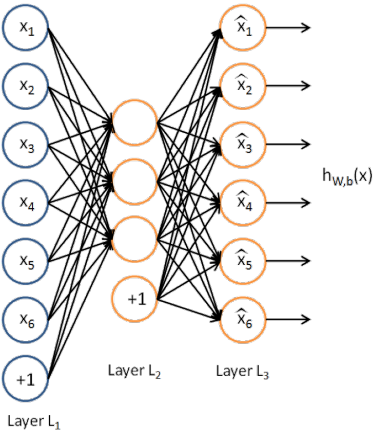
L'accord sur l'Espace économique européen a été signé en août 1992

The agreement on the European Economic Area was signed in August 1992

Encoder-Decoder Modeling

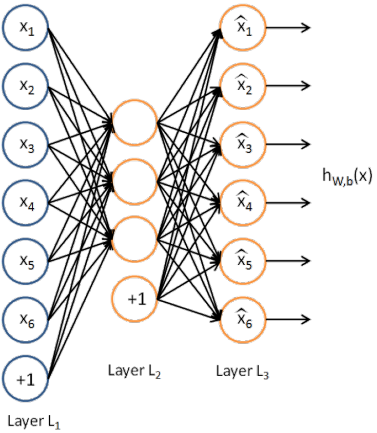


Autoencoders



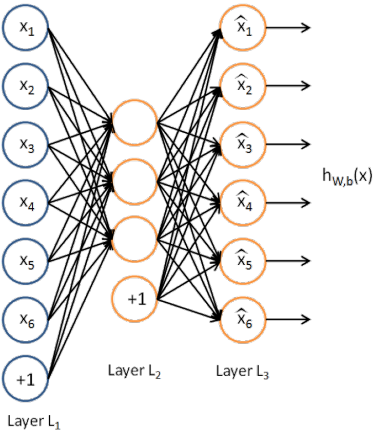
- An **autoencoder neural network** is an unsupervised learning model that applies backpropagation, setting target values to be equal to inputs themselves, i.e. $\mathbf{y}_i = \mathbf{x}_i$

Autoencoders



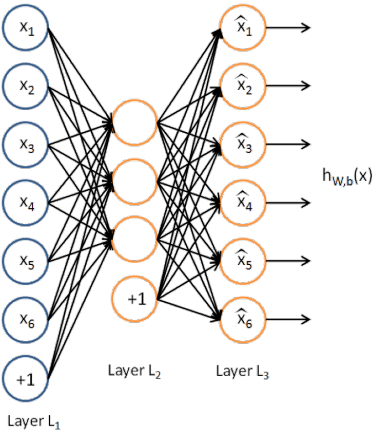
- An **autoencoder neural network** is an unsupervised learning model that applies backpropagation, setting target values to be equal to inputs themselves, i.e. $\mathbf{y}_i = \mathbf{x}_i$
- Learns a function $f_{W,b}(\mathbf{x})\mathbf{x}$; in other words, learn an approximation to identity function, output $\hat{\mathbf{x}}$ close to $\mathbf{x}$

Autoencoders



- An **autoencoder neural network** is an unsupervised learning model that applies backpropagation, setting target values to be equal to inputs themselves, i.e. $\mathbf{y}_i = \mathbf{x}_i$
- Learns a function $f_{W,b}(\mathbf{x})\mathbf{x}$; in other words, learn an approximation to identity function, output $\hat{\mathbf{x}}$ close to $\mathbf{x}$
- Loss function?

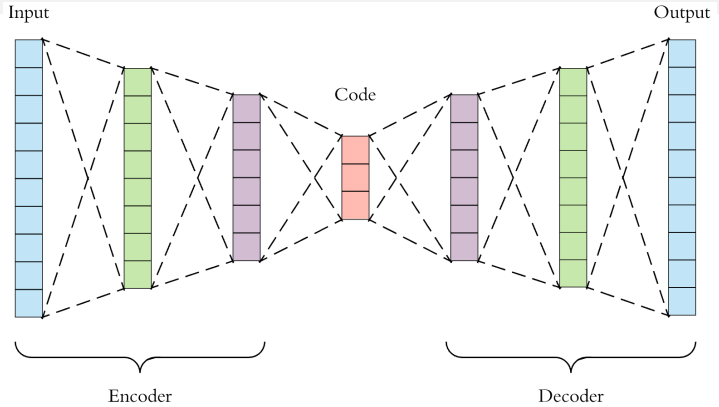
Autoencoders



- An **autoencoder neural network** is an unsupervised learning model that applies backpropagation, setting target values to be equal to inputs themselves, i.e. $\mathbf{y}_i = \mathbf{x}_i$
- Learns a function $f_{W,b}(\mathbf{x})\mathbf{x}$; in other words, learn an approximation to identity function, output $\hat{\mathbf{x}}$ close to $\mathbf{x}$
- Loss function? Mean Squared Error $\mathcal{L} = ||\mathbf{x} - \hat{\mathbf{x}}||_2^2$

Credit: Andrew Ng, CS294A, Stanford Univ

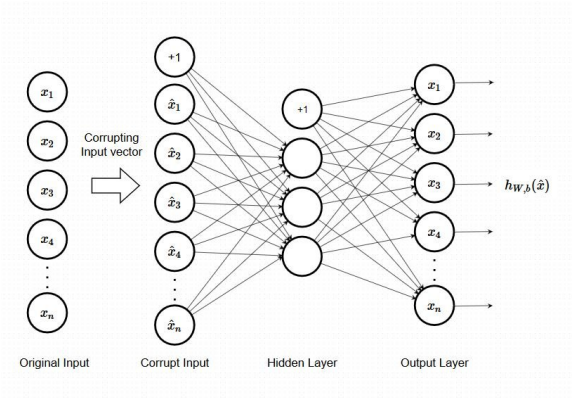
Deep Autoencoders



Both encoder and decoder can have many layers too - in standard deep autoencoders, the architecture in encoder is often mirrored in decoder (not always so though)

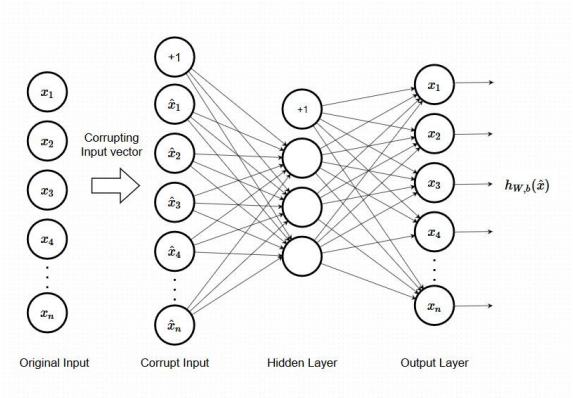
Credit: Arden Dertat, TowardsDataScience

Denoising Autoencoders



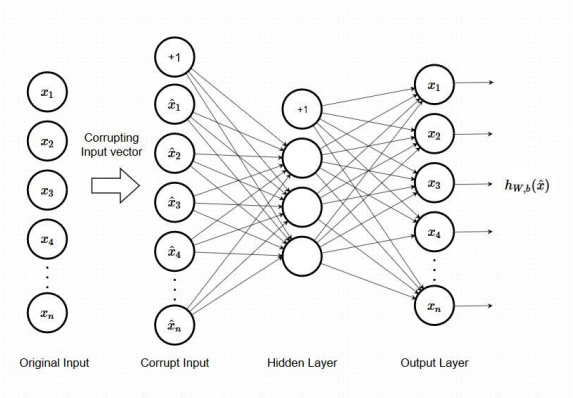
- Variant of autoencoder, where input is perturbed with noise (e.g. Gaussian), but network is asked to predict original input without noise

Denoising Autoencoders



- Variant of autoencoder, where input is perturbed with noise (e.g. Gaussian), but network is asked to predict original input without noise
- Loss function?

Denoising Autoencoders



- Variant of autoencoder, where input is perturbed with noise (e.g. Gaussian), but network is asked to predict original input without noise
- Loss function? Mean Squared Error between output and original uncorrupted input

Credit: Kumar et al, Static hand gesture recognition using stacked Denoising Sparse Autoencoders, IC3 2014

More on Autoencoders

Why should the hidden layers be smaller in size than input layer?

More on Autoencoders

Why should the hidden layers be smaller in size than input layer?

- Autoencoder (AE) with hidden layer with lesser dimension than input layer called **undercomplete AE** $\implies$ AE learns a lower-dimensional representation on suitable manifold of input data

More on Autoencoders

Why should the hidden layers be smaller in size than input layer?

- Autoencoder (AE) with hidden layer with lesser dimension than input layer called **undercomplete AE** $\implies$ AE learns a lower-dimensional representation on suitable manifold of input data
- Autoencoder (AE) with hidden layer with higher dimension than input layer called **overcomplete AE** $\implies$ AE could learn trivial solutions, by copying input!

More on Autoencoders

Why should the hidden layers be smaller in size than input layer?

- Autoencoder (AE) with hidden layer with lesser dimension than input layer called **undercomplete AE** $\implies$ AE learns a lower-dimensional representation on suitable manifold of input data
- Autoencoder (AE) with hidden layer with higher dimension than input layer called **overcomplete AE** $\implies$ AE could learn trivial solutions, by copying input!

Are autoencoders then dimensionality reduction methods?

More on Autoencoders

Why should the hidden layers be smaller in size than input layer?

- Autoencoder (AE) with hidden layer with lesser dimension than input layer called **undercomplete AE** $\implies$ AE learns a lower-dimensional representation on suitable manifold of input data
- Autoencoder (AE) with hidden layer with higher dimension than input layer called **overcomplete AE** $\implies$ AE could learn trivial solutions, by copying input!

Are autoencoders then dimensionality reduction methods?

- Yes, indeed; undercomplete AEs are dimensionality reduction methods

More on Autoencoders

Why should the hidden layers be smaller in size than input layer?

- Autoencoder (AE) with hidden layer with lesser dimension than input layer called **undercomplete AE** $\implies$ AE learns a lower-dimensional representation on suitable manifold of input data
- Autoencoder (AE) with hidden layer with higher dimension than input layer called **overcomplete AE** $\implies$ AE could learn trivial solutions, by copying input!

Are autoencoders then dimensionality reduction methods?

- Yes, indeed; undercomplete AEs are dimensionality reduction methods
- Do you see connections to PCA?

More on Autoencoders

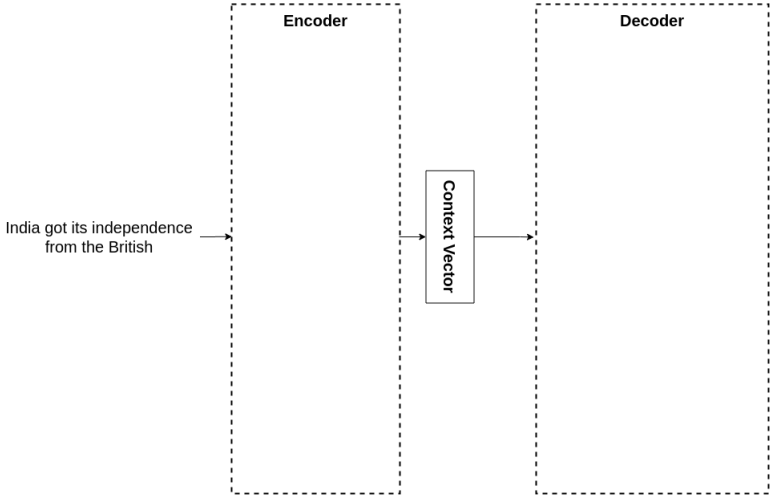
Why should the hidden layers be smaller in size than input layer?

- Autoencoder (AE) with hidden layer with lesser dimension than input layer called **undercomplete AE** $\implies$ AE learns a lower-dimensional representation on suitable manifold of input data
- Autoencoder (AE) with hidden layer with higher dimension than input layer called **overcomplete AE** $\implies$ AE could learn trivial solutions, by copying input!

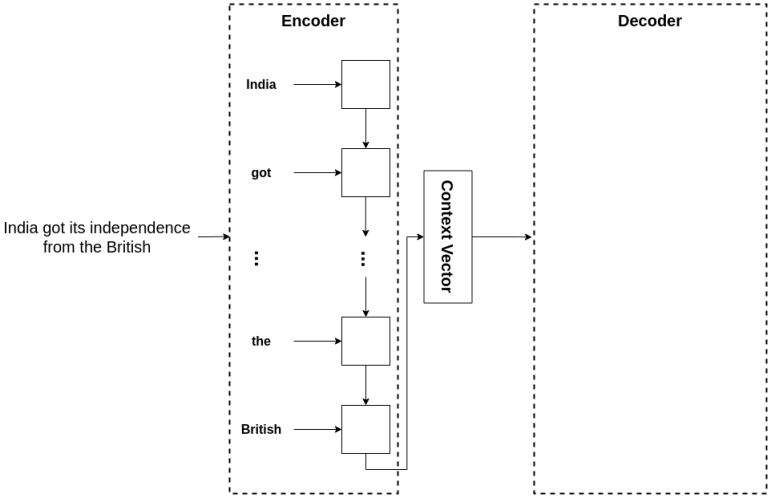
Are autoencoders then dimensionality reduction methods?

- Yes, indeed; undercomplete AEs are dimensionality reduction methods
- Do you see connections to PCA? Homework!

Back to NMT Model (Seq2Seq Model)



Back to NMT Model (Seq2Seq Model)



Back to NMT Model (Seq2Seq Model)

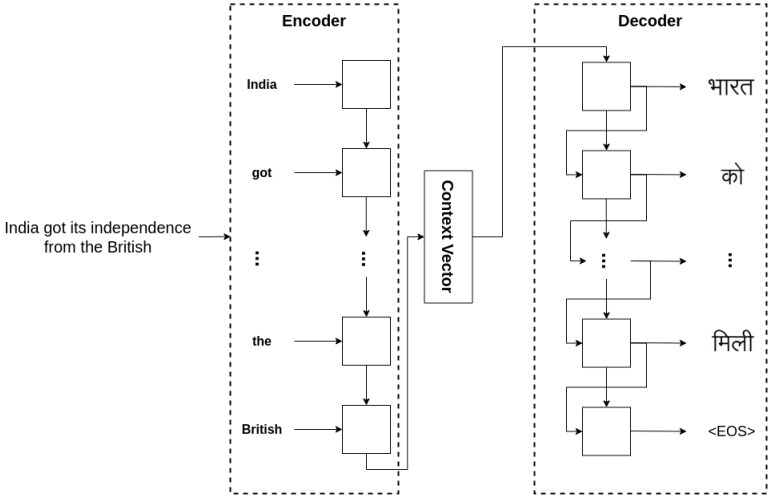


Image Captioning Model

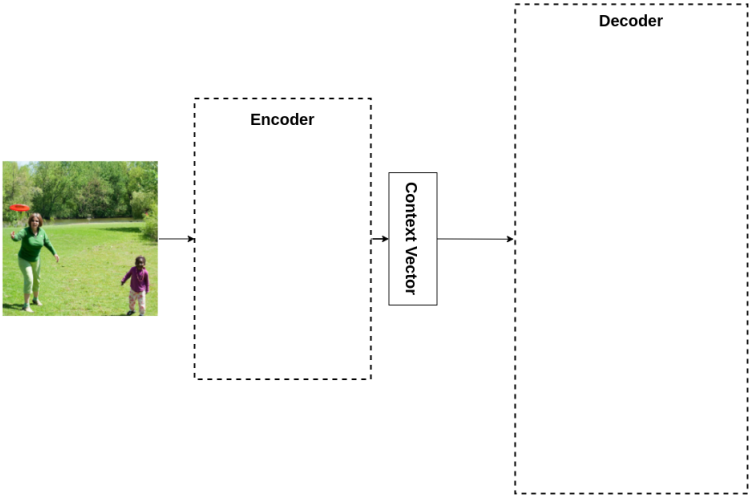


Image Captioning Model

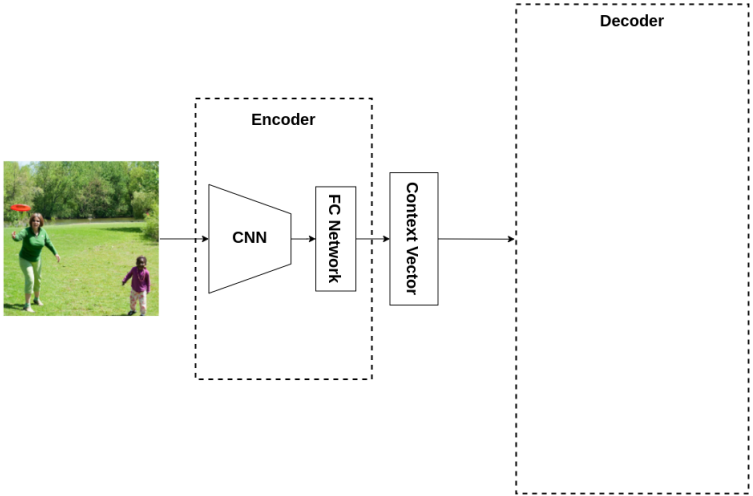
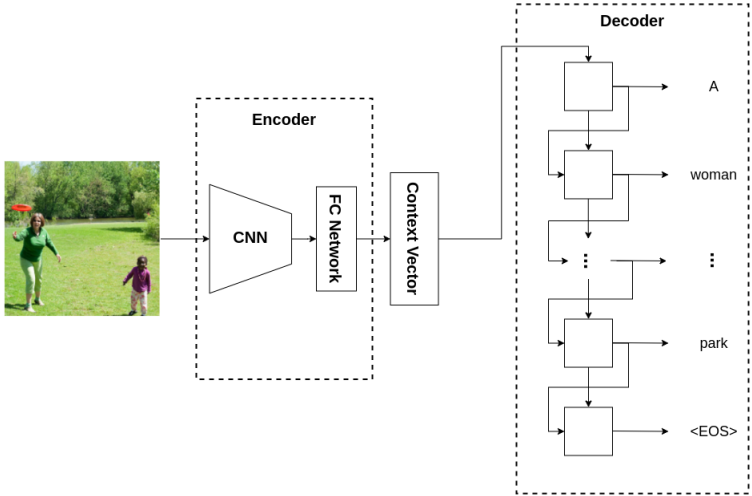
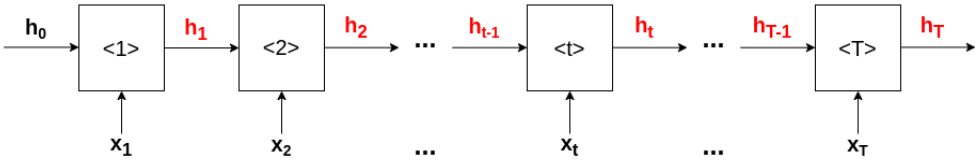


Image Captioning Model

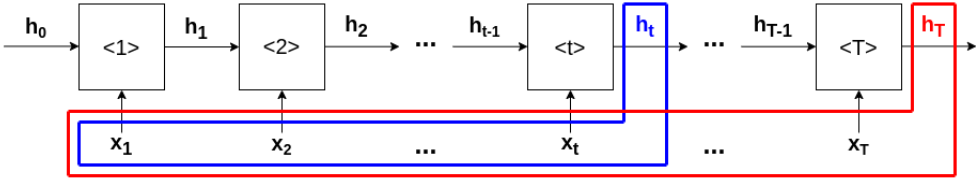


What is the problem?



Hidden states (h_t) are responsible for storing relevant input information in RNNs

What is the problem?



A hidden state at time step t (h_t) is a compressed form of all previous inputs $(x_1, x_2, ..., x_t)$

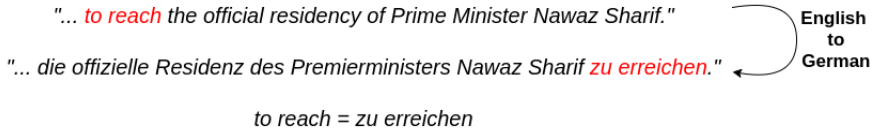
What is the problem?

After meeting a comrade at the last post station but one before Moscow, Denisov had drunk three bottles of wine with him and, despite the jolting ruts across the snow-covered road, did not once wake up on the way to Moscow, but lay at the bottom of the sleigh beside Rostov, who grew more and more impatient the nearer they got to Moscow. <EOS>

Excerpts from the work of Leo Tolstoy(~60 words)

But what if input is very long? Can h_T encode all information without forgetting? Information bottleneck!

What is the problem?



Can we guarantee that words seen at earlier input time steps be reproduced in later time steps of output?

What is the problem?

Question: What is the name of the book?
Answer: The name of the book is Lord of the Rings.

Credit: Bharath Kishore, Flickr CC License

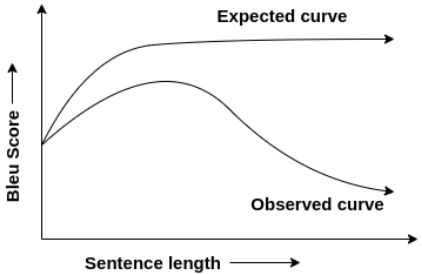
Visual Question Answering (VQA)
(To be discussed later)

Relevant information in a cluttered image should also be preserved

Failure of Encoder-Decoder Modeling

BLEU score:

- Stands for Bilingual Evaluation Understudy
- Metric for evaluating quality of machine translated text
- Can be used for other language tasks (like Image captioning, VQA, etc)
- For more information, see BLEU score, Wikipedia



Solution?

Attention: Intuition

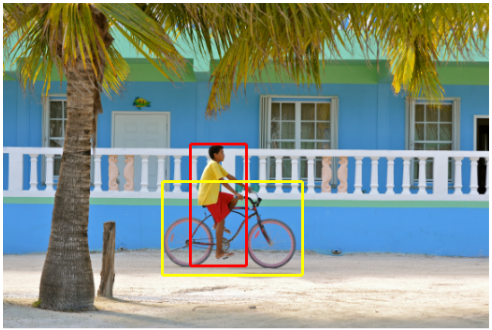
How do you answer this?

What is the boy doing?

Credit: Cecilia Schubert, Wikimedia.org, CC License

Attention: Intuition

How do you answer this?



What is the boy doing?

Identify artifacts in the image

Attention: Intuition

How do you answer this?

What is the boy doing?

Pay attention to the relevant artifacts

Attention: Intuition

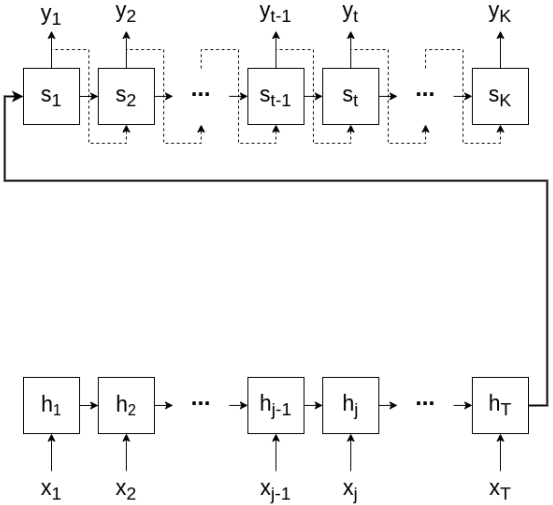
Similarly,

No student of a foreign language needs to be told that grammar is complex. By changing word sequences and by adding a range of auxiliary verbs and suffixes, we are able to communicate tiny variations in meaning. We can turn a statement into a question, state whether an action has taken place or is soon to take place, and perform many other word tricks to convey subtle differences in meaning. Nor is this complexity inherent to the English language. All languages, even those of so-called 'primitive' tribes have clever grammatical components. The Cherokee pronoun system, for example, can distinguish between 'you and I', 'several other people and I' and 'you, another person and I'. In English, all these meanings are summed up in the one, crude pronoun 'we'. Grammar is universal and plays a part in every language, no matter how widespread it is. So the question which has baffled many linguists is - who created grammar?

Summary

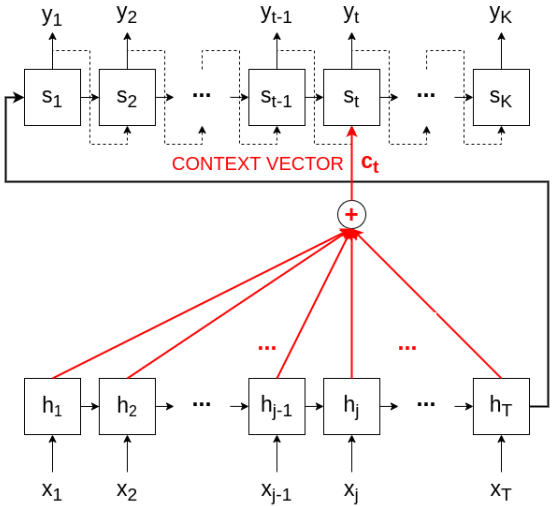
Grammar is inherently complex. This complexity is independant of how widely a language is used. If grammar is universal, who created it?

Attention Mechanism: Temporal Data



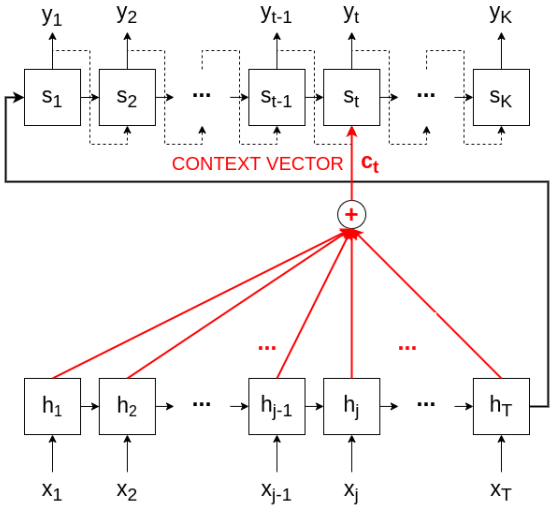
- Given an encoder, with h_j as hidden state at time-step j, and decoder, with s_t as hidden state at time-step t

Attention Mechanism: Temporal Data



- Given an encoder, with h_j as hidden state at time-step j, and decoder, with s_t as hidden state at time-step t
- Attention mechanism creates shortcut connections between context vector (c_t) and the entire source input (X)

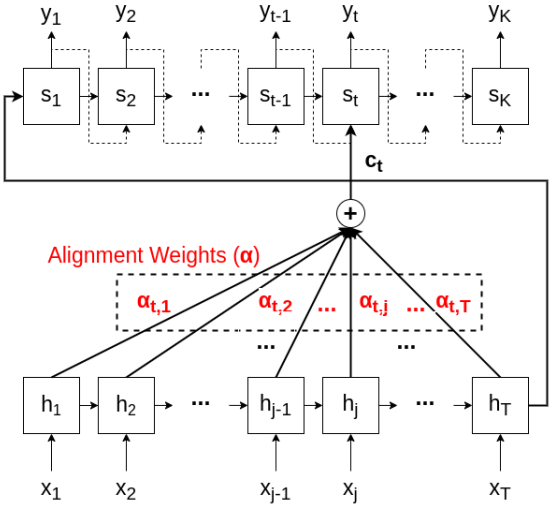
Attention Mechanism: Temporal Data



- Given an encoder, with h_j as hidden state at time-step j, and decoder, with s_t as hidden state at time-step t
- Attention mechanism creates shortcut connections between context vector (c_t) and the entire source input (X)
- Decoder hidden state at time t (s_t) given by:

$$s_t = f(s_{t-1}, y_{t-1}, c_t)$$

Attention Mechanism: Temporal Data



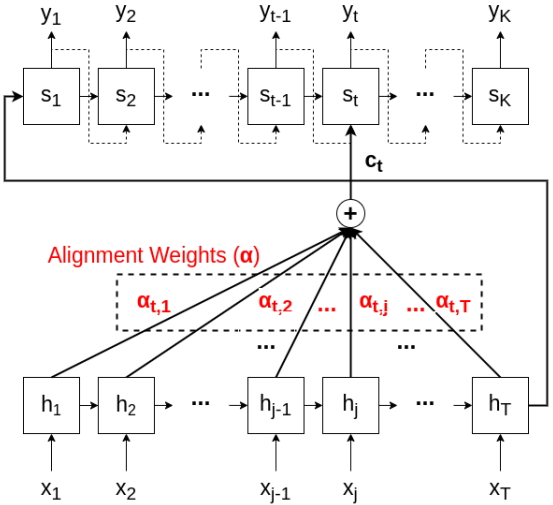
- Context vector (c_t) given by:

$$c_t = \sum_{j=1}^{T} \alpha_{t,j} h_j$$

$\alpha_{t,j}$ gives degree of alignment between s_{t-1} and h_j:

$$\alpha_{t,j} = \frac{\exp(score(s_{t-1}, h_j))}{\sum_{j'=1}^{T} \exp(score(s_{t-1}, h_{j'}))}$$

Attention Mechanism: Temporal Data



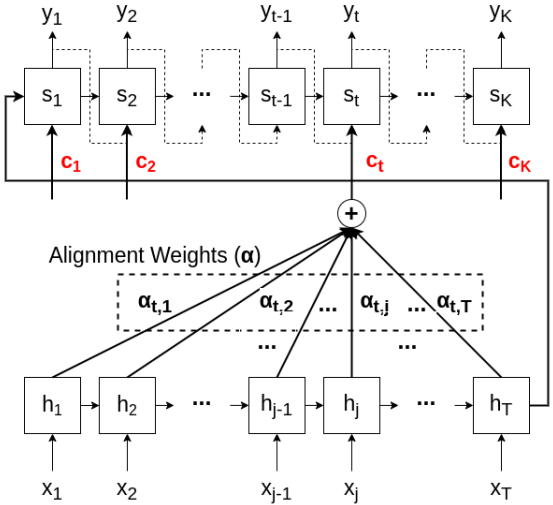
- Context vector (c_t) given by:

$$c_t = \sum_{j=1}^{T} \alpha_{t,j} h_j$$

$\alpha_{t,j}$ gives degree of alignment between s_{t-1} and h_j:

$$\alpha_{t,j} = \frac{\exp(score(s_{t-1}, h_j))}{\sum_{j'=1}^{T} \exp(score(s_{t-1}, h_{j'}))}$$

Attention Mechanism: Temporal Data



- Context vector (c_t) given by:

$$c_t = \sum_{j=1}^{T} \alpha_{t,j} h_j$$

$\alpha_{t,j}$ gives degree of alignment between s_{t-1} and h_j:

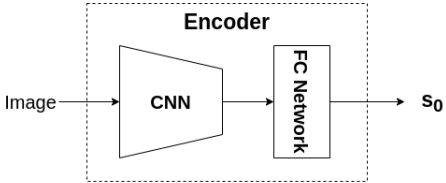
$$\alpha_{t,j} = \frac{\exp(score(s_{t-1}, h_j))}{\sum_{j'=1}^{T} \exp(score(s_{t-1}, h_{j'}))}$$

Alignment Scores

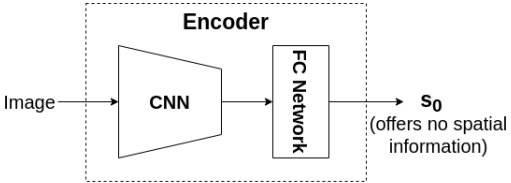
Name	Alignment Score Function
Content-based Attention	$\text{score}(s_t, h_i) = \text{cosine}(s_t, h_i)$
Additive Attention	$\text{score}(s_t, h_i) = v_a^T \tanh(W_a[s_t; h_i])$
Location-Based Attention	$\alpha_{t,j} = \text{softmax}(W_a s_t)$
General Attention	$\text{score}(s_t, h_i) = s_t^T W_a h_i$
Dot-Product Attention	$\text{score}(s_t, h_i) = s_t^T h_i$
Scaled Dot-Product Attention	$\text{score}(s_t, h_i) = \frac{s_t^T h_i}{\sqrt{n}}$

Credit: Lilian Weng, Attention? Attention!, Github Blog

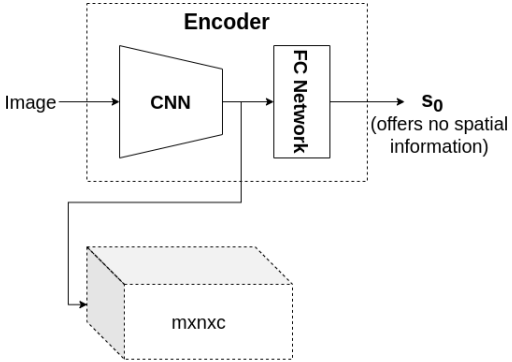
Attention Mechanism: Spatial Data



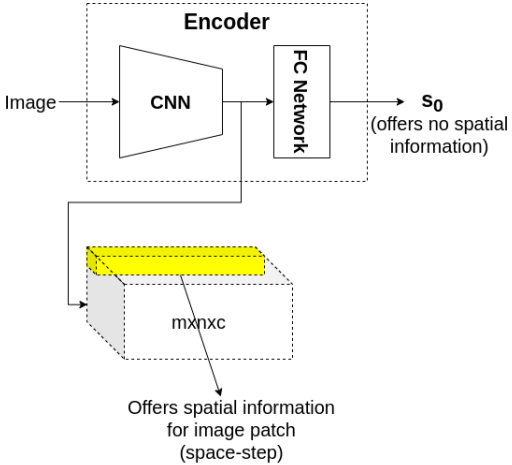
Attention Mechanism: Spatial Data



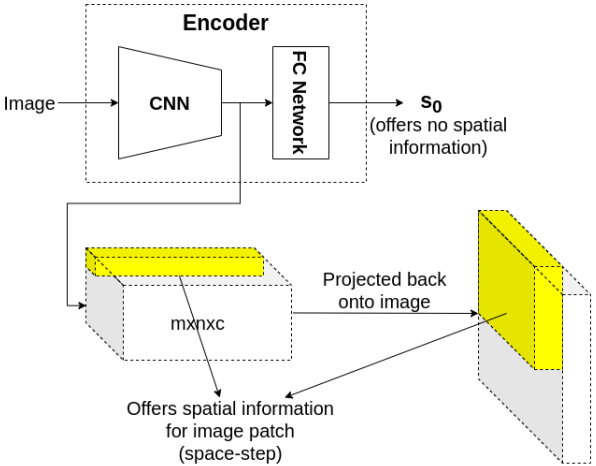
Attention Mechanism: Spatial Data



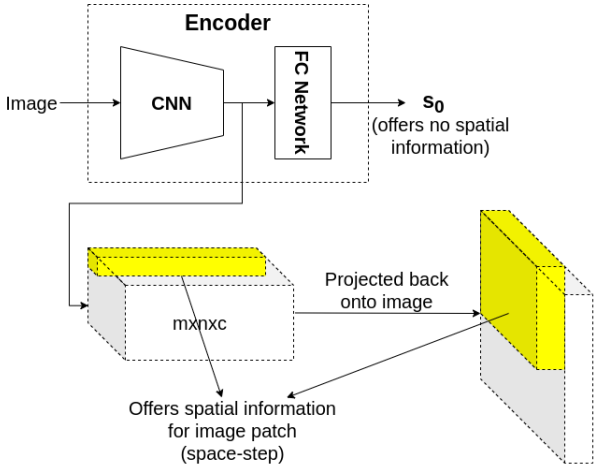
Attention Mechanism: Spatial Data



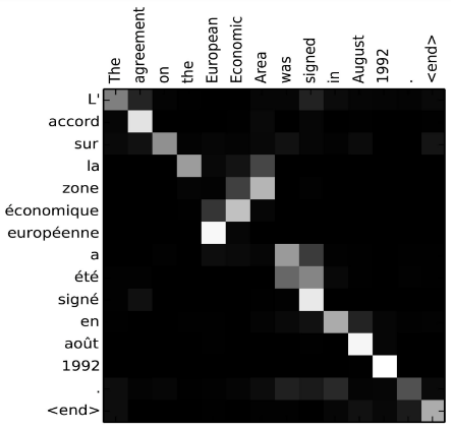
Attention Mechanism: Spatial Data



Attention Mechanism: Spatial Data



Byproduct of Attention: Explainability



English-to-French Translation

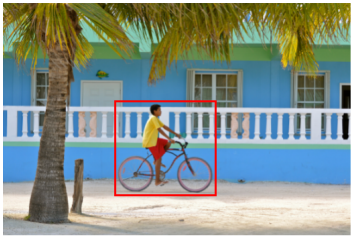
Credit: *Bahdanau et al, Neural Machine Translation by jointly learning to align and translate, ICLR 2015*

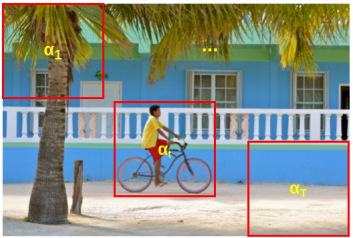
Byproduct of Attention: Explainability

Credit: Xu et al, Show, Attend and Tell: Neural Image Caption Generation with Visual Attention, ICML 2015

Modes of Attention

Hard vs Soft Attention:





- Single position is chosen for full alignment (1.0)
- Position-choosing is stochastic and hence non-differentiable

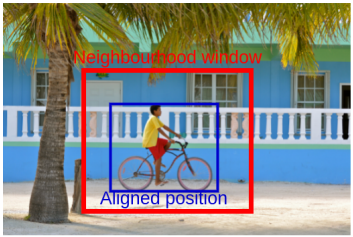
- All positions get partial alignment weights (0-1)
- Deterministic and hence differentiable, as no position-choosing

Modes of Attention

Global vs Local Attention:

What is the boy doing?

All input positions are chosen for attention

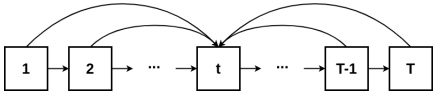


What is the boy doing?

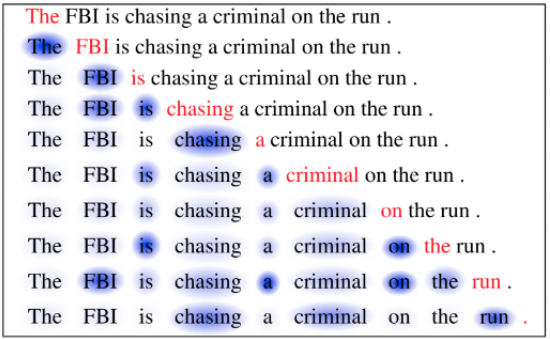
Neighbourhood of aligned position is chosen for attention

Modes of Attention

Self-Attention:



- Also known as intra attention
- Used extensively in advanced attention models (will see more soon)



Credit: *Cheng et al, Long Short-Term Memory-Networks for Machine Reading, ACL 2016*

Homework

Readings

- Lilian Weng, Attention? Attention!, Github Blog

Exercise

- What is the connection between an autoencoder and Principal Component Analysis (PCA)?

References

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate". In: *arXiv preprint arXiv:1409.0473* (2014).

[2] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. "Effective approaches to attention-based neural machine translation". In: *arXiv preprint arXiv:1508.04025* (2015).

[3] Kelvin Xu et al. "Show, attend and tell: Neural image caption generation with visual attention". In: *International conference on machine learning*. 2015, pp. 2048–2057.

[4] Jianpeng Cheng, Li Dong, and Mirella Lapata. "Long short-term memory-networks for machine reading". In: *arXiv preprint arXiv:1601.06733* (2016).

[5] Lilian Weng. *Attention? Attention!* 2018. URL: https://lilianweng.github.io/lil-log/2018/06/24/attention-attention.html.