Module 2 :

Deep Networks

Sequence Modeling

Géraldine Schaller-Conti, August 2024

Discussion Session

- Review of Notebooks 6.1 and 6.2:
  - Computer vision (6.1) :
    - Convolution, padding, pooling layer
    - Using pretrained model (Resnet-50)
  - Classification (6.2) :
    - Transfer learning with Tensorflow Hub

Bibliography

- Deep Learning book (Goodfellow, Bengio, Courville)
- Machine Learning @ Stanford (Prof Andrew Ng)
- Hands-On Machine Learning with Scikit-Learn & Tensorflow (Aurélien Géron)

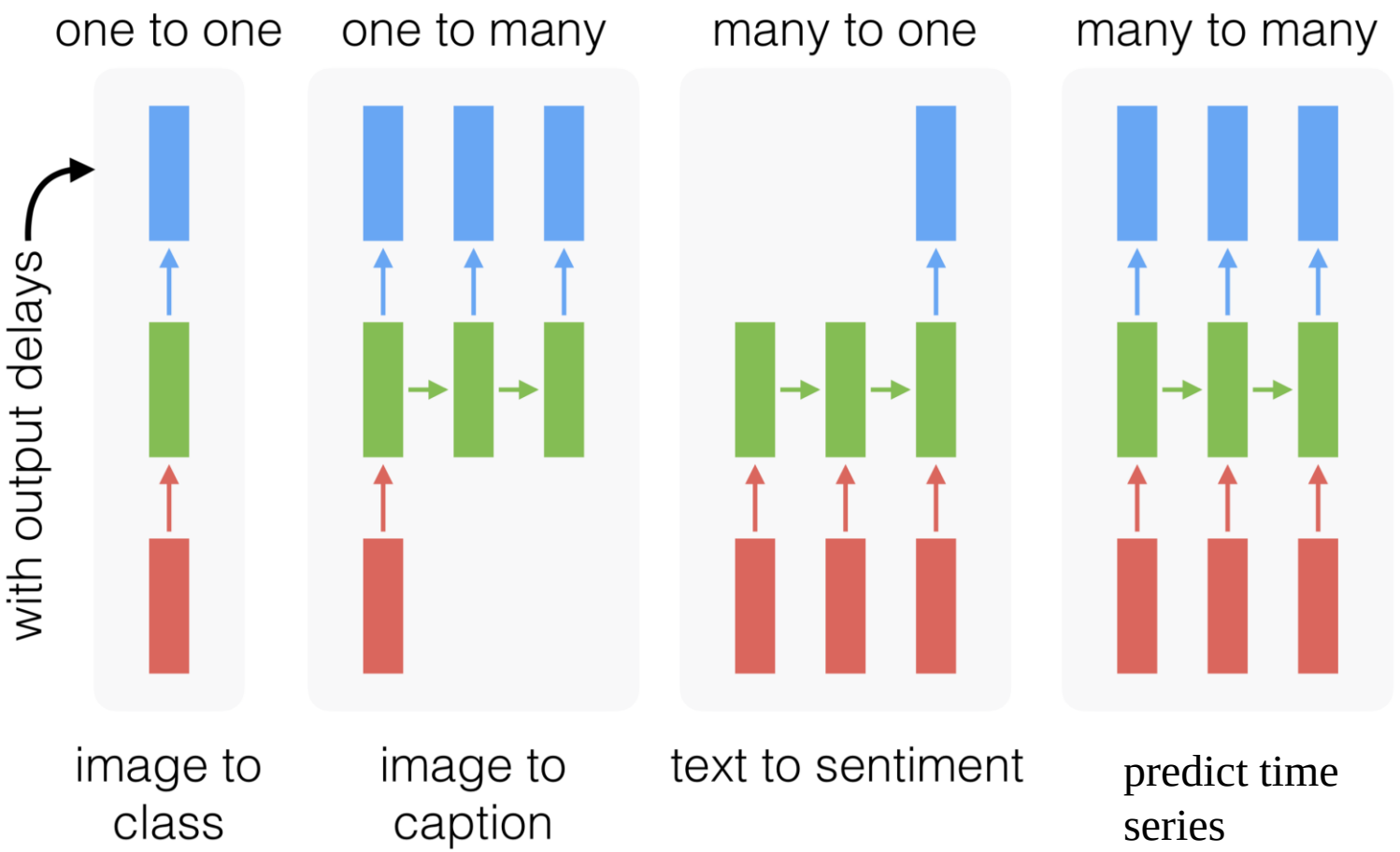
Learning Objectives

1. Introduction to Sequences
2. Recurrent Neural Networks (RNNs)
3. Text generation example
4. Natural Language Processing and word representation
5. Machine Translation example
6. Image captioning example

1. Introduction to sequences

- Input/output sequences
- Use cases

Input and Output Sequences

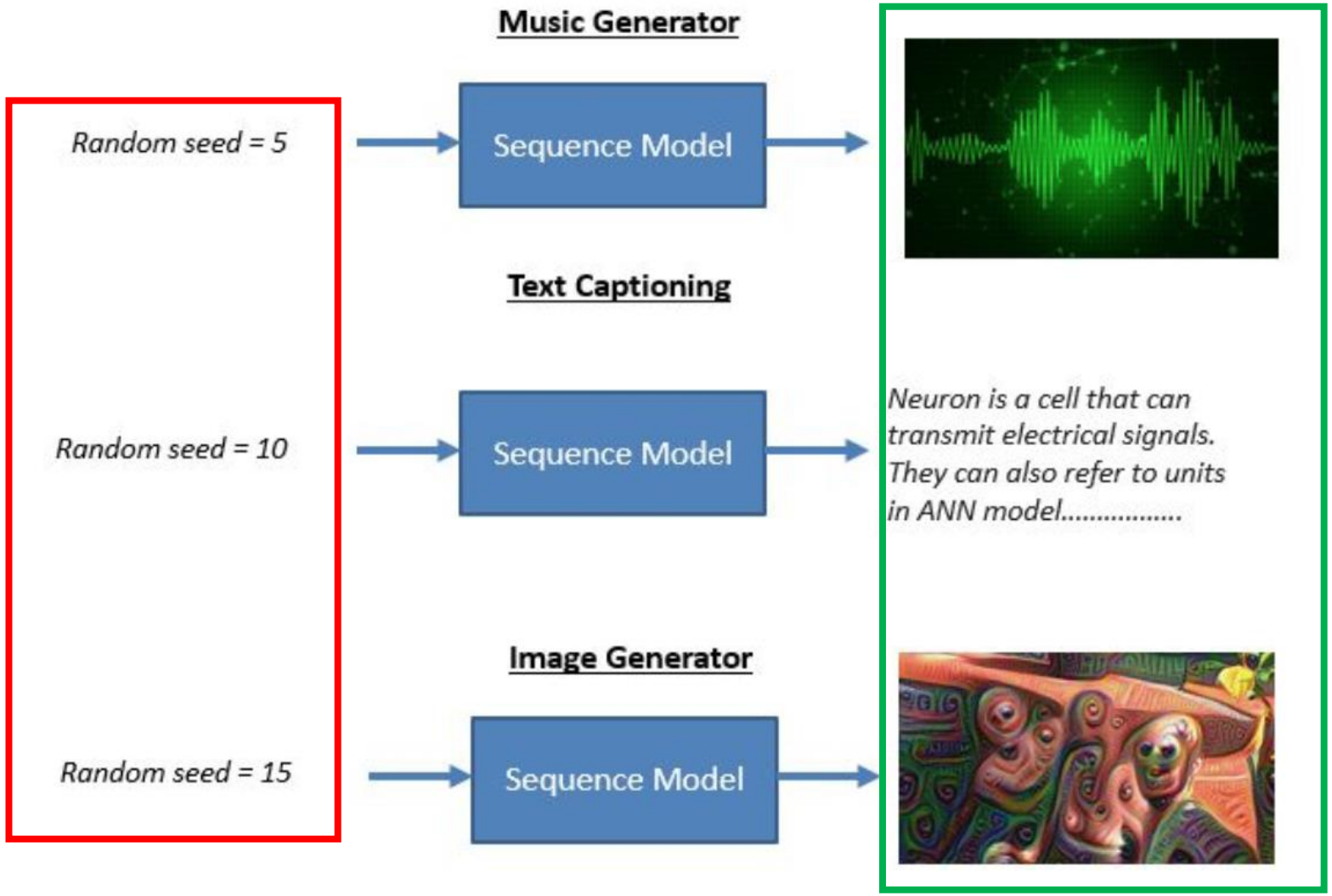


Use Cases

Input		Target		Use Cases
Type	Elements	Type	Elements	
Scalar	One	Trends	Many	Pattern generation
		Audio	Many	Music Generation
		Text	Many	Text Generation
		Image	Many	Image generation
Trends	Many	Scalar	One	Stock Trading decisions
				Forecasting KPI for fixed duration
		Trends	Many	DNA Sequence analysis
				Time series forecasts
Text	Many	Scalar	One	Sentiment Classification
				Topic Classification
				Answer Selection
		Text	Many	Text Summarization
				Machine translation
				Chatbots
				Name Entity Recognition
				Subject Extraction
				Part of Speech Tagging
				Textual Entailment
				Relation Classification
		Trends	Many	Path Query Answering
		Audio	Many	Speech Generation
Image	Many	Scalar	One	Facial expression tagging
				Entity classification
		Text	Many	Image Captioning
		Image	Many	Image Modification
Audio	Many	Scalar	One	Sentiment Classification
				Number of speaker tagging
				Topic Classification
		Text	Many	Speech Recognition
				Conference Summarization
		Audio	Many	Speech Assistant
Video	Many	Scalar	One	Activity Recognition
		Text	Many	Subtitle generation

Scalar → X

Input		Target		Use Cases
Type	Elements	Type	Elements	
Scalar	One	Trends	Many	Pattern generation
		Audio	Many	Music Generation
		Text	Many	Text Generation
		Image	Many	Image generation



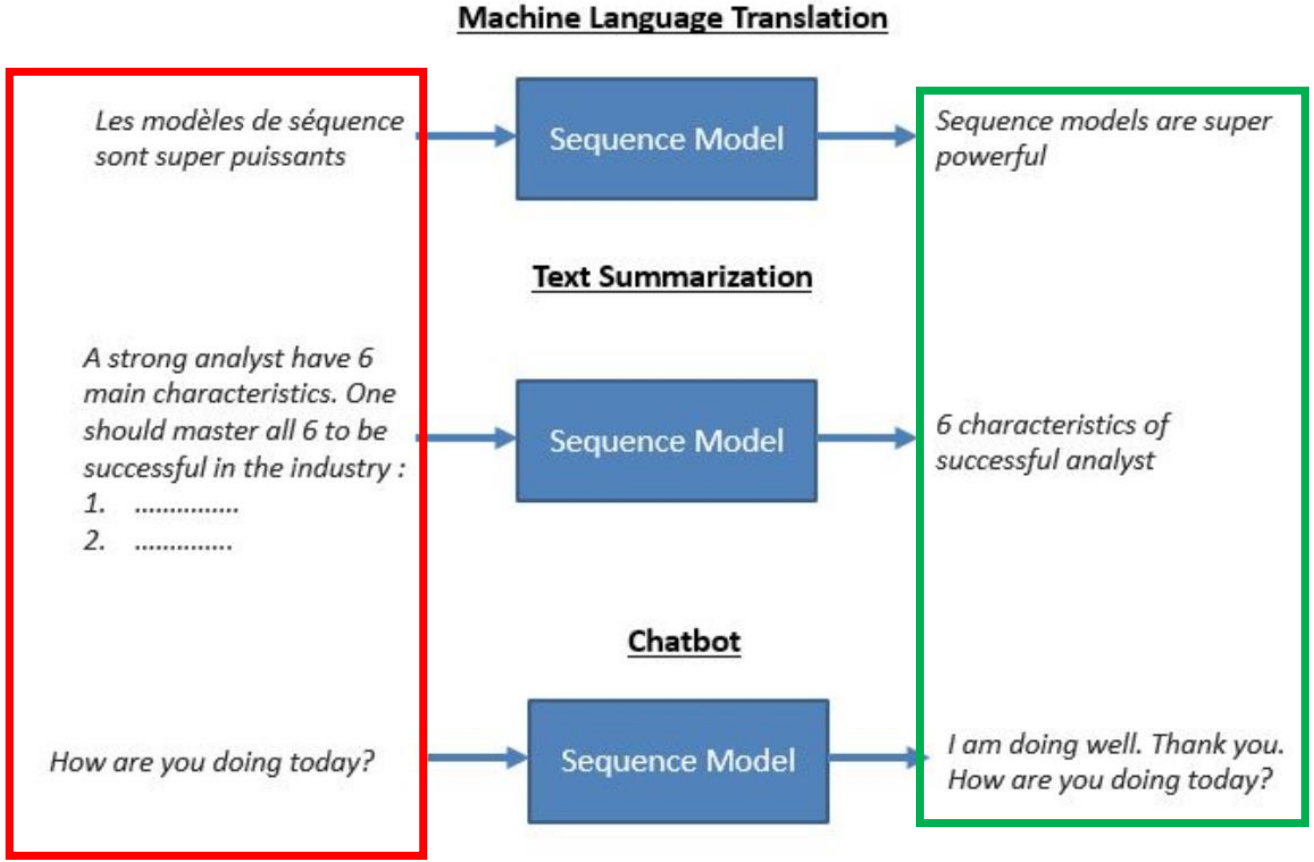
Text → Scalar

Input		Target		Use Cases
Type	Elements	Type	Elements	
Text	Many	Scalar	One	Sentiment Classification
				Topic Classification
				Answer Selection
		Text	Many	Text Summarization
				Machine translation
				Chatbots
				Name Entity Recognition
				Subject Extraction
				Part of Speech Tagging
				Textual Entailment
				Relation Classification
		Trends	Many	Path Query Answering
		Audio	Many	Speech Generation

Sentiment	Tweets
Negative	@united is the worst. Nonrefundable First class tickets? Oh because when you select Global/FC their system auto selects economy w/upgrade.
	@united I will not be flying you again
Neutral	@VirginAmerica my drivers license is expired by a little over a month. Can I fly Friday morning using my expired license?
	@VirginAmerica any plans to start flying direct from DAL to LAS?
Positive	@VirginAmerica done! Thank you for the quick response, apparently faster than sitting on hold ;)
	@united I appreciate your efforts getting me home!

Text → Text

Input		Target		Use Cases
Type	Elements	Type	Elements	
Text	Many	Scalar	One	Sentiment Classification
				Topic Classification
				Answer Selection
		Text	Many	Text Summarization
				Machine translation
				Chatbots
				Name Entity Recognition
				Subject Extraction
				Part of Speech Tagging
				Textual Entailment
				Relation Classification
		Trends	Many	Path Query Answering
		Audio	Many	Speech Generation



X → Text

Input		Target		Use Cases
Type	Elements	Type	Elements	
Image	Many	Scalar	One	Facial expression tagging
				Entity classification
		Text	Many	Image Captioning
		Image	Many	Image Modification
Audio	Many	Scalar	One	Sentiment Classification
				Number of speaker tagging
				Topic Classification
		Text	Many	Speech Recognition
				Conference Summarization
		Audio	Many	Speech Assistant
Video	Many	Scalar	One	Activity Recognition
		Text	Many	Subtitle generation

Speech Recognition

Sequence Model → *I love oranges*

Image Captioning

Sequence Model → *Two dogs are playing with a ball*

Subtitle Generator

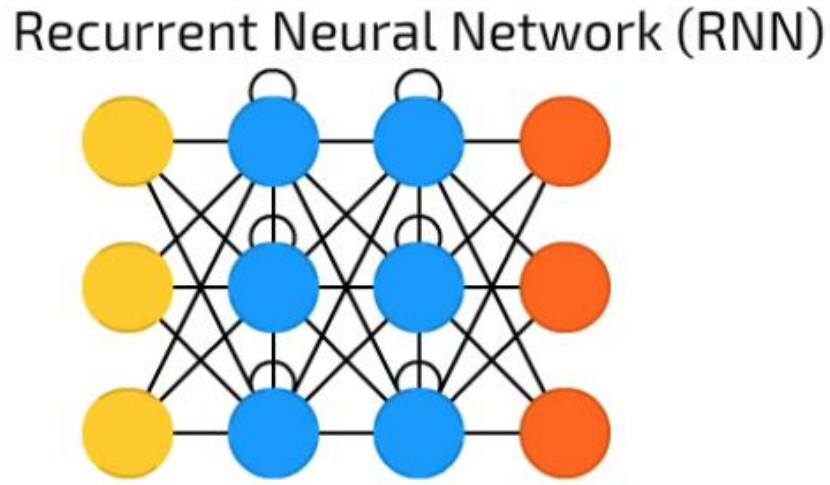
Sequence Model → *How you doin?*

2. RNNs

- Introduction
- Training
- Memory cells
- Deep RNNs
- Bidirectional RNNs
- Example: Time Series

Introduction

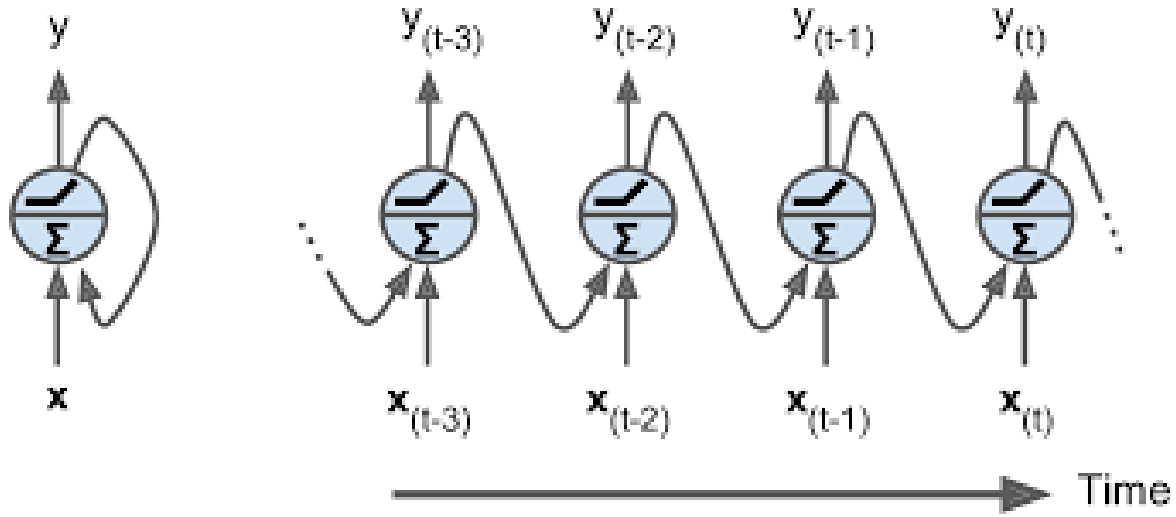
- RNN is a class of nets that can work on sequences of arbitrary lengths to predict the future
- Connections between neurons include loops
- **Recurrent cells** (or memory cells) used
  - Weight sharing between *time-steps*



Recurrent Neurons

- At each time step t, the recurrent neuron receives the inputs $x_{(t)}$ as well as its own output from the previous time steps $y_{(t-1)}$

- Unrolling the network through time



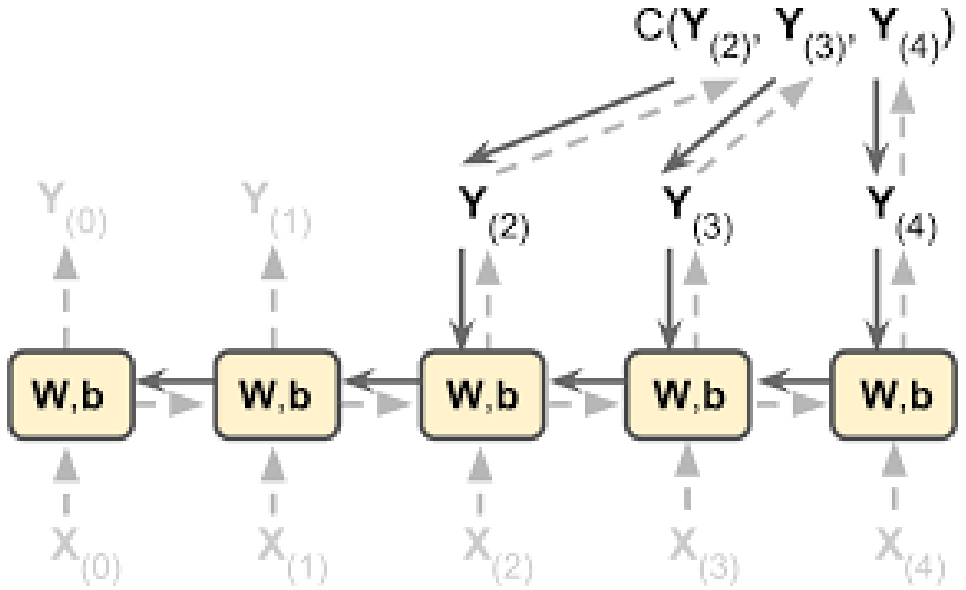
- Output at time step t is a function of previous states and current input → *memory*

$$y_t = \varphi(x_t^T \cdot w_x + y_{t-1}^T \cdot w_y + b)$$

  - Two sets of weights w_x and w_y

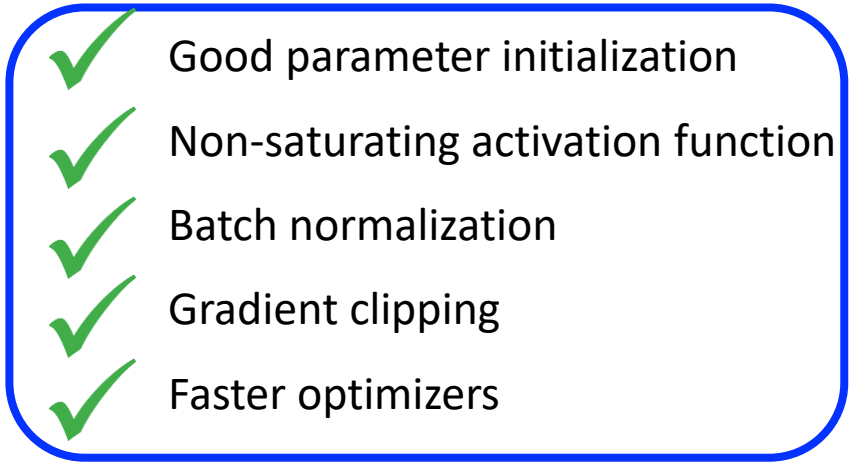
Training Loop

- Trick is to unroll it through time and use regular backpropagation (backpropagation through time BPTT)
  - First forward pass through the unrolled network
  - output sequence evaluated using a cost function
  - Gradients of the cost function are propagated backward through the unrolled network
  - Model parameters are updated using computed gradients



Training over many time steps

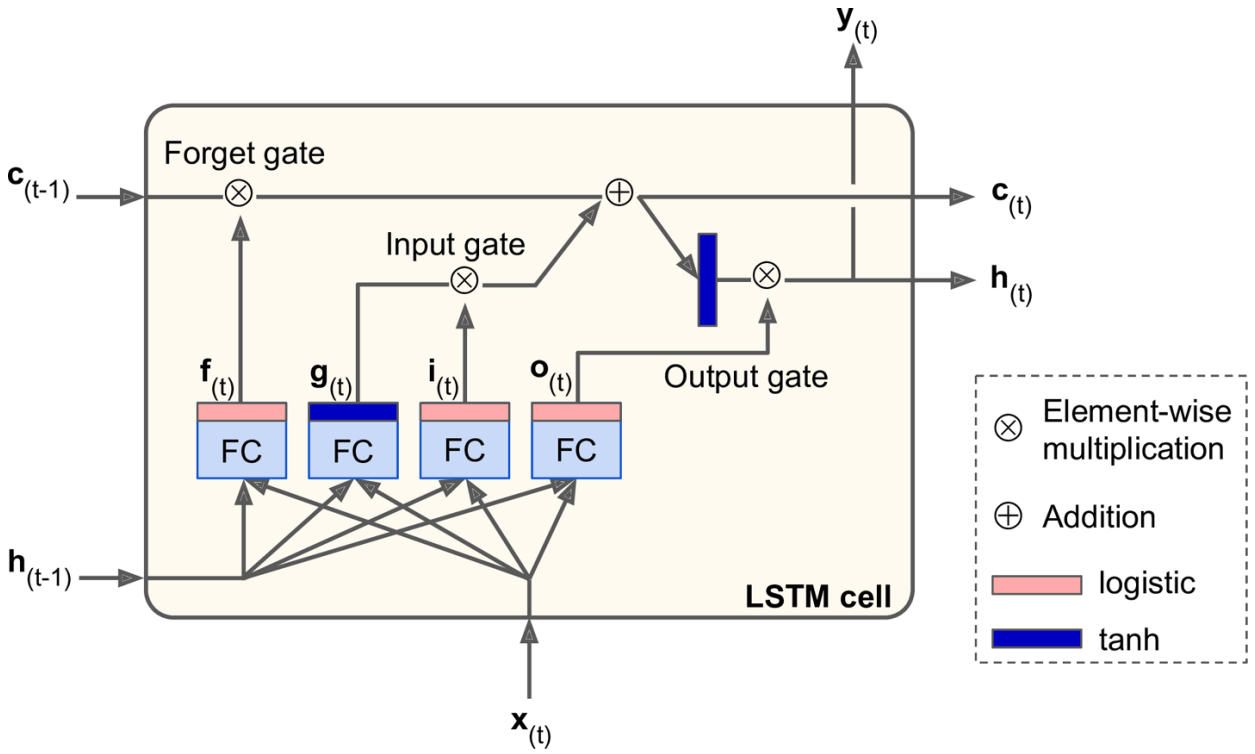
- Many time steps needed to train a RNN on long sequences
- Vanishing/exploding gradients



- One solution : truncated backpropagation through time, ie unroll the RNN only over a limited number of time steps during training
- Limits :
  - Missing part of crucial data in your training sample (specific events/dates,…)
  - Memory of the first inputs gradually fades away

Memory Cell: LSTM

- Long Short-Term Memory (LSTM) cell (1997)
  - Converges faster and detects long-term dependencies in the data

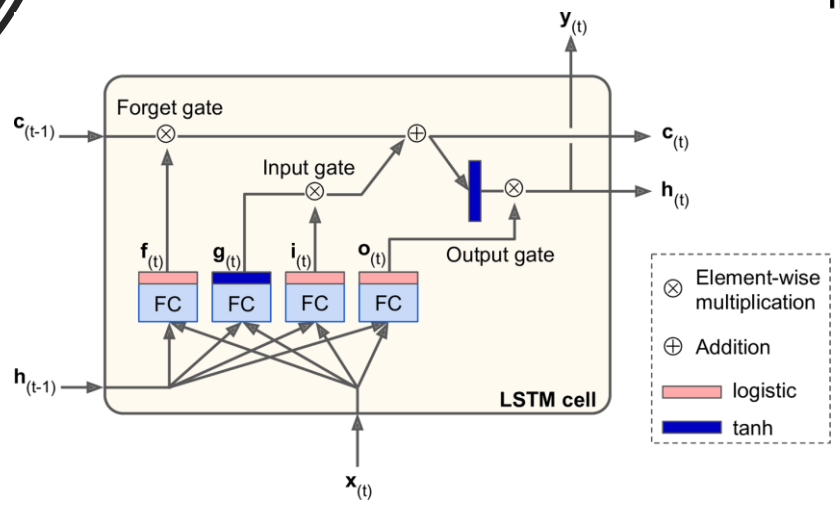


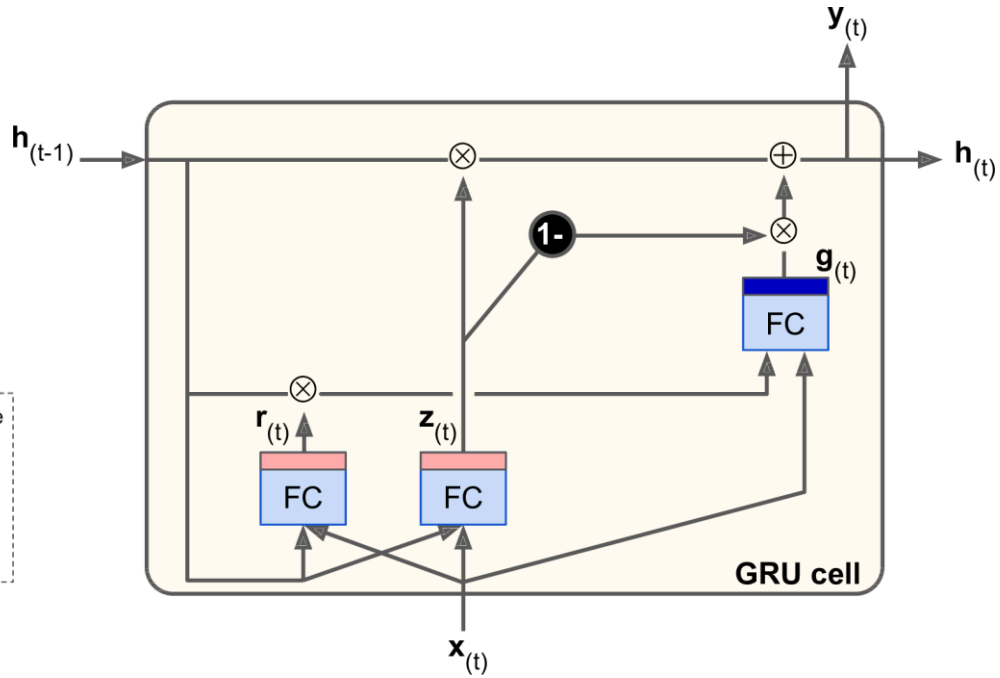
- Cell state is split in two vectors h_t (short-term state) and c_t (long-term state)

- Network that can learn what to store in the long-term state, what to throw away, and what to read from it

Memory Cell: GRU

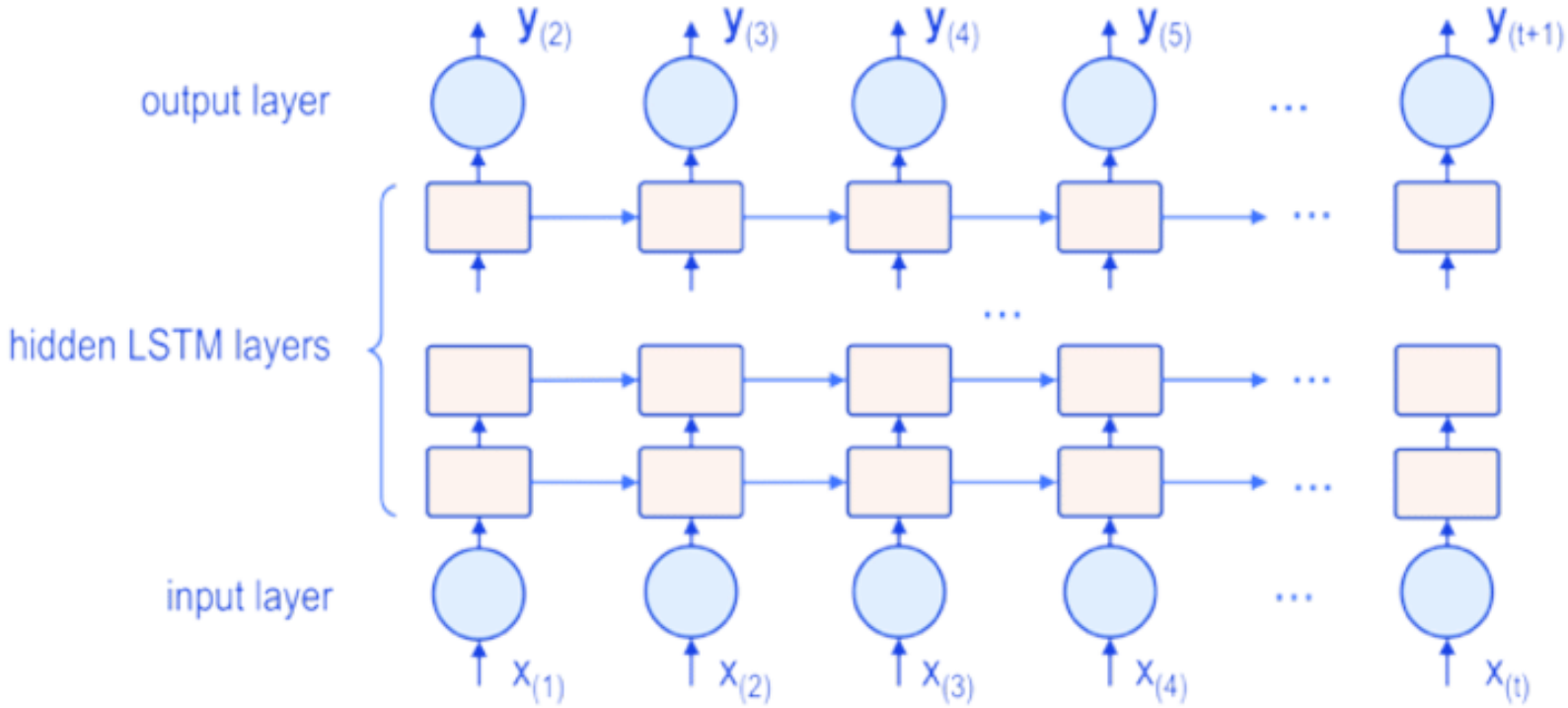
- Gated Recurrent Unit (GRU) cell is a simplified version of the LSTM cell
- Both state vectors are merged into a single vector h_t





Deep RNNs

- Stack multiple layers of cells

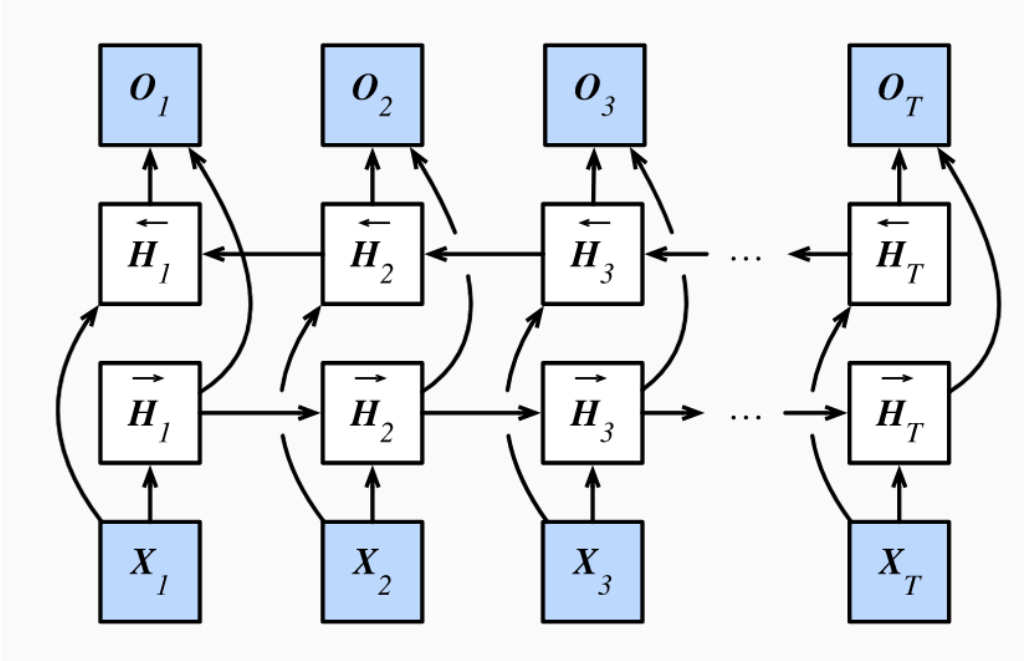


- Apply dropout to reduce overfitting

Bidirectional RNN

- A RNN limitation is that it only uses information from earlier in the sequence and not after

- Solution : bidirectional RNN



- Bidirectional training provides a deeper sense of language context (learns the context of a word based on all of its left and right surroundings)

Example : Time Series (1)

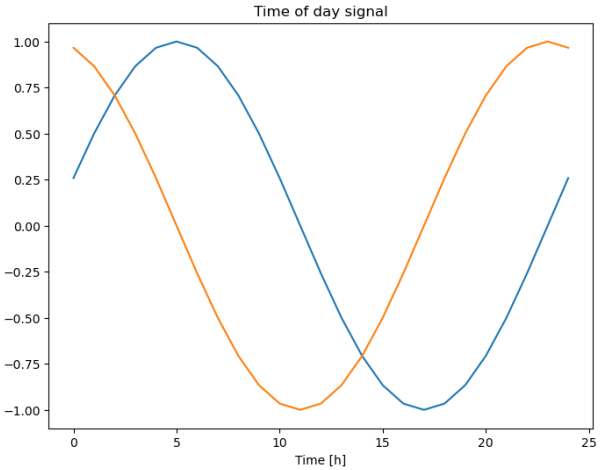
- Input data :
  - Stationarity : statistical properties do not change with time (constant mean, variance,…)
  - RNNs and LSTMs should be able to model non-stationary time series
  - Could be helpful to use stationary data to ease the training OR if the sample size is small
    - Preprocessing : apply transformations like seasonal decomposition to make them stationary

Example : Time Series (2)

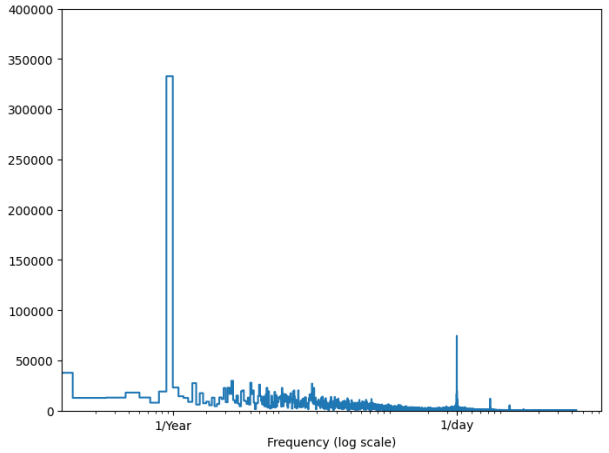
- Trigonometric encoding : encode periodicity or seasonality in time series data (daily, yearly,…)
  - Weather data

```
day = 24*60*60
year = (365.2425)*day

df['Day sin'] = np.sin(timestamp_s * (2 * np.pi / day))
df['Day cos'] = np.cos(timestamp_s * (2 * np.pi / day))
df['Year sin'] = np.sin(timestamp_s * (2 * np.pi / year))
df['Year cos'] = np.cos(timestamp_s * (2 * np.pi / year))
```

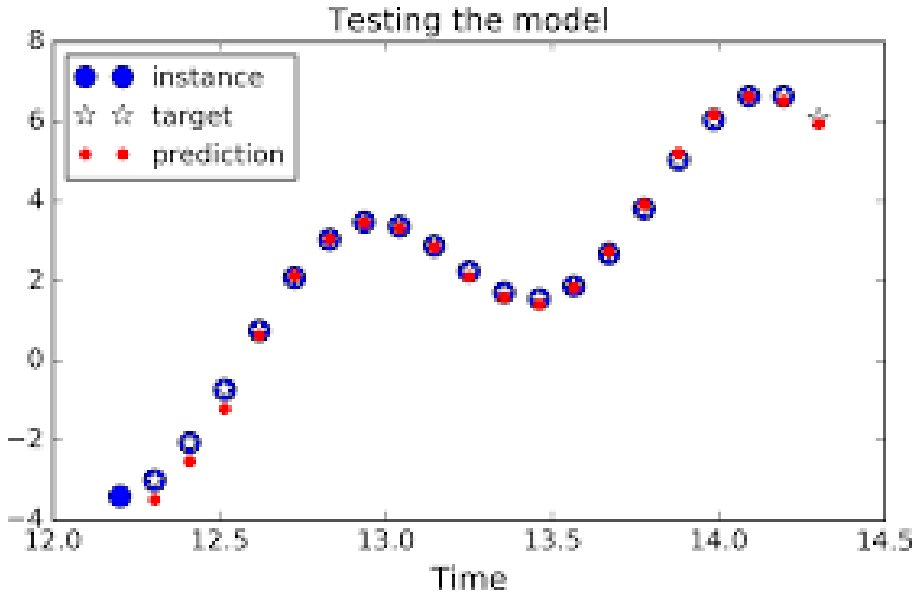


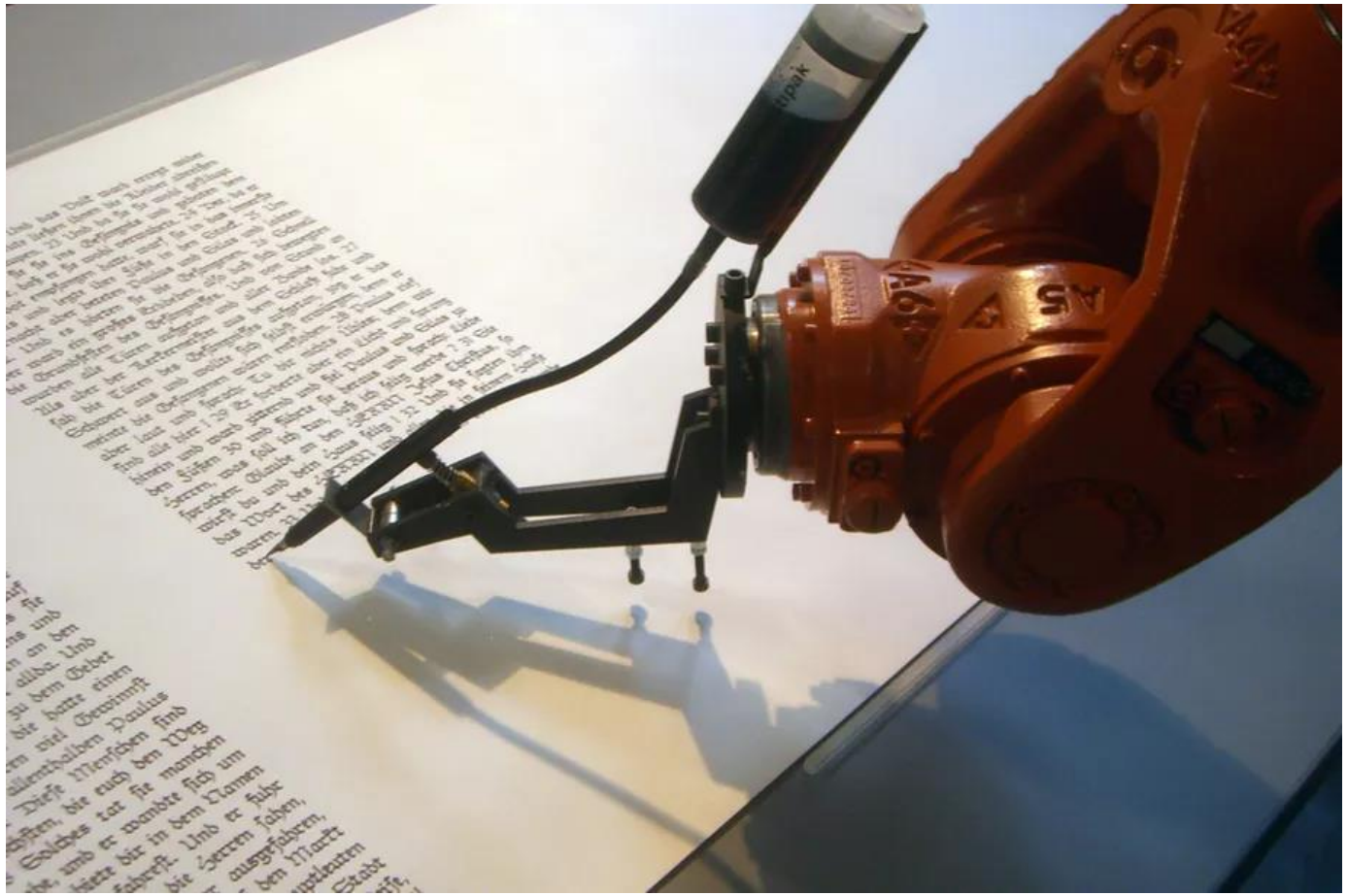
- To find the important frequencies : extract features with Fast Fourier Transform
  - Example : temperature data



Example : Time Series (3)

- Training instance : randomly selected sequence of 20 consecutive values from the time series
  - In principle, have several input features
- Target sequence : similar as the input sequence, except it is shifted by one time step into the future





3. Text Generation example

Task

- Given a character, or a sequence of characters, what is the most probable next character?

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.

The Sonnets,
W. Shakespeare

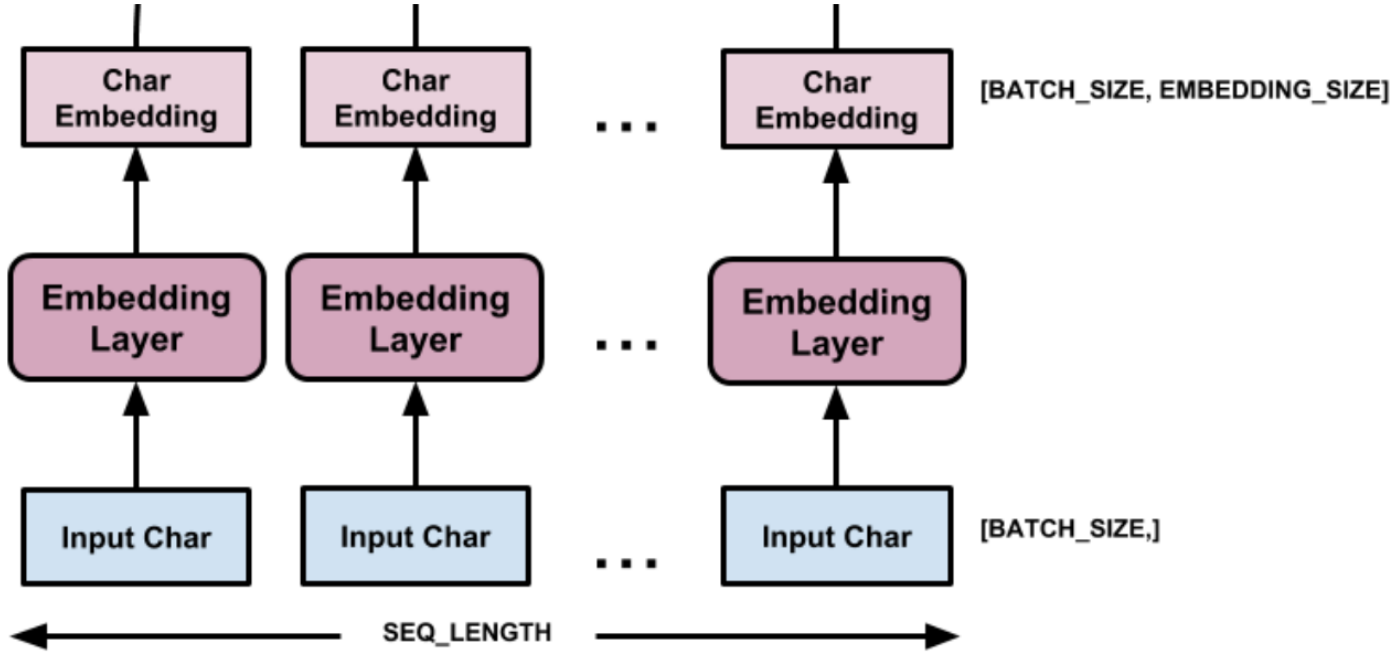
- For each sequence, the corresponding target contain the same length of text, except shifted one character to the right
  - Input sequence : “Hell”
  - Target sequence : “ello”

Input Data

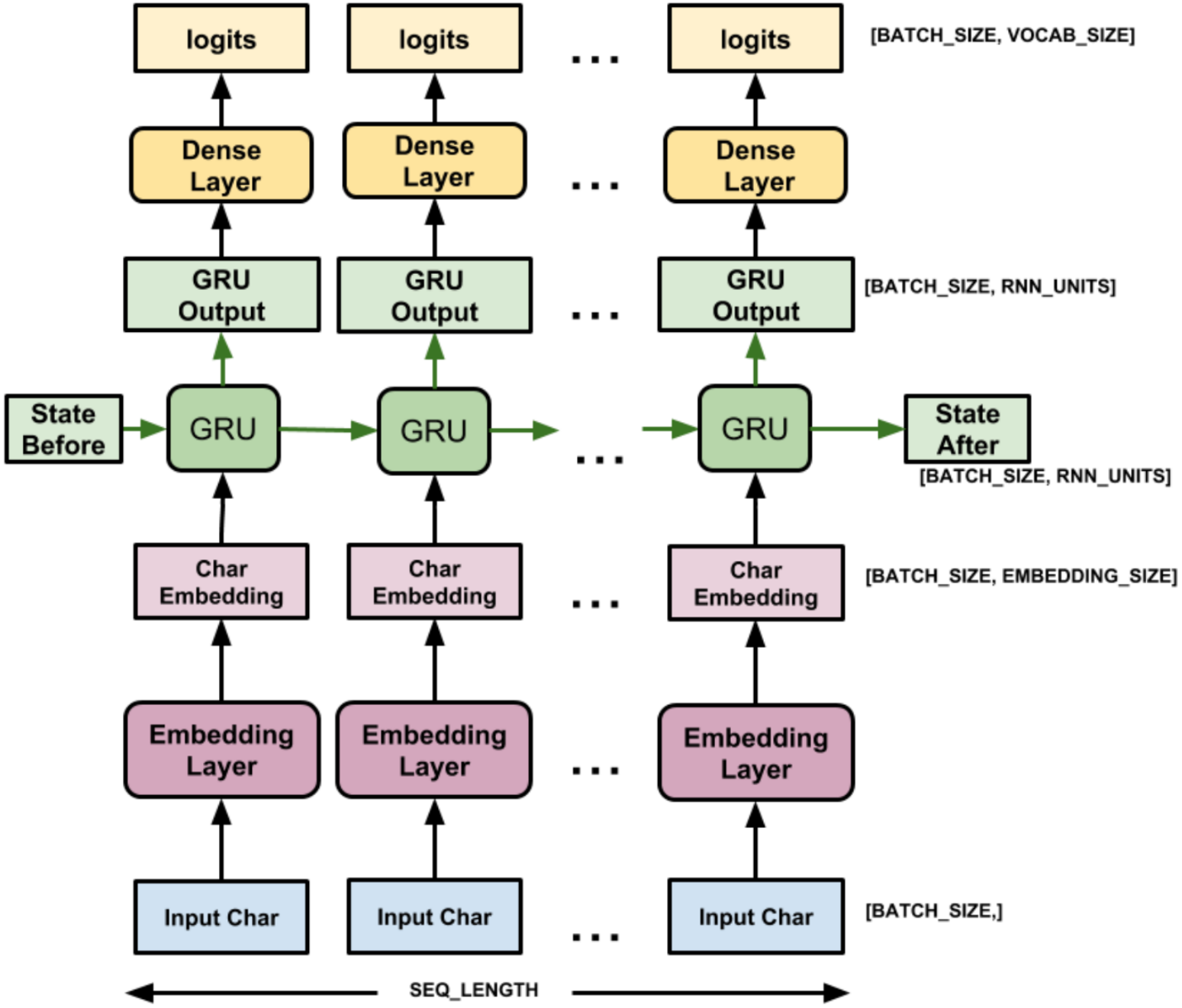
1) divide the text into sequences of given length

2) Char embedding :

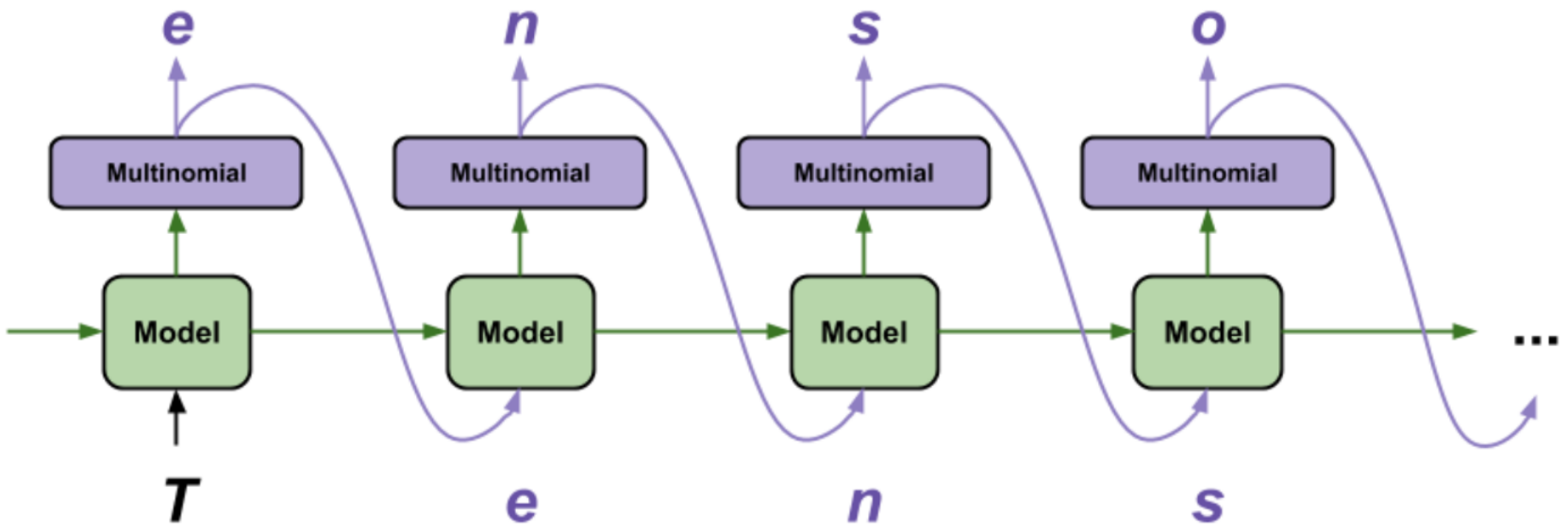
'First Citizen' → [18 47 56 57 58 1 15 47 58 47 64 43 52]



Training


logits
logits
logits
[BATCH_SIZE, VOCAB_SIZE]
Dense Layer
Dense Layer
Dense Layer
GRU Output
GRU Output
GRU Output
[BATCH_SIZE, RNN_UNITS]
State Before
GRU
GRU
GRU
State After
[BATCH_SIZE, RNN_UNITS]
Char Embedding
Char Embedding
Char Embedding
[BATCH_SIZE, EMBEDDING_SIZE]
Embedding Layer
Embedding Layer
Embedding Layer
Input Char
Input Char
Input Char
[BATCH_SIZE,]
SEQ_LENGTH

Inference


e
n
s
o
Multinomial
Multinomial
Multinomial
Multinomial
Model
Model
Model
Model
...
T
e
n
s

Result

```
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.
```

https://www.youtube.com/watch?v=Jkkjy7dVdaY

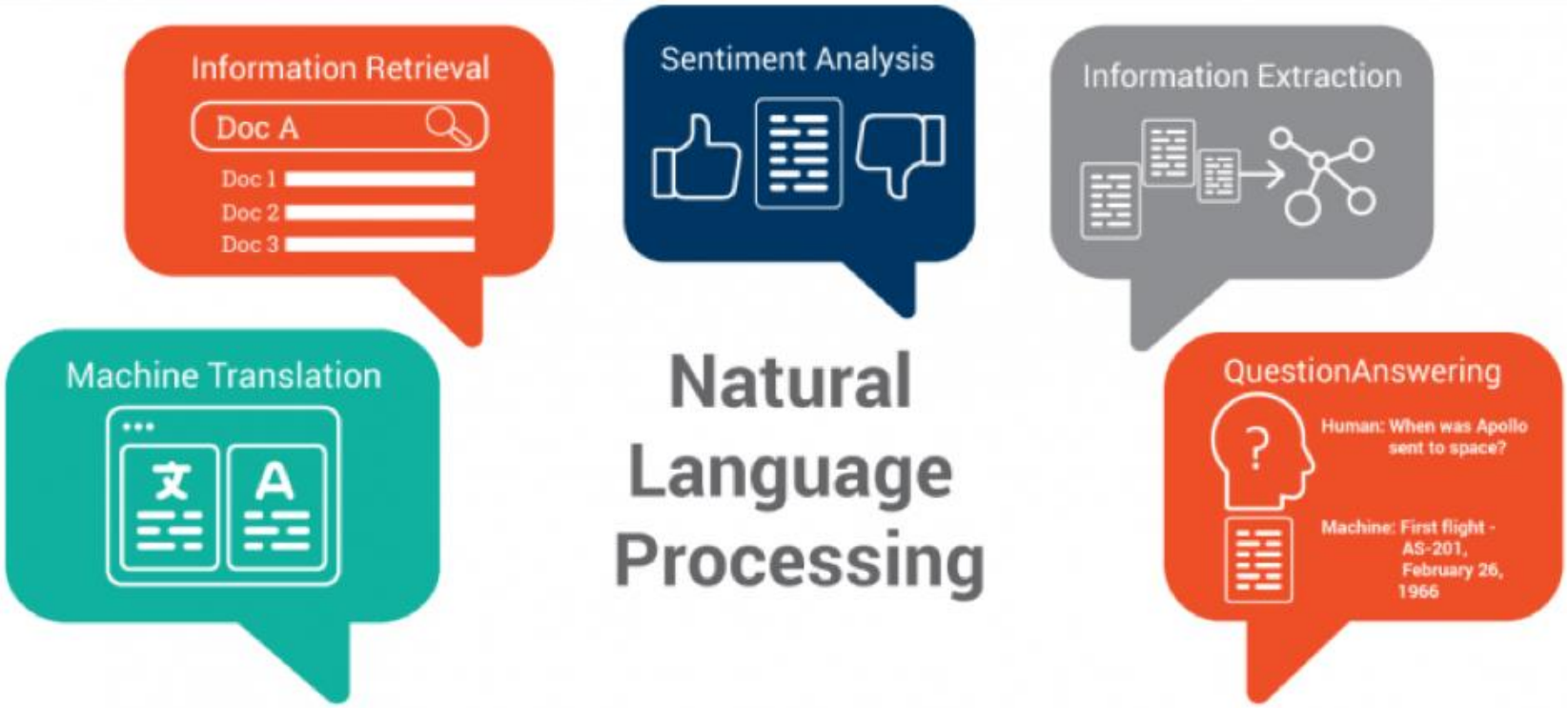
Two-Minute Papers

4. Natural Language Processing (NLP) and word representation

- one-hot encoding
- feature vectors
- word embedding

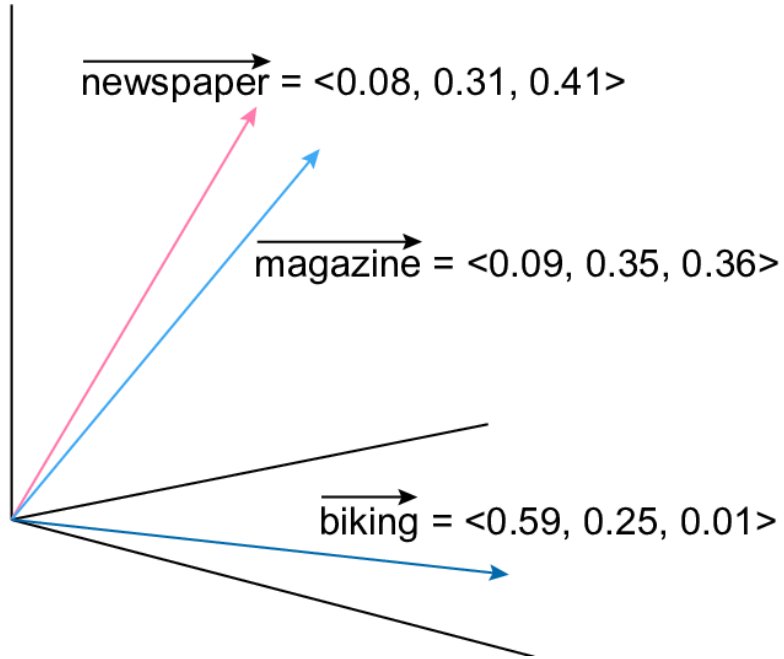
Natural Language Processing

- Deals with machine translation, automatic summarization, parsing, sentiment analysis,….



Word Embeddings

- Convert symbolic representations (words, dates, categories,…) into meaningful numbers
  - Capture the underlying semantic relations



- Several methods :
  1) Use a one-hot vector to represent words
  2) Feature vectors
  3) (Pre-trained) word embeddings

1) One-Hot Encoding

- Embed the colour “orange”

$$\text{Orange} = [1,0,0,0,0,0,\ldots]$$

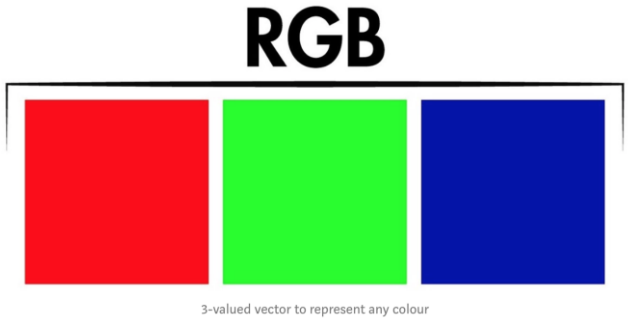
Each location in the vector represents a different colour

- Limitations :
  - Vectors can be prohibitively large
  - Assumption that there are no inherent relationships between any of the colours being embedded
    - Similarity between the vectors for “orange” and “red” will not be different to the similarity between the vectors for “orange” and “green”

2) Feature Vectors

- Vector able to represent any colour with only 3 values each time

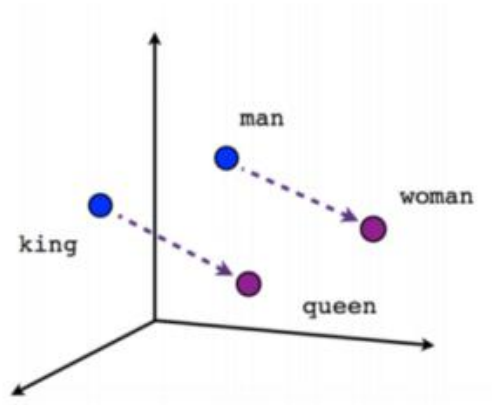
$$\textbf{Red} = [1,0,0]$$
$$\textbf{Green} = [0,1,0]$$
$$\textbf{Orange} = [1,0.5,0]$$



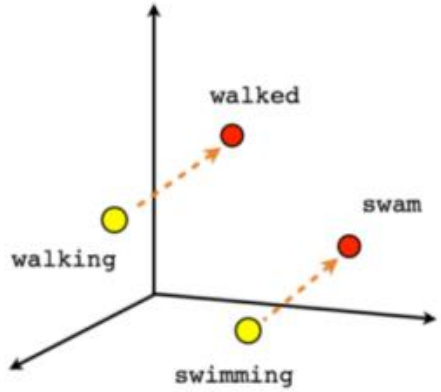
- Similarly, create fixed-length vectors that represent items like words (usually between 50 and 300 values)

3) Pre-trained Word Embeddings

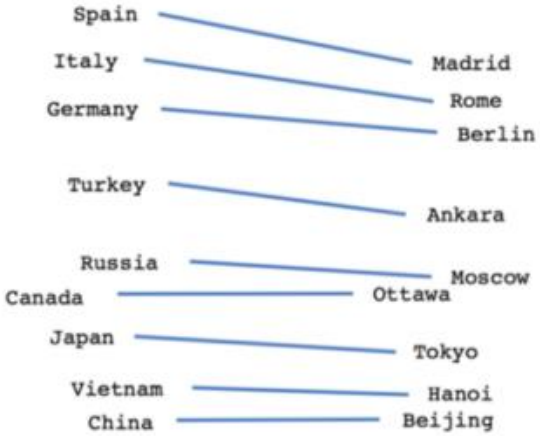
- Glove (2014) / Word2Vec (2014)



Male-Female

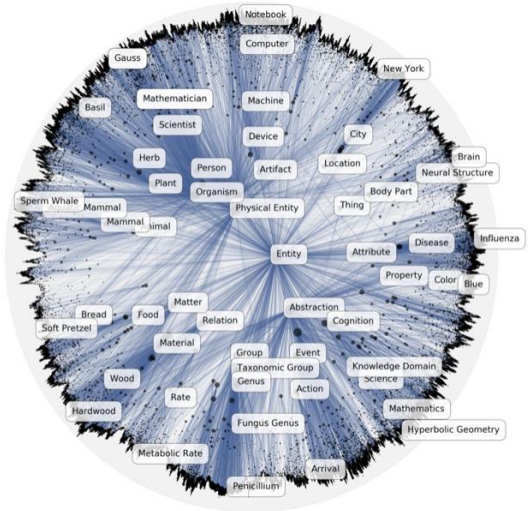


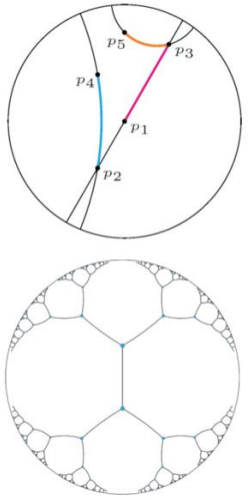
Verb tense

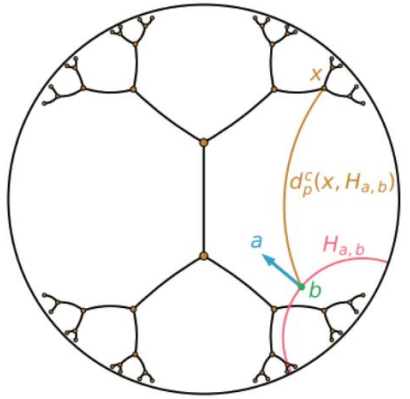


Country-Capital

- Poincaré







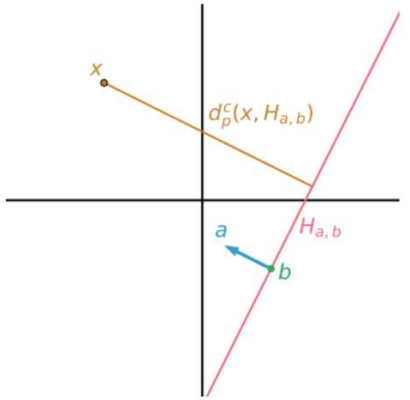
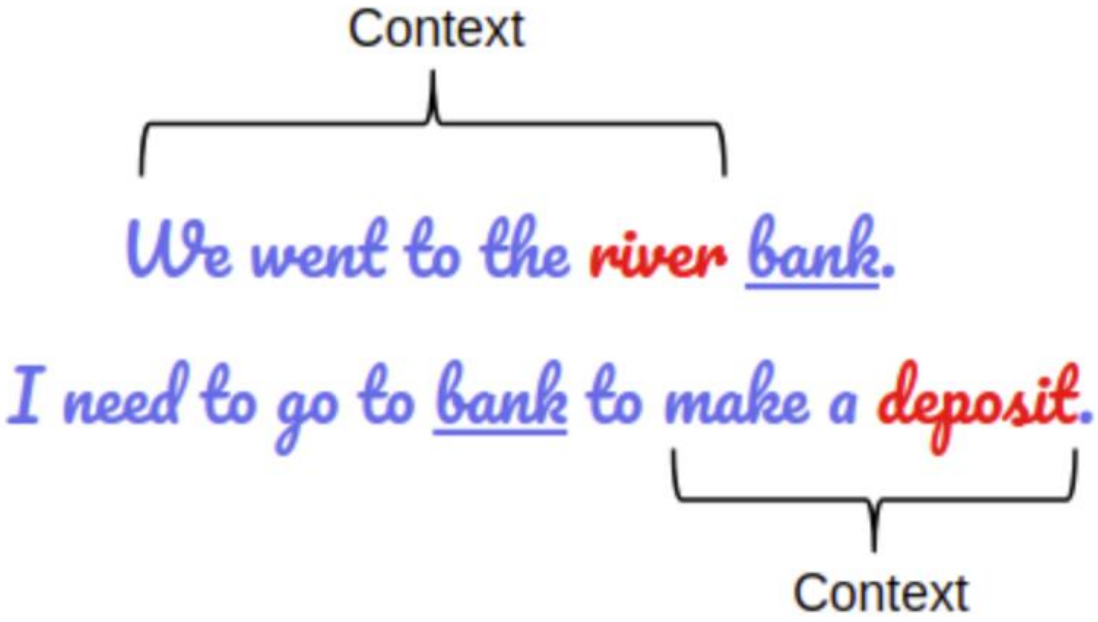


Figure 3: Illustration of an orthogonal projection on a hyperplane in a Poincaré disc $\mathbb{B}_c^2$ (Left) and an Euclidean plane (Right). Those hyperplanes are *decision boundaries*.

Limit of 2014 pre-trained word embeddings



- Do NOT take the context of the word into account
- Word2Vec will give the same vector for “bank” in both contexts

5. Machine Translation Example

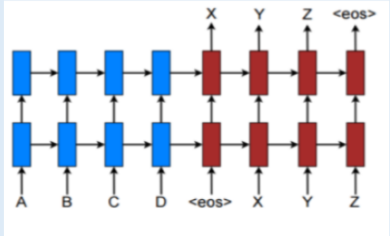
Machine Translation

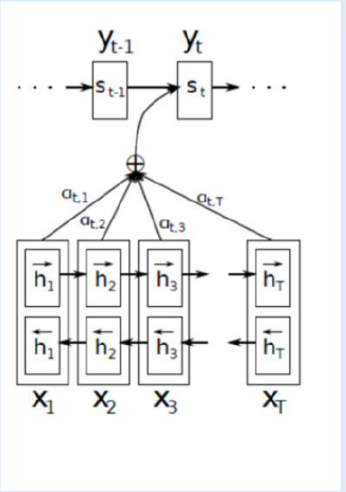
(*French*) *Si mon tonton tond ton tonton, ton tonton sera tondu.*

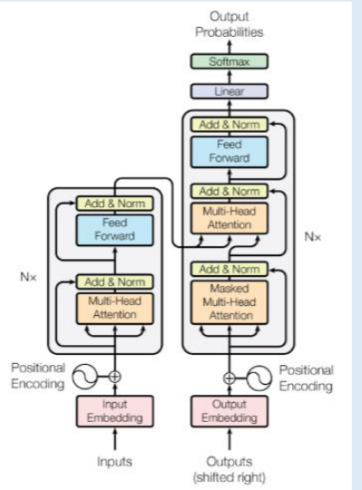
(*English*) *If my uncle shaves your uncle, your uncle will be shaved.*

$$\text{argmax}_{t \in \text{TARGET\_LANGUAGE}} P(t|s) =$$

$$\text{argmax}_{t \in \text{TARGET\_LANGUAGE}} P(s|t) \times P(t)$$







Phrase-based statistical MT with a translation model and a language model (Koehn et al. 2007)	RNN encoder-decoder with LSTM units (Sutskever / Cho et al. 2014)	RNN encoder-decoder with attention model (Bahdanau et al. 2015)	The Transformer: self-attention networks with positional encoding (Vaswani et al. 2017)

Courtesy Prof. A. Popescu

Time

RNN encoder-decoder (2014)

6) At each step, the decoder outputs a score for each word in the output vocabulary. The French word with highest probability (softmax layer) is output

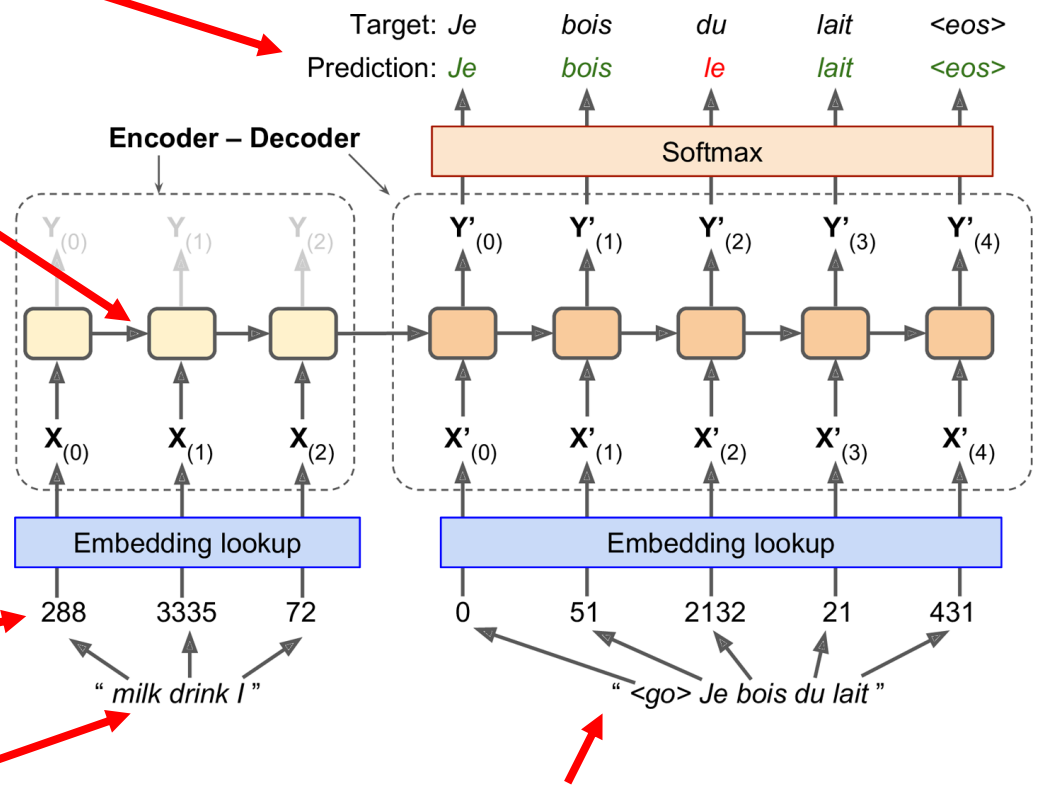
4) Hidden state used for the next input word

3) Word embedding that is fed to the encoder

2) Each word is initially represented by a simple integer identifier

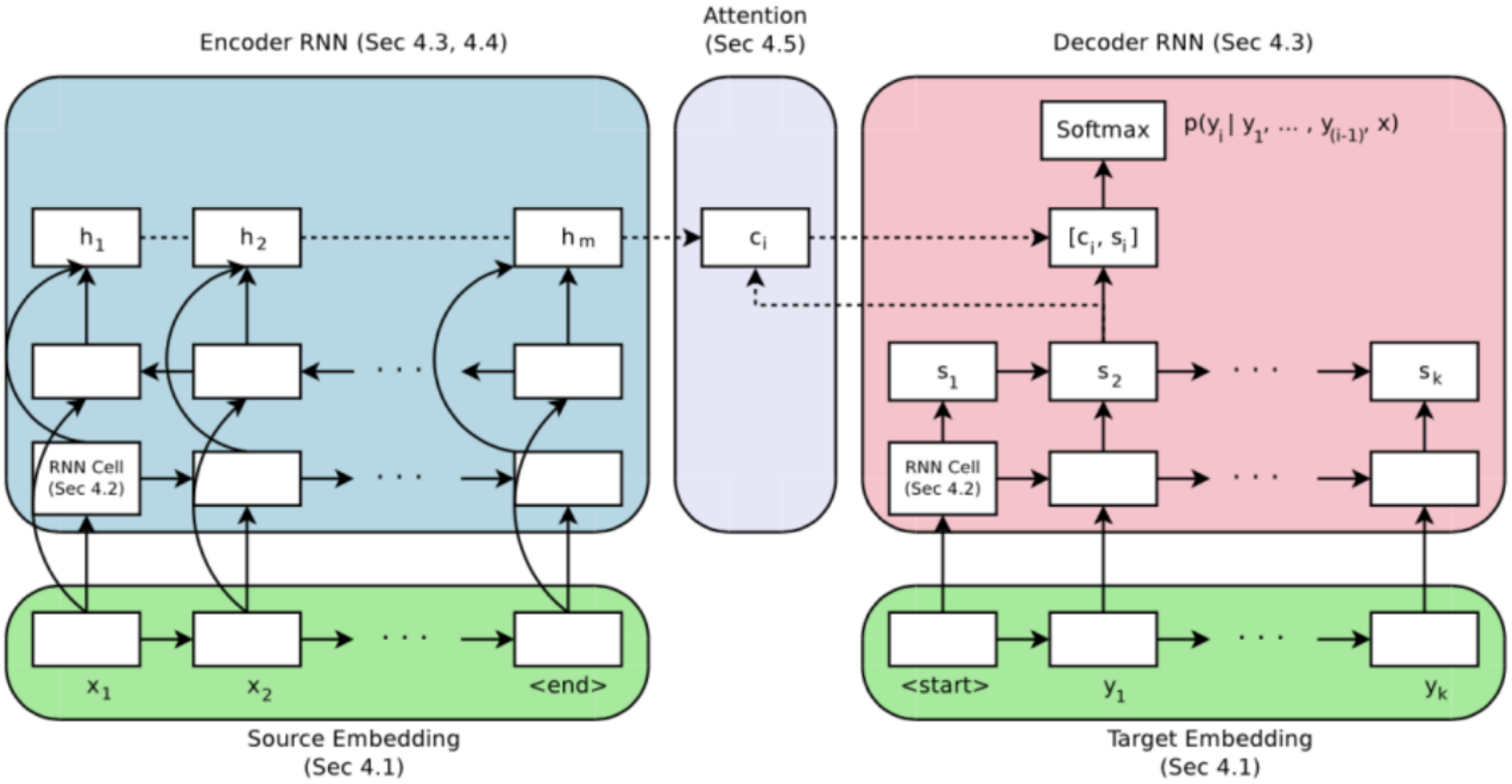
1) English sentences fed to the encoder

Why is the English sentence reversed ?



5) French translations also used as inputs to the decoder (shifted by one step)

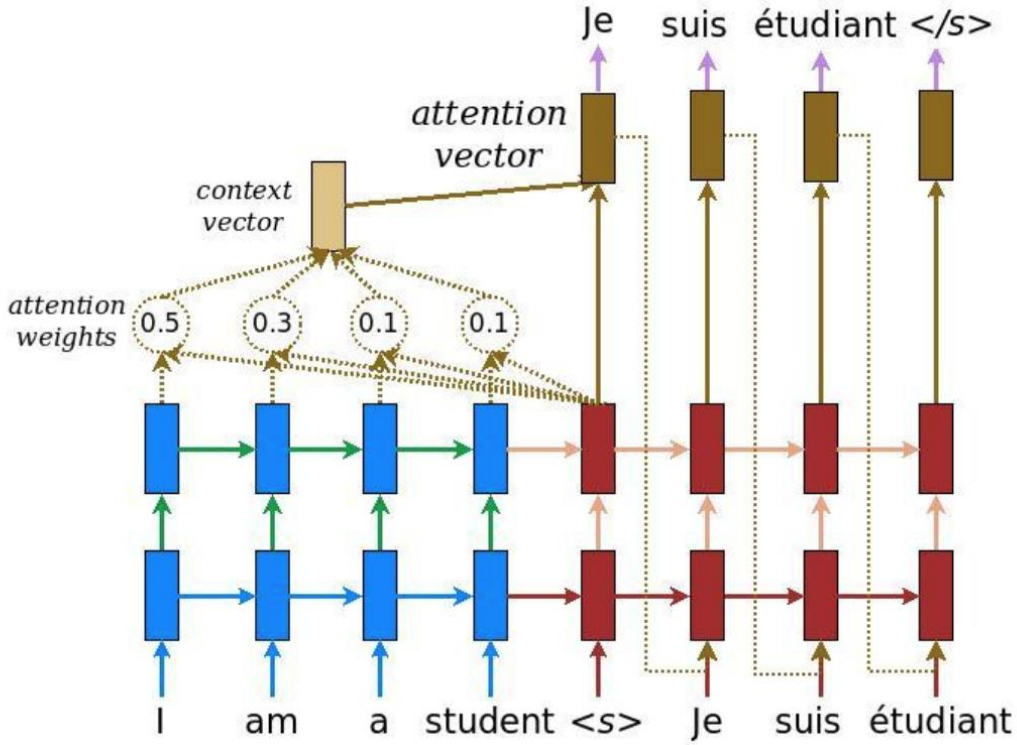
RNN encoder-decoder with attention (2015)



- Attention mechanism helps focusing on **specific parts** of the input

Attention Mechanism

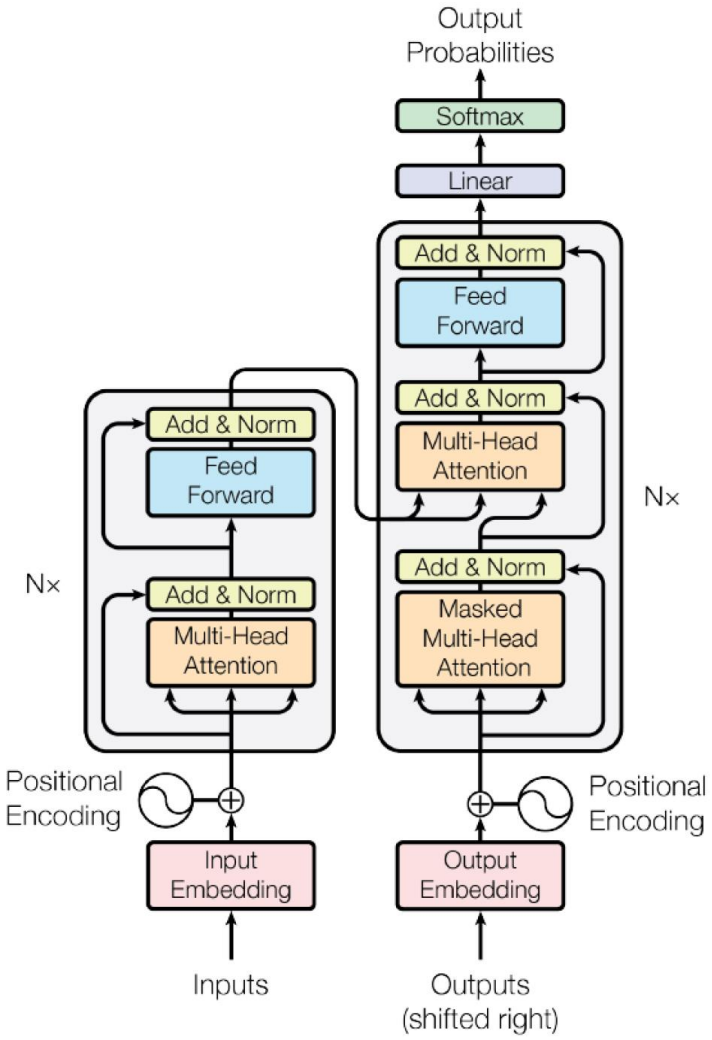
- Let the decoder learn to focus over a specific range of the input sequence



- Each input word is assigned a weight by the attention mechanism, which is then used by the decoder to predict the next word in the sentence

The Transformer (2017)

- Based on the attention mechanism *without recurrent sequential processing (unlike RNNs)*
  - does NOT require that the sequence be processed in order → Parallelization
  - All token processed at the same time and attention weights between them calculated
- Learns contextual relations between words in a text
- Building block of most state-of-the-art architectures in NLP
  - Machine translation
  - Text summarization
- Architecture : a set of encoders that are chained together and a set of decoders chained together



Pretrained Word Embeddings

Reading

- word2vec (Google)
  - O. Levy & Y. Goldberg, “Neural Word Embeddings as Implicit Matrix Factorization”, *NIPS*2014
- GloVe (Stanford)
  - J. Pennington, R. Socher, C. D. Manning, “GloVe: Global Vectors for Word Representation”, *EMNLP*2014
- fastText (Facebook)
  - P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, “Enriching Word Vectors with Sub word Information”, *TACL*2017
- ELMo (AllenNLP)
  - M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, “Deep contextualized word representations”, *NAACL*2018
- BERT (Google) Bidirectional Encoder Representation from Transformer
  - J. Devlin, M.W. Chang, K. Lee, K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", *Arxiv*oct. 2018

- Pre-trained on the entire Wikipedia (2500 million words) and Book Corpus (800 million words)
- can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of NLP tasks

A young boy is playing basketball.

Two dogs play in the grass.

A dog swims in the water.

A little girl in a pink shirt is swinging.

A group of people walking down a street.

A group of women dressed in formal attire.

Two children play in the water.

A dog jumps over a hurdle.

6. Image to Caption Example

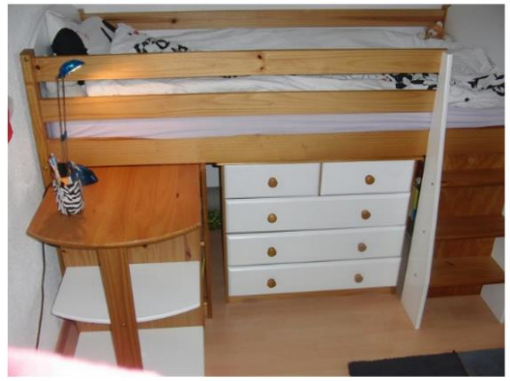
COCO Dataset

- Given an image, what is the most probable caption describing it ?
- Microsoft-COCO dataset (>82000 images)

The man at bat readies to swing at the pitch while the umpire looks on.

A large bus sitting next to a very tall building.

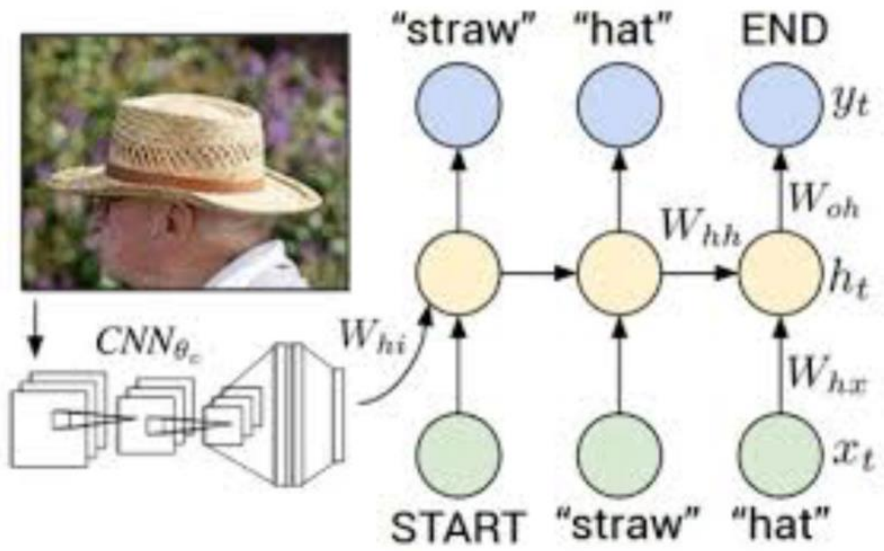
A horse carrying a large load of hay and two people sitting on it.

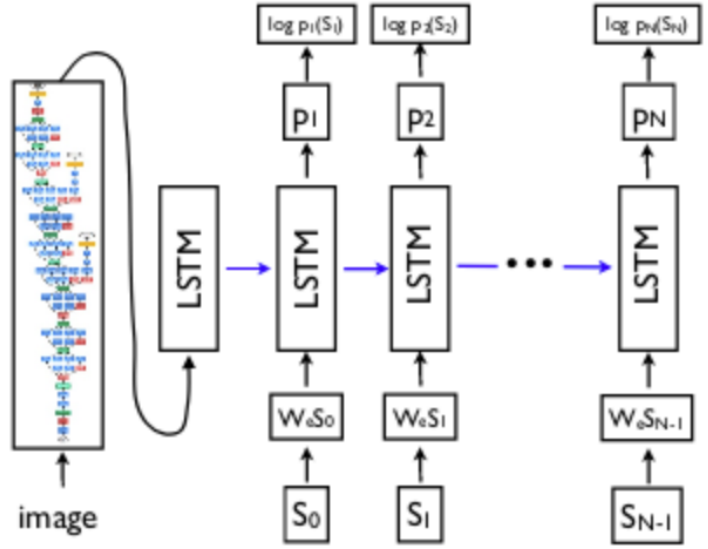


Bunk bed with a narrow shelf sitting underneath it.

Training

- Idea is to use a CNN as encoder and a RNN as decoder





- CNN encoder produces a representation of the input image by embedding it to a fixed-length vector
  - Inception network

- RNN decoder uses as input the last hidden layer of the CNN. It returns the predictions and the decoder hidden state.

Result

man in black shirt is playing guitar.

construction worker in orange safety vest is working on road.

two young girls are playing with lego toy.

boy is doing backflip on wakeboard.

https://www.youtube.com/watch?v=e-WB4lfg30M

Two-Minute Papers

Quiz

https://b.socrative.com/login/student/

Room : CONTI6128