Language Technologies Institute

Carnegie Mellon University

Tutorial on Multimodal Machine Learning

Louis-Philippe Morency
Tadas Baltrusaitis

Tutorial Objectives – Part 2

- Basic concepts – Part 2 [40 minutes]
  - Language models
    - Unigrams, bigrams
  - Unimodal sequence modeling
    - Recurrent Neural Networks (RNN)
    - Long short term memory (LSTM) models
  - Optimization
    - Back-propagation through time
  - RNN language models

Tutorial Objectives – Part 2

- Multimodal translation and mapping [20 minutes]
  - Encoder – decoder models
    - Machine translation
    - Image captioning
  - Generative and Retrieval models
    - Cross-modal retrieval
    - Speech synthesis
    - Visual puppetry
- Small break [5-10 minutes]

Tutorial Objectives – Part 2

- Multimodal Alignment [20 minutes]
  - Latent alignment
    - Machine translation
    - Captioning with alignment
  - Explicit alignment
    - Dynamic time warping

Tutorial Objectives – Part 2

- Multimodal Fusion and co-learning [20 minutes]
  - Motivation
    - Complementarity and robustness
  - Model Free approaches
    - Early, late, hybrid
  - Model based approaches
    - Neural networks
    - Multiple Kernel Learning
    - Graphical models
  - Co-learning

Basic concepts part 2

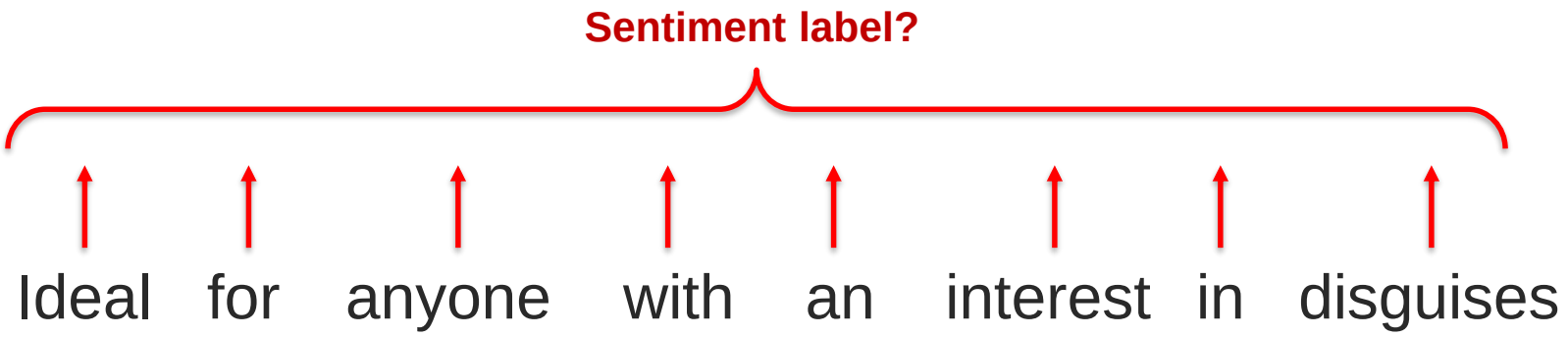
Language modeling – Sequence Modeling

- Need a way to computationally model sequences of words - language
- Need ways of representing language
  - So far mostly talked on how to represent individual words

Sequence Modeling

Prediction →

Sentiment ?
(positive or negative)



Sequence Modeling

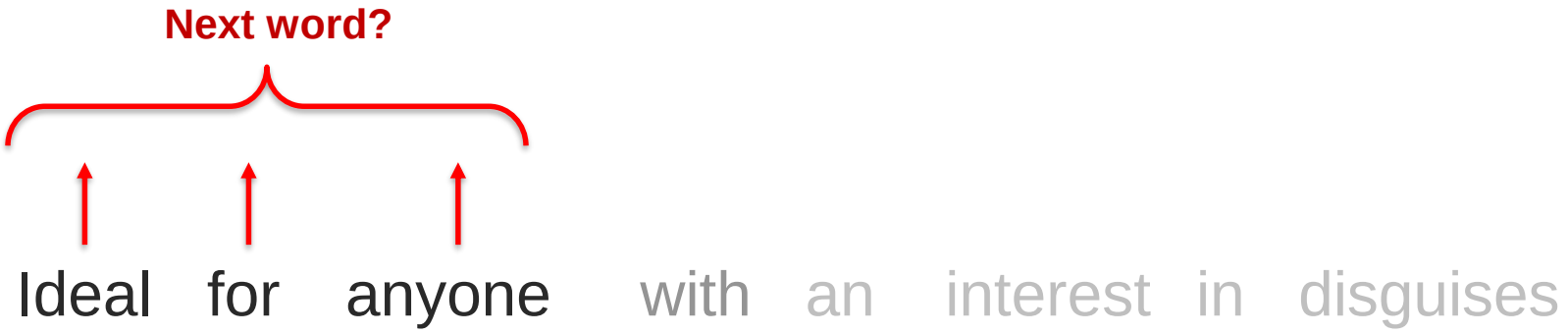
Prediction →

Part-of-speech ?
(noun, verb,…)

POS?	POS?	POS?	POS?	POS?	POS?	POS?	POS?
Ideal	for	anyone	with	an	interest	in	disguises

Sequence Modeling

Prediction → Language Model



Sequence Modeling

- One way to model a language is by predicting what word comes next
- This allows us to estimate the likelihood of a sentence
- If we can predict the next word reliably we have a good understanding of the language and that piece of text

N-Gram Model Formulas

- Word sequences

$$w_1^n = w_1 \ldots w_n$$

- Chain rule of probability

$$P(w_1^n) = P(w_1)P(w_2 \mid w_1)P(w_3 \mid w_1^2)\ldots P(w_n \mid w_1^{n-1}) = \prod_{k=1}^{n} P(w_k \mid w_1^{k-1})$$

- Bigram approximation

$$P(w_1^n) = \prod_{k=1}^{n} P(w_k \mid w_{k-1})$$

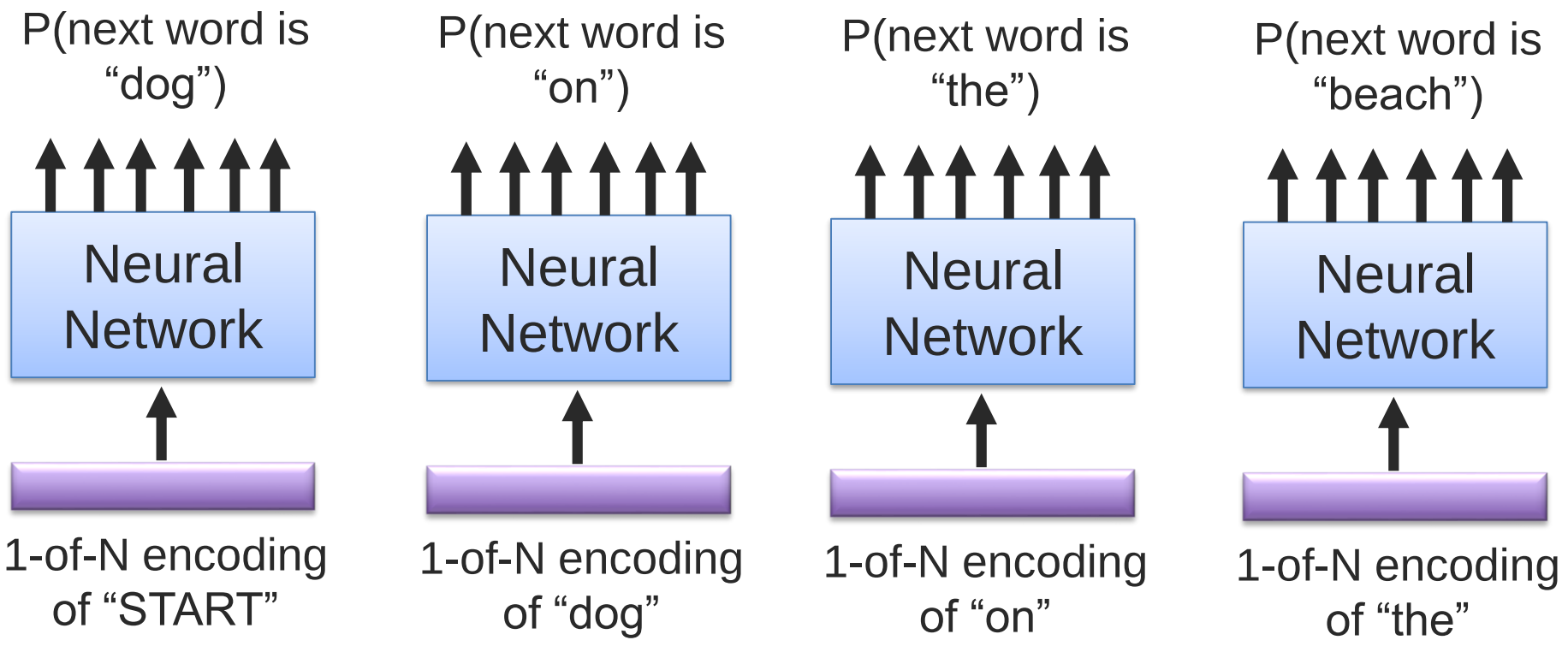
- N-gram approximation

$$P(w_1^n) = \prod_{k=1}^{n} P(w_k \mid w_{k-N+1}^{k-1})$$

Neural-network based Language Model (LM)

P(“dog on the beach”)
=P(dog|START)P(on|dog)P(the|on)P(beach|the)

P(b|a): not from count, but the NN that can predict the next word.



Sequence Modeling – vectorial representation

Learning →

Sequence representation

[0,1; 0,0004; 0;….; 0,01; 0.09; 0,05]

Ideal for anyone with an interest in disguises

Sequence Modeling – vectorial representation

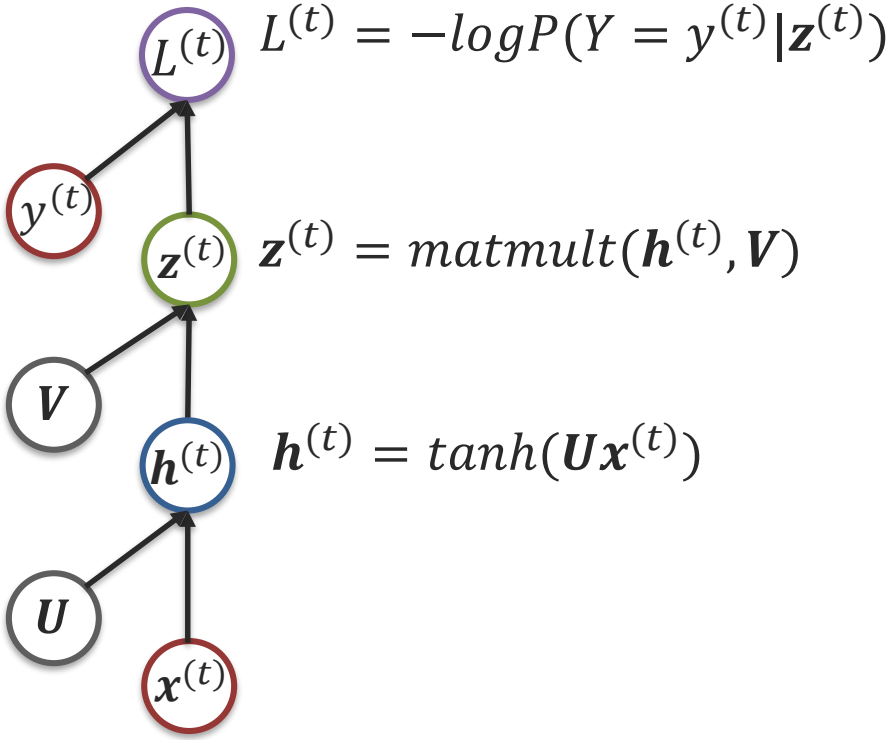
- Part-of-speech ?
  (noun, verb,…)
- Sentiment ?
  (positive or negative)
- Language Model
- Sequence representation

Main Challenges:

- Sequences of variable lengths (e.g., sentences)
- Keep the number of parameters at a minimum
- Take advantage of possible redundancy

Recurrent Neural Networks

Feedforward Neural Network



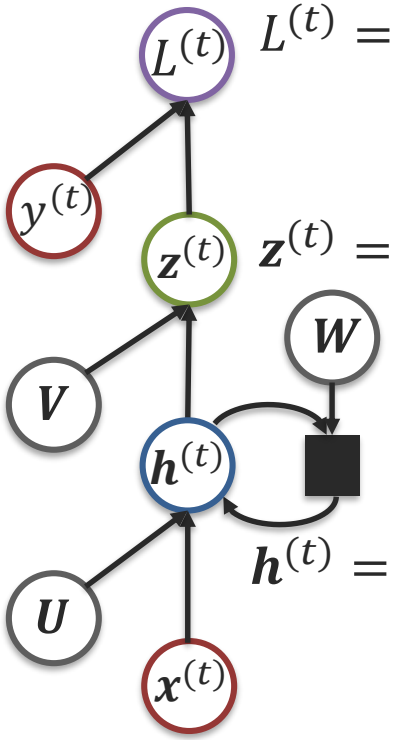
$$L^{(t)} = -logP(Y = y^{(t)}|\boldsymbol{z}^{(t)})$$

$$\boldsymbol{z}^{(t)} = matmult(\boldsymbol{h}^{(t)}, \boldsymbol{V})$$

$$\boldsymbol{h}^{(t)} = tanh(\boldsymbol{U}\boldsymbol{x}^{(t)})$$

Recurrent Neural Networks

$$L = \sum_t L^{(t)}$$



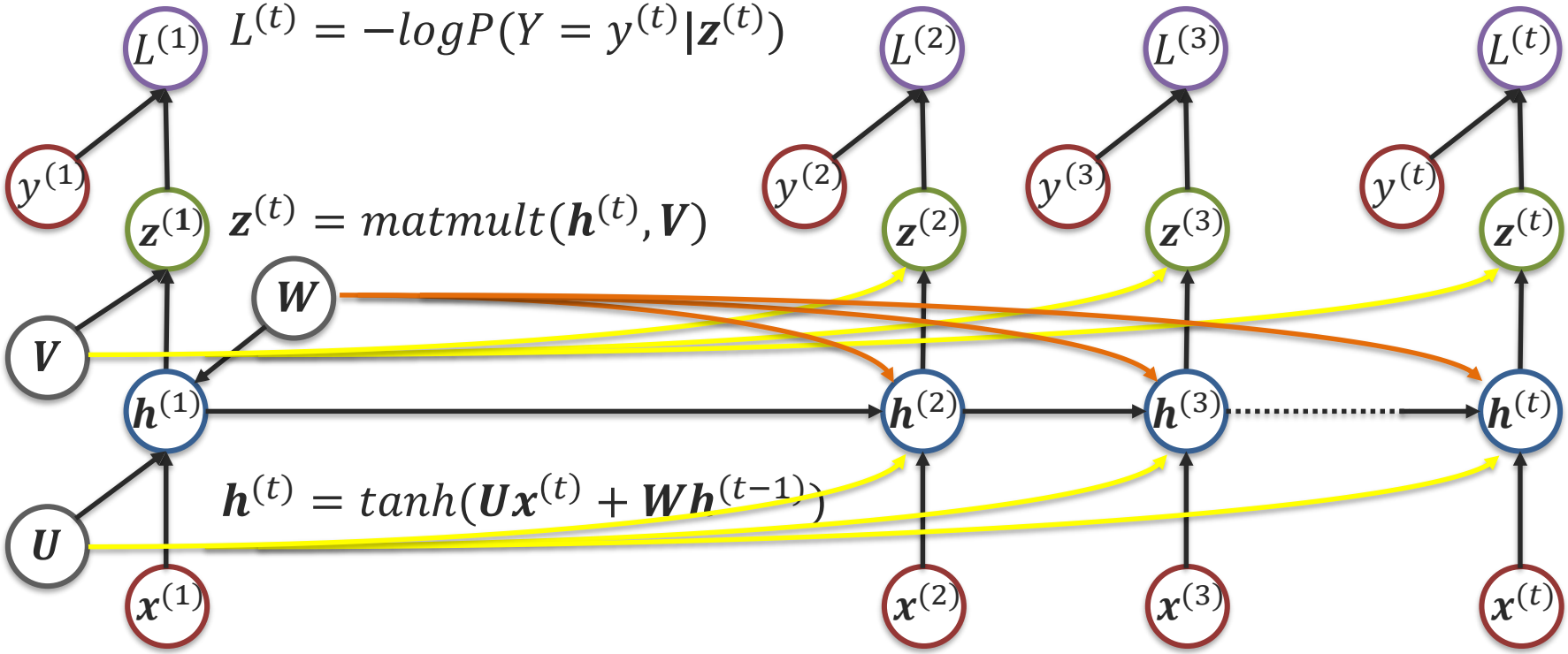
$$L^{(t)} = -logP(Y = y^{(t)}|\boldsymbol{z}^{(t)})$$

$$\boldsymbol{z}^{(t)} = matmult(\boldsymbol{h}^{(t)}, \boldsymbol{V})$$

$$\boldsymbol{h}^{(t)} = tanh(\boldsymbol{U}\boldsymbol{x}^{(t)} + \boldsymbol{W}\boldsymbol{h}^{(t-1)})$$

Recurrent Neural Networks - Unrolling

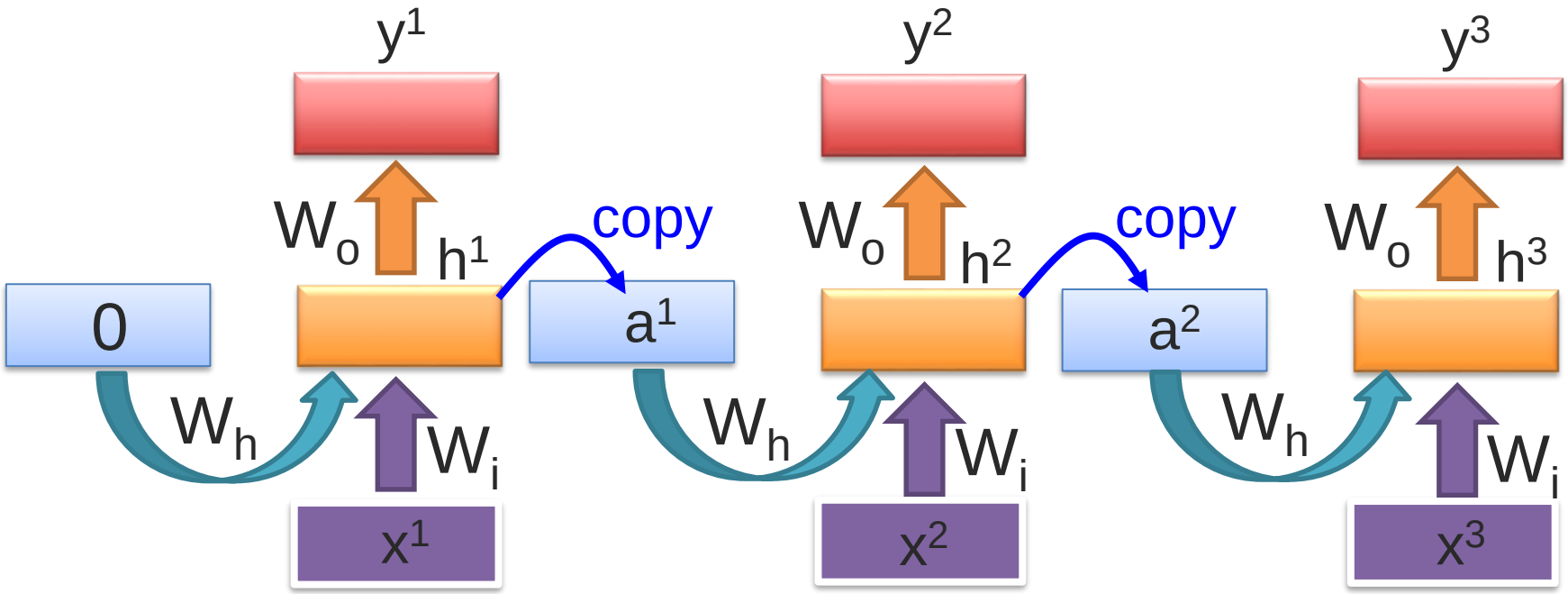
$$L = \sum_t L^{(t)}$$



Same model parameters are used for all time parts.

Recurrent Neural Networks - Unrolling

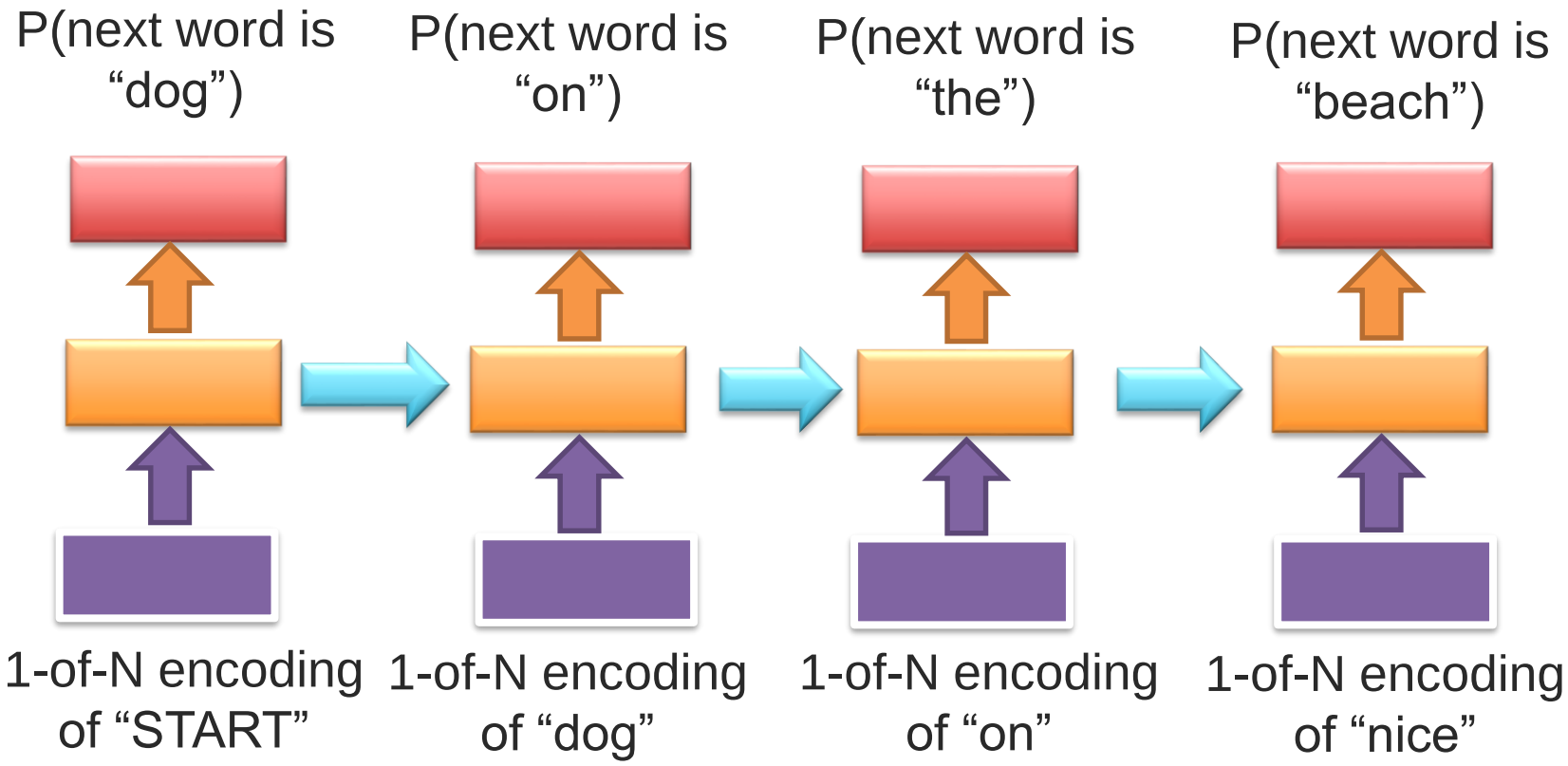
Input data: x^1 x^2 x^3 ……(x^i are vectors)



The same model parameters are used again and again.

Can be trained using backpropagation

Recurrent Neural Networks – Language models

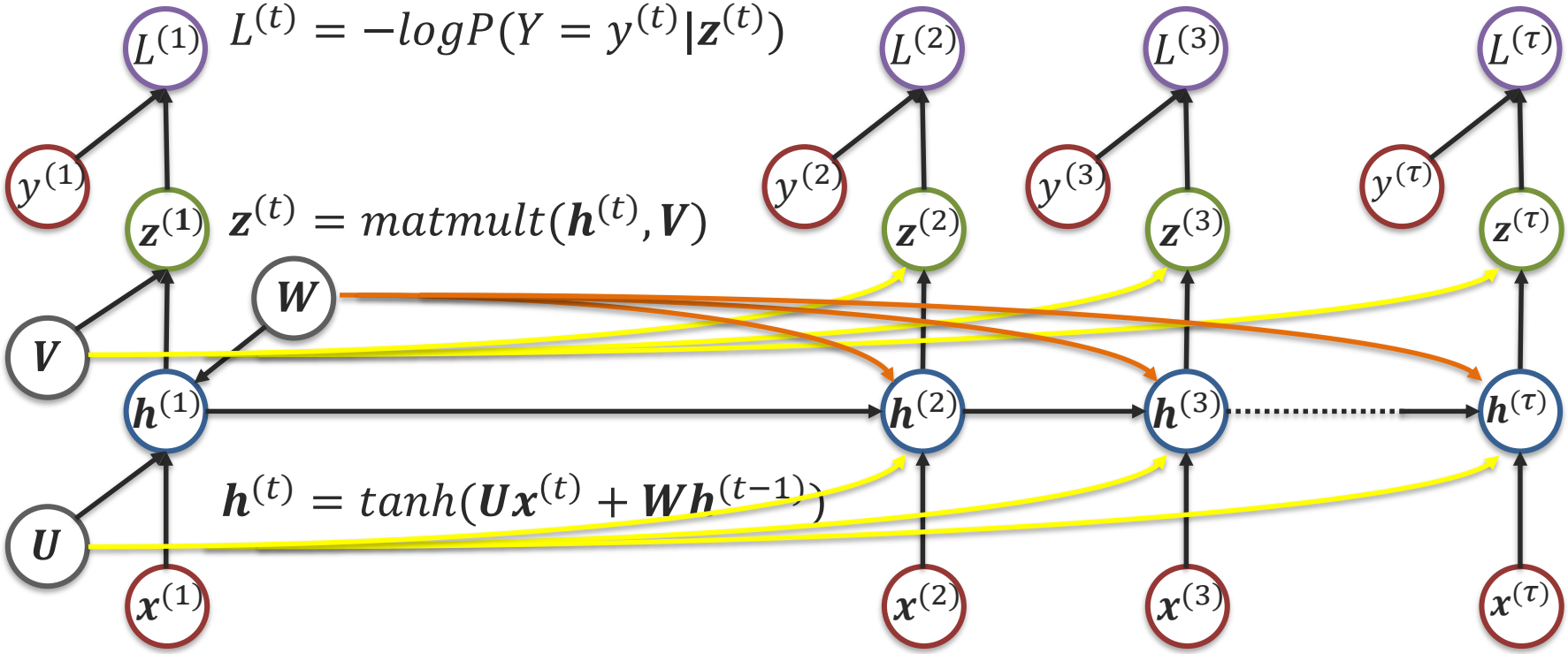


- Model long-term information

Backpropagation Through Time

Recurrent Neural Networks

$$L = \sum_t L^{(t)}$$



Backpropagation Through Time

$$L = \sum_t L^{(t)} = -\sum_t logP(Y = y^{(t)}|\boldsymbol{z}^{(t)})$$

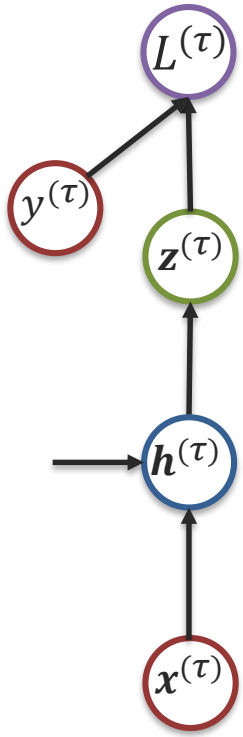
Gradient = "backprop" gradient x "local" Jacobian

$L^{(\tau)}$ or $L^{(t)}$ $\quad \frac{\partial L}{\partial L^{(t)}} = 1$

$\boldsymbol{z}^{(\tau)}$ or $\boldsymbol{z}^{(t)}$ $\quad (\nabla_{\boldsymbol{z}^{(t)}} L)_i = \frac{\partial L}{\partial z_i^{(t)}} = \frac{\partial L}{\partial L^{(t)}} \frac{\partial L^{(t)}}{\partial z_i^{(t)}} = sigmoid(z_i^t) - \mathbf{1}_{i,y^{(t)}}$

$\boldsymbol{h}^{(\tau)}$ $\quad \nabla_{\boldsymbol{h}^{(\tau)}} L = \nabla_{\boldsymbol{z}^{(\tau)}} L \frac{\partial z^{(\tau)}}{\partial \boldsymbol{h}^{(\tau)}} = \nabla_{\boldsymbol{z}^{(\tau)}} L \boldsymbol{V}$

$\boldsymbol{h}^{(t)} \rightarrow \boldsymbol{h}^{(t+1)}$ $\quad \nabla_{\boldsymbol{h}^{(t)}} L = \nabla_{\boldsymbol{z}^{(t)}} L \frac{\partial \boldsymbol{o}^{(t)}}{\partial \boldsymbol{h}^{(t)}} + \nabla_{\boldsymbol{z}^{(t+1)}} L \frac{\partial \boldsymbol{h}^{(t+1)}}{\partial \boldsymbol{h}^{(t)}}$



Backpropagation Through Time

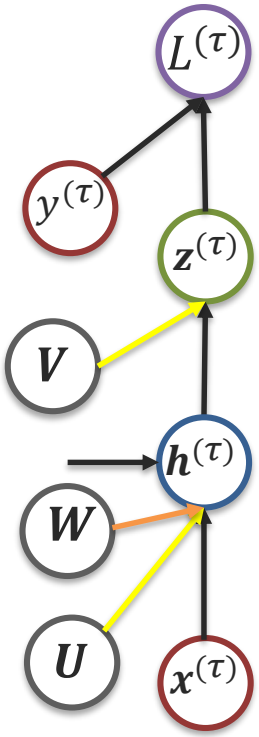
$$L = \sum_t L^{(t)} = -\sum_t logP(Y = y^{(t)}|\boldsymbol{z}^{(t)})$$

Gradient = "backprop" gradient x "local" Jacobian

$$\boldsymbol{V} \quad \nabla_{\boldsymbol{V}} L = \sum_t \left(\nabla_{\boldsymbol{z}^{(t)}} L\right) \frac{\partial \boldsymbol{z}^{(t)}}{\partial \boldsymbol{V}}$$

$$\boldsymbol{W} \quad \nabla_{\boldsymbol{W}} L = \sum_t \left(\nabla_{\boldsymbol{h}^{(t)}} L\right) \frac{\partial \boldsymbol{h}^{(t)}}{\partial \boldsymbol{W}}$$

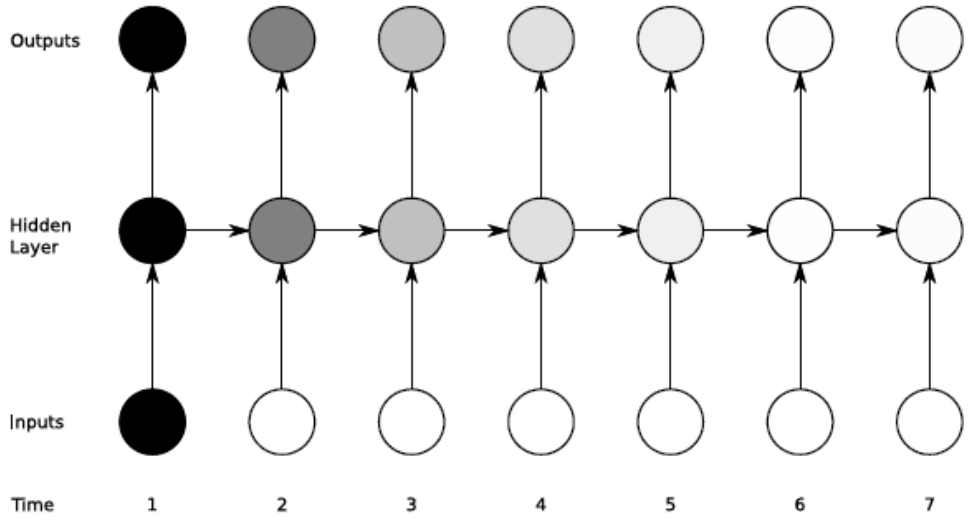
$$\boldsymbol{U} \quad \nabla_{\boldsymbol{U}} L = \sum_t \left(\nabla_{\boldsymbol{h}^{(t)}} L\right) \frac{\partial \boldsymbol{h}^{(t)}}{\partial \boldsymbol{U}}$$



Long-term Dependencies

Vanishing gradient problem for RNNs:

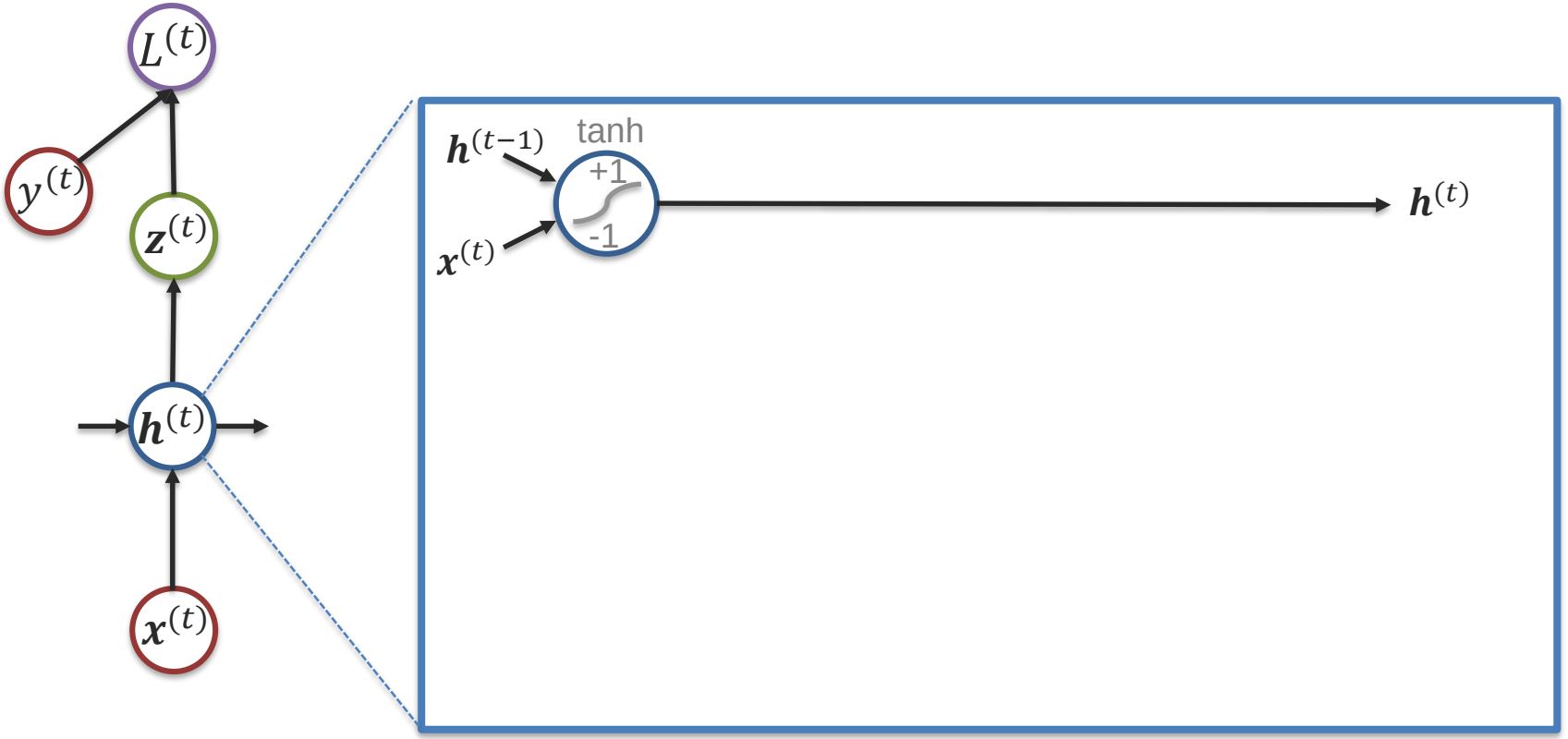
$$\boldsymbol{h}^{(t)} \sim tanh(\boldsymbol{W}\boldsymbol{h}^{(t-1)})$$



- The influence of a given input on the hidden layer, and therefore on the network output, either decays or blows up exponentially as it cycles around the network's recurrent connections.

Gated Recurrent Neural Networks

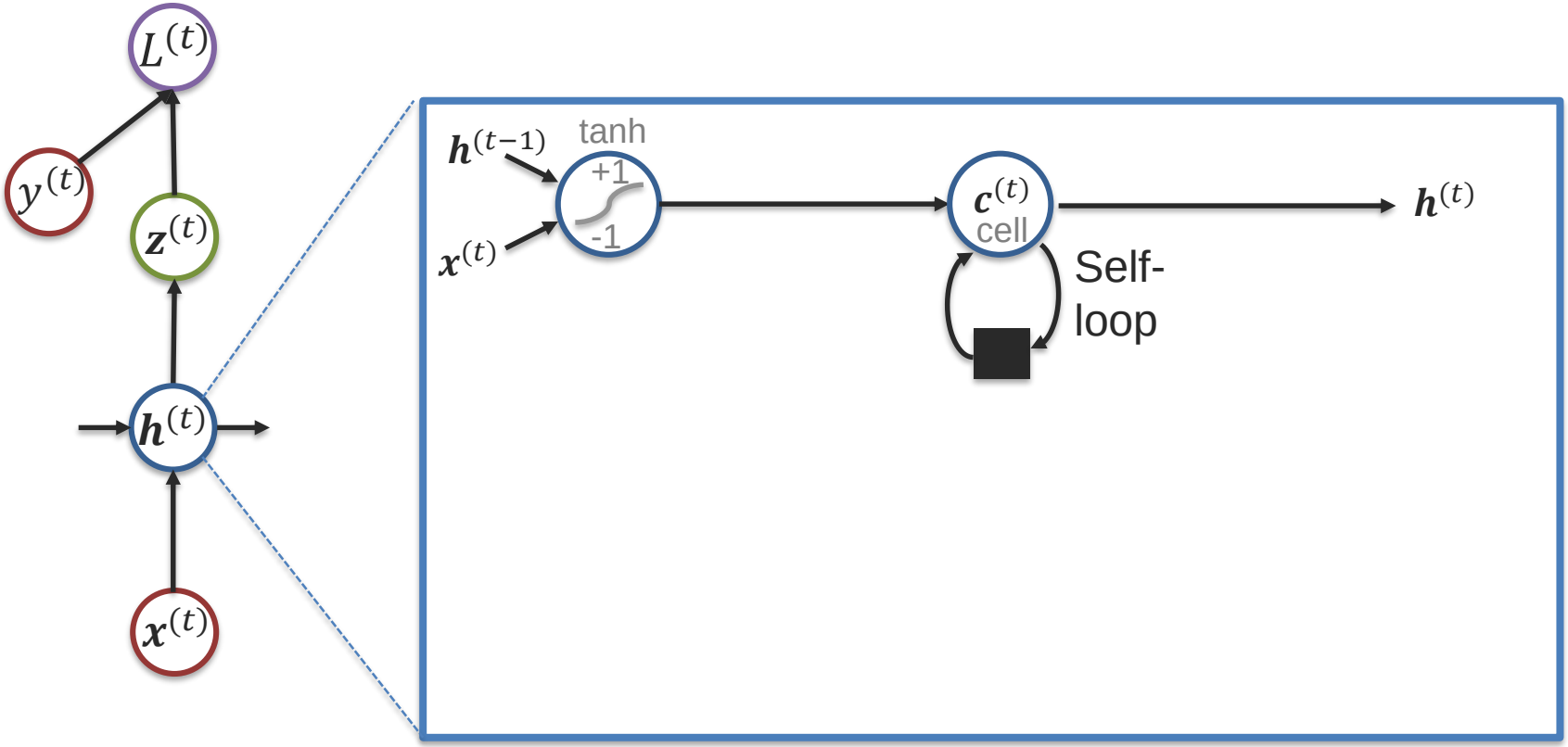
Recurrent Neural Networks



LSTM ideas: (1) "Memory" Cell and Self Loop

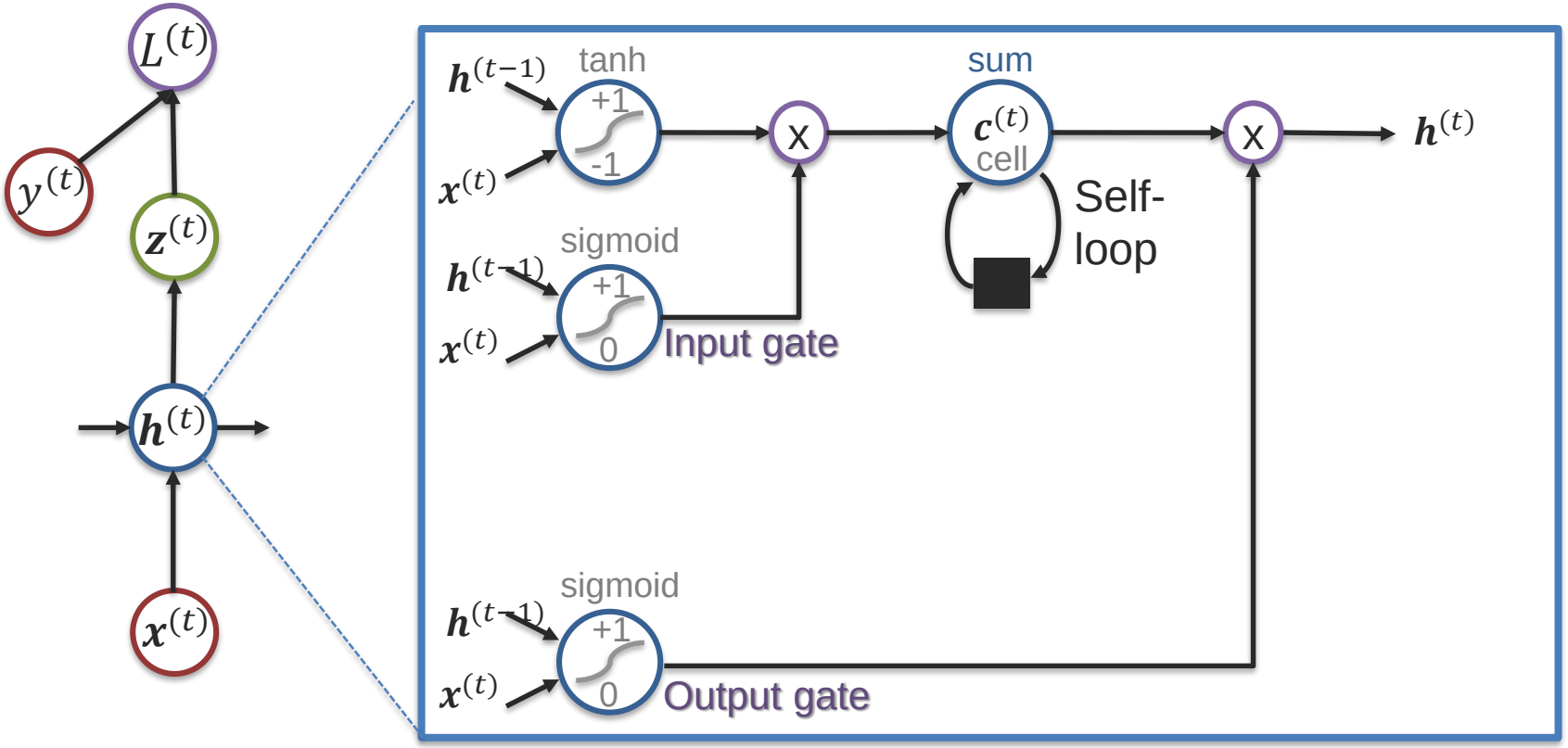
[Hochreiter and Schmidhuber, 1997]

Long Short-Term Memory (LSTM)



LSTM Ideas: (2) Input and Output Gates

[Hochreiter and Schmidhuber, 1997]

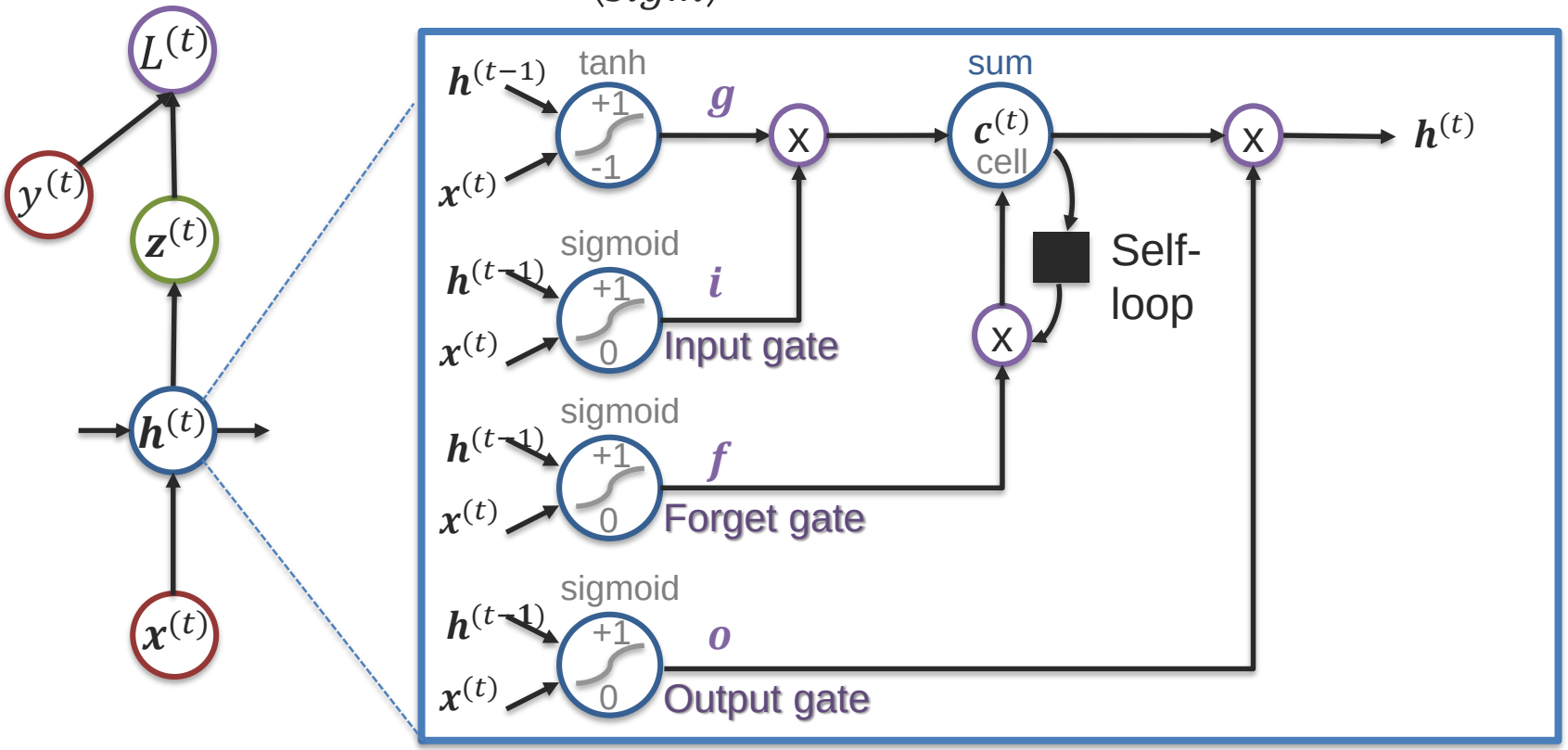


LSTM Ideas: (3) Forget Gate

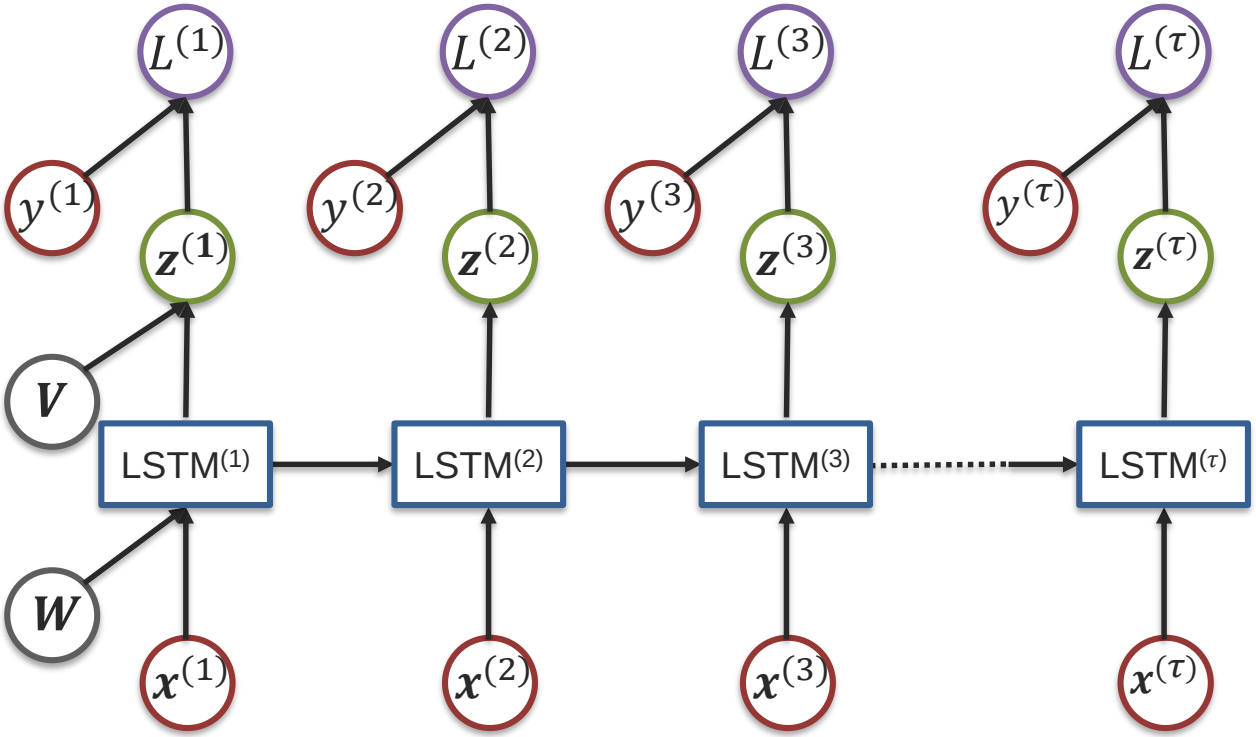
$$\begin{pmatrix} \boldsymbol{g} \\ \boldsymbol{i} \\ \boldsymbol{f} \\ \boldsymbol{o} \end{pmatrix} = \begin{pmatrix} tanh \\ sigm \\ sigm \\ sigm \end{pmatrix} \boldsymbol{W} \begin{pmatrix} \boldsymbol{h}^{(t-1)} \\ \boldsymbol{x}^{(t)} \end{pmatrix}$$

$$\boldsymbol{c}^{(t)} = \boldsymbol{f} \odot \boldsymbol{c}^{(t-1)} + \boldsymbol{i} \odot \boldsymbol{g}$$

$$\boldsymbol{h}^{(t)} = \boldsymbol{o} \odot \tanh(\boldsymbol{c}^{(t)})$$

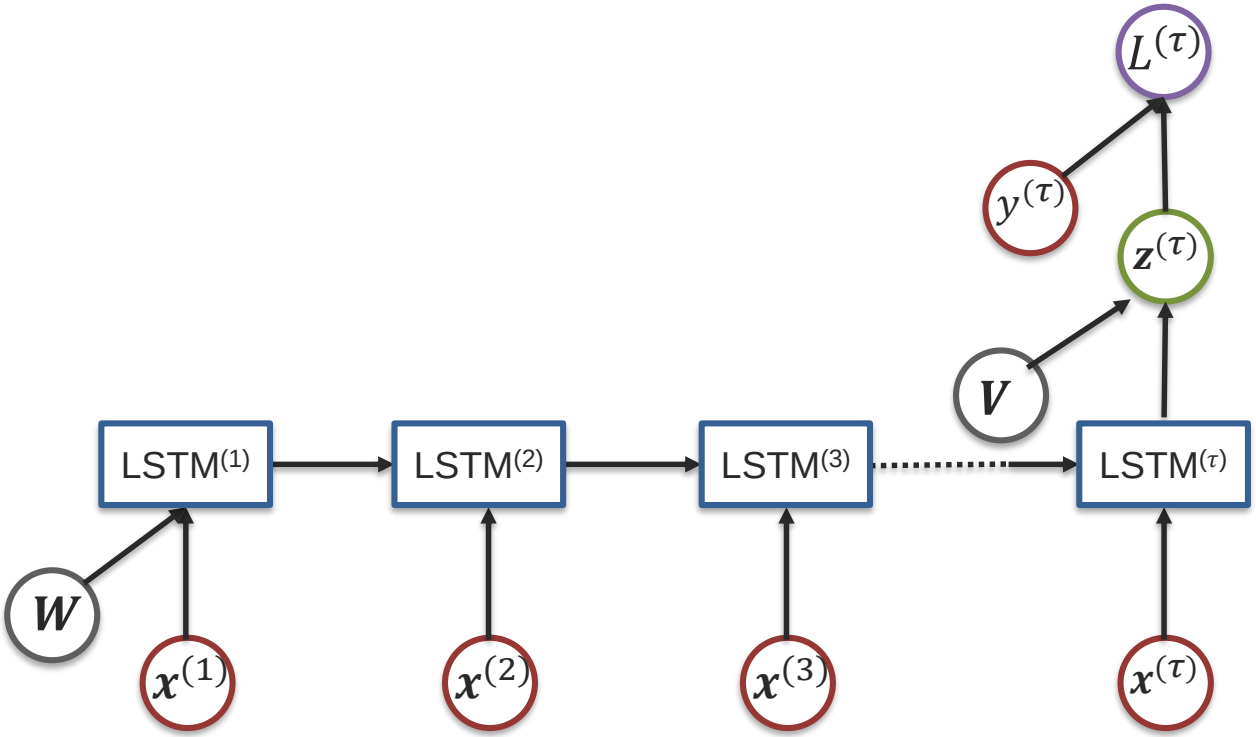


Recurrent Neural Network using LSTM Units



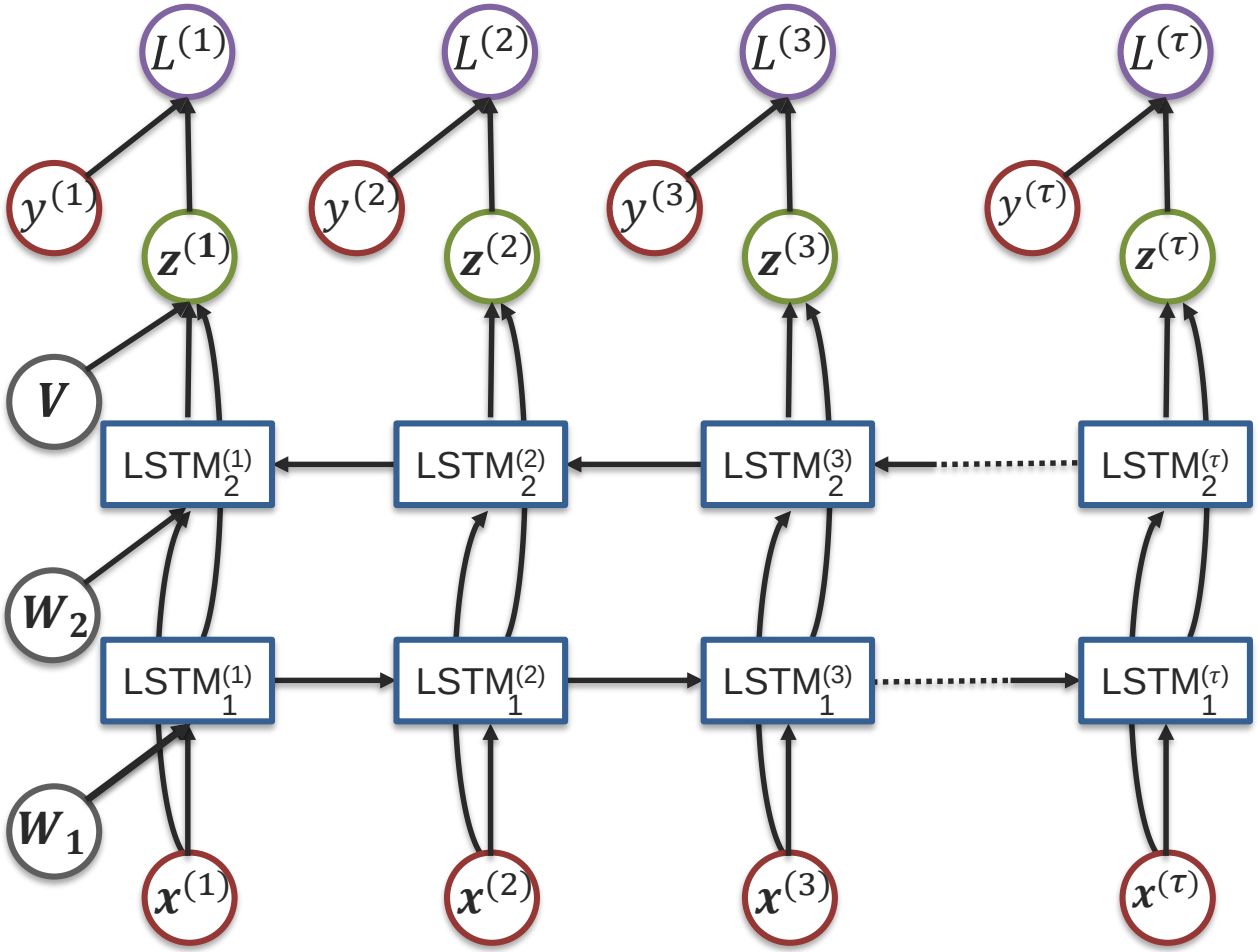
Gradient can still be computer using backpropagation!

Recurrent Neural Network using LSTM Units

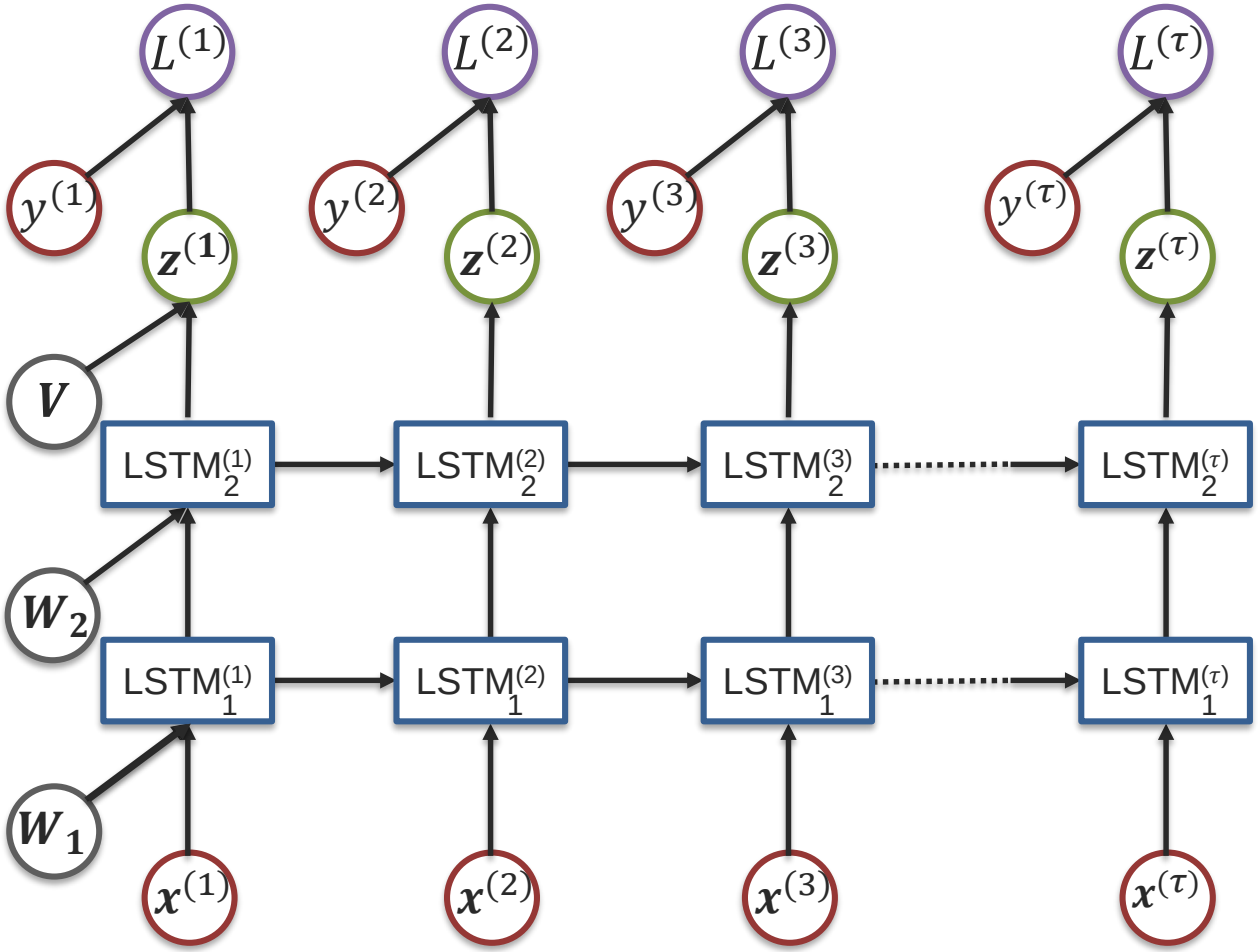


Gradient can still be computer using backpropagation!

Bi-directional LSTM Network



Deep LSTM Network

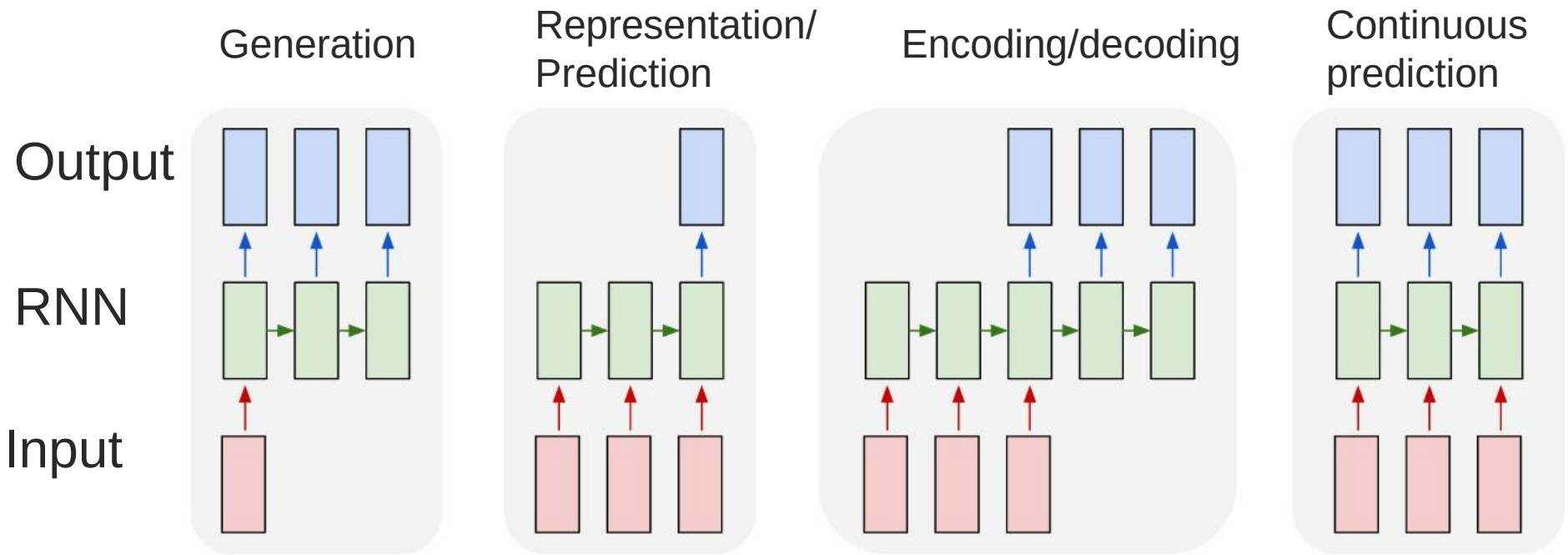


RNNs as language models

Recap

- Have a way to model sequences
- Using RNNs we can build a representation of a sequence by using the final hidden layer
- Also have representations at a time step so far

Use cases of RNNs



[http://karpathy.github.io/2015/05/21/rnn-effectiveness/]

Other uses in language modeling

- Allows to also assess the probability of a sentence given the model – perplexity
  - Can act as a scoring mechanism
  - Allows for reranking of sentences generated by an external system (early uses of RNNs in machine translation and speech recognition)
  - Can use to evaluate language models as well

RNN language model examples

- Trained on character level

```
PANDARUS: Alas, I think he shall be come approached and the day
When little srain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:
They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO:
Well, your wit is in the care of side and that.

Second Lord:
They would be ruled after this chamber, and
my fair nues begun out of the fact, to be conveyed,
Whose noble souls I'll have the heart of the wars.

Clown:
Come, sir, I will make did behold your worship.

VIOLA: I'll drink it.
```

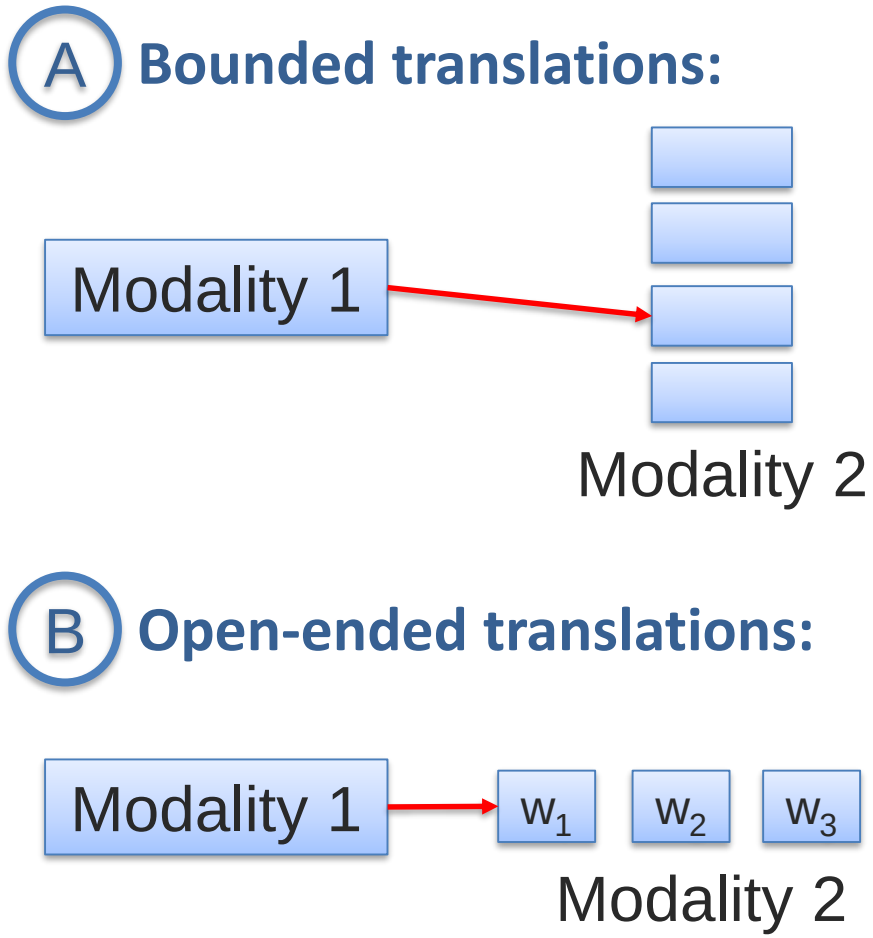
```
\begin{proof}
We may assume that $\mathcal{I}$ is an abelian sheaf on $\mathcal{C}$.
 \item Given a morphism $\Delta : \mathcal{F} \to \mathcal{I}$
is an injective and let $\mathfrak q$ be an abelian sheaf on $X$.
Let $\mathcal{F}$ be a fibered complex. Let $\mathcal{F}$ be a category.
\begin{enumerate}
\item \hyperref[setain-construction-phantom]{Lemma}
\label{lemma-characterize-quasi-finite}
Let $\mathcal{F}$ be an abelian quasi-coherent sheaf on $\mathcal{C}$.
Let $\mathcal{F}$ be a coherent $\mathcal{O}_X$-module.
Then $\mathcal{F}$ is an abelian catenary over $\mathcal{C}$.
\item The following are equivalent
\begin{enumerate}
\item $\mathcal{F}$ is an $\mathcal{O}_X$-module.
\end{lemma}
```

[http://karpathy.github.io/2015/05/21/rnn-effectiveness/]

Multimodal translation

What is translation

- Translating/Mapping from a source instance in one modality to a target instance in another
- For example:
  - Image ⟷ Description
  - Text ⟷ Speech
  - Image ⟷ Sound
  - Speech ⟷ Animation



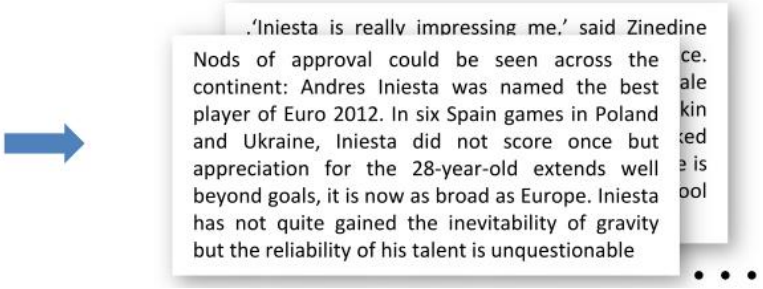
Translation challenges

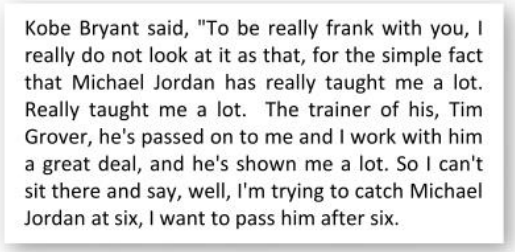
- Why is it difficult?
- Representation and generation
  - Involves being able to both represent and generate various modalities
  - May require a joint or comparable representation of modalities (in case of retrieval and ranking)
- Evaluation
  - Often very difficult to evaluate – the mapping is often open ended and not unique
  - Often the quality of translation is highly subjective

Bounded translation

Retrieval based translation

- Cross-media retrieval – bounded task
- Multimodal representation plays a key role here

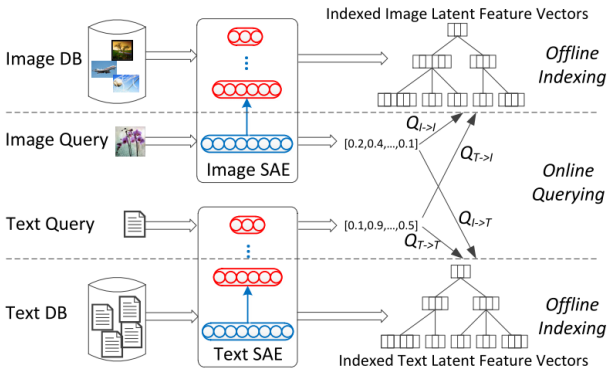




[Wei et al. 2015]

Retrieval based translation

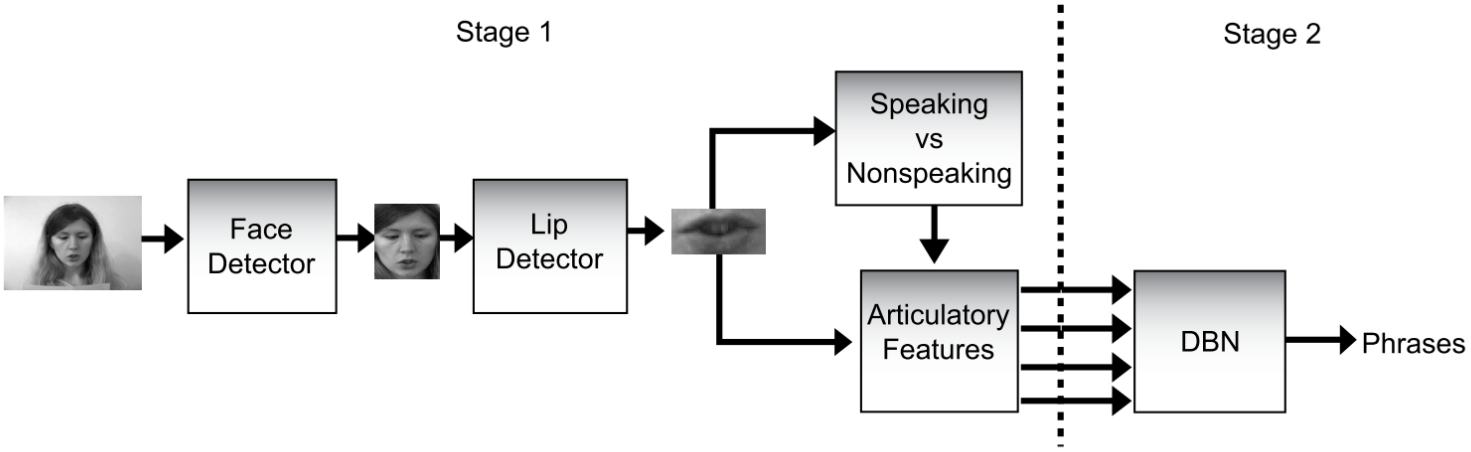
- Need a way to measure similarity between the modalities
- Remember multimodal representations
  - CCA
  - Coordinated
  - Joint
  - Hashing
- Can use pairs of instances to train them and retrieve closest ones during retrieval stage
- Objective and bounded task



[Wang et al. 2014]

Visual Speech Recognition - Lipreading

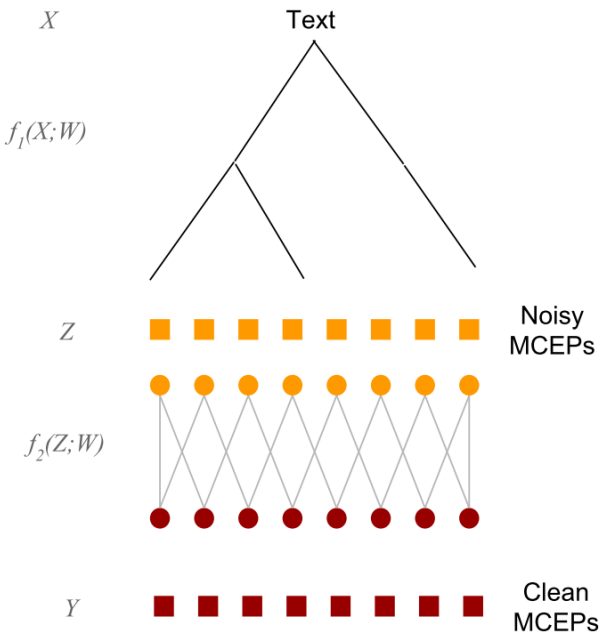
- Vision → Language
- Bounded and objective problem that is easier to evaluate
- Difficult problem as the mapping from a viseme and a phoneme is ambiguous (many sounds look the same on the lips)



[Saenko et al., 2005; Bear and Harvey 2016]

Speech synthesis

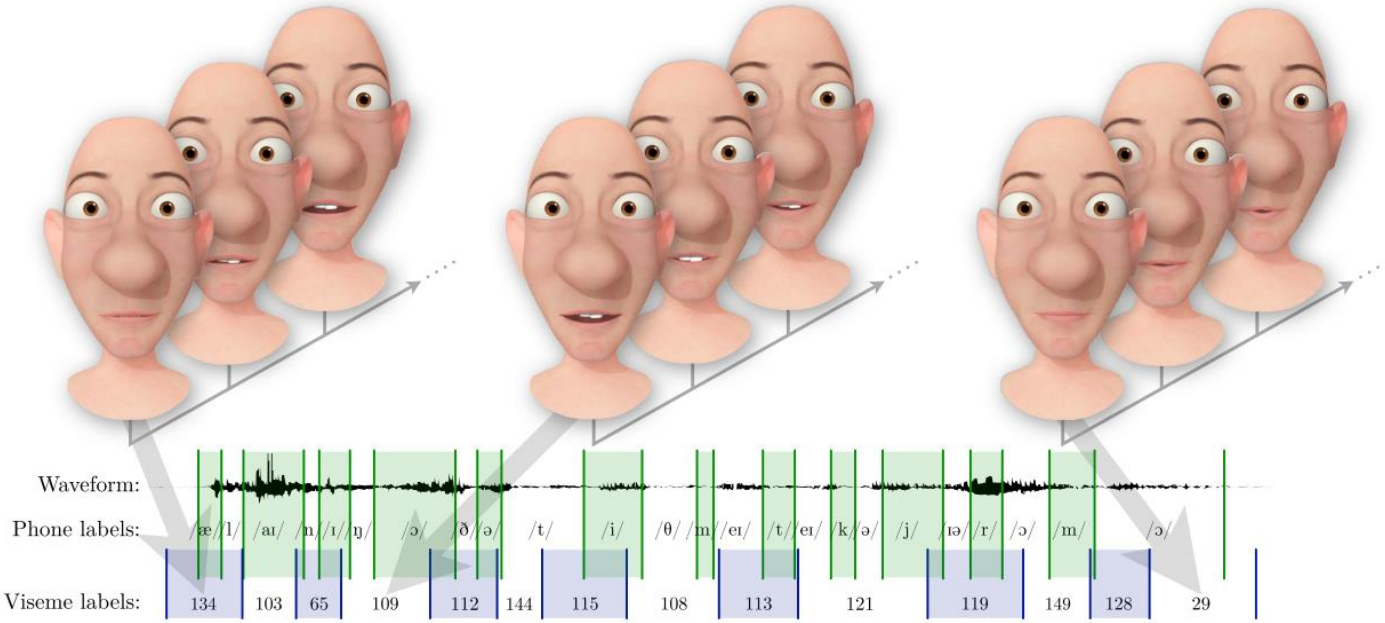
- Text → Sound
- Many intermediate building blocks
- End to end training approaches are becoming popular
- Works best for synthesizing a particular person's speech (with lots of training data for that individual)
- Very difficult to evaluate
- Parametrising for different people?



[Muthukumar and Black, 2016]

Visual speech generation

- Waveform → Phone → Viseme
- Text → Waveform → Phone → Viseme
- Difficult to evaluate – a subjective task



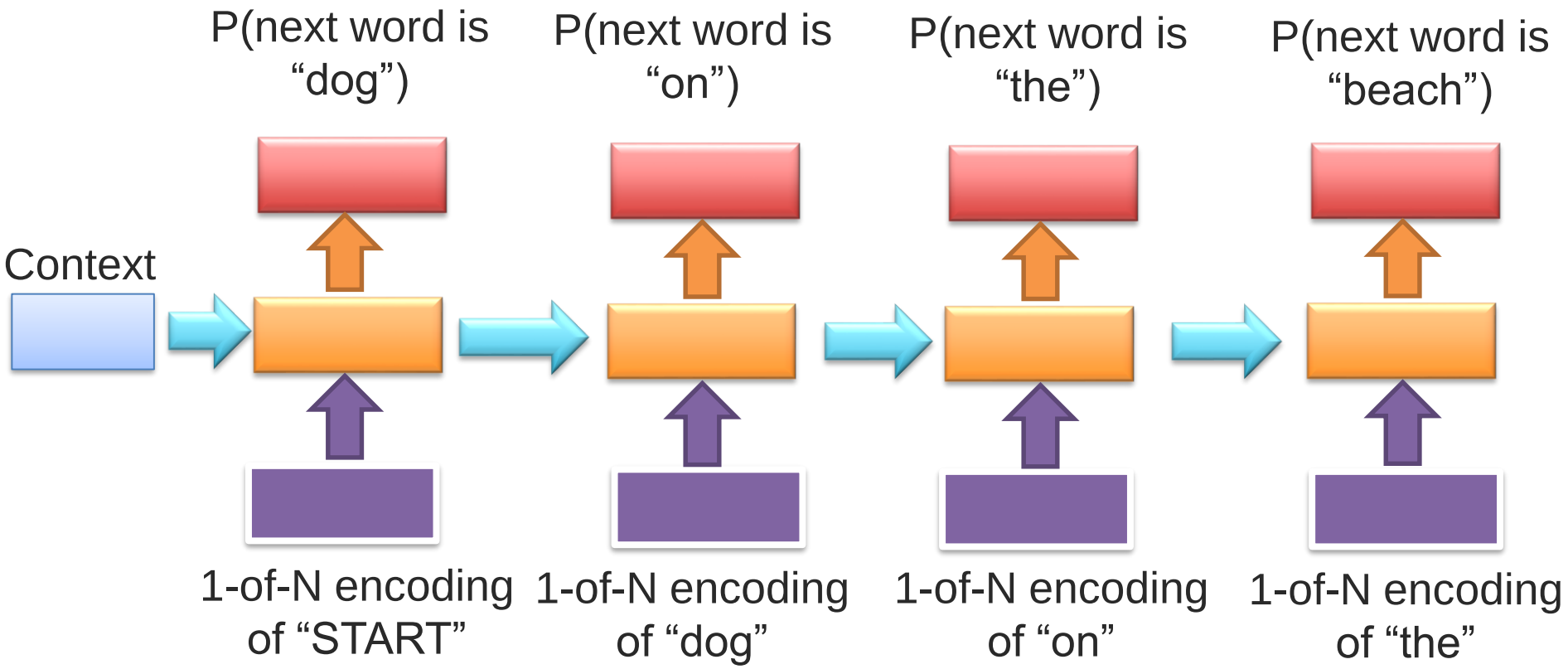
[Taylor et al., “Dynamic Units of Visual Speech”, SIGGRAPH 2012]

Unbounded translation

Machine translation

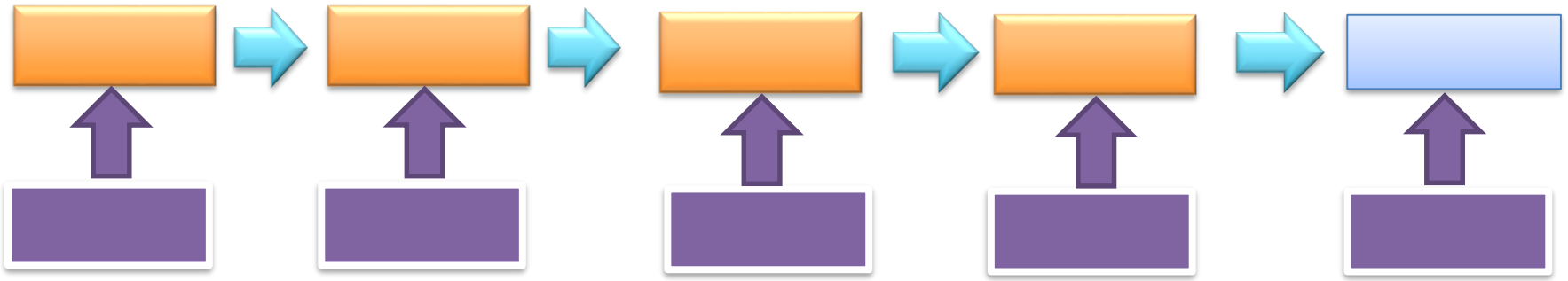
- Not exactly multimodal but a multi-view problem
- Good start for our discussion
- Given a sentence in one language translate it to another
- Dog on the beach → le chien sur la plage

RNN-based Sentence Generation (Decoder)



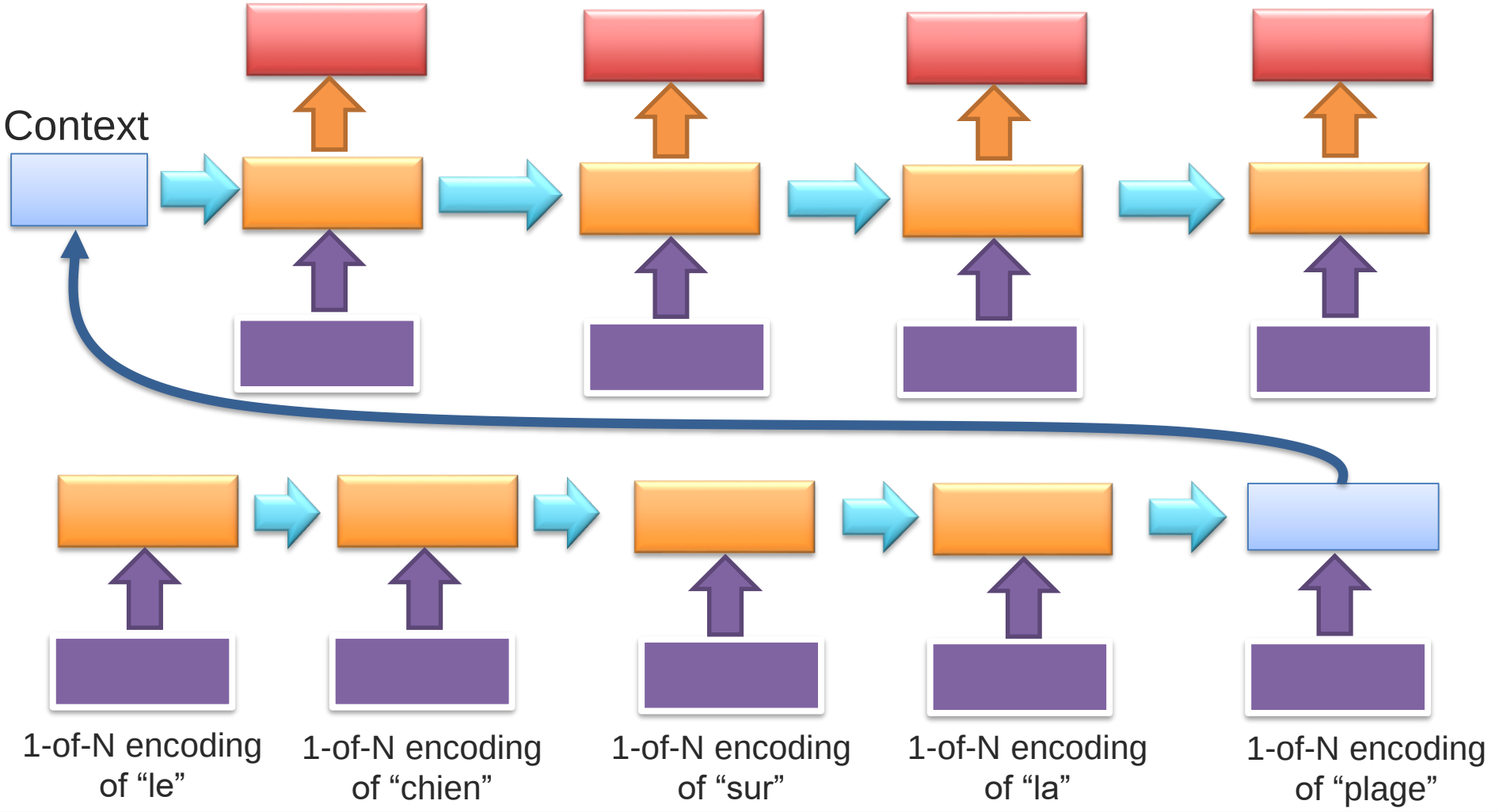
- Model long-term information

RNN-based Sentence Representation (Encoder)



Encoder-Decoder Architecture

[Cho et al., “Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation”, EMNLP 2014]



Training

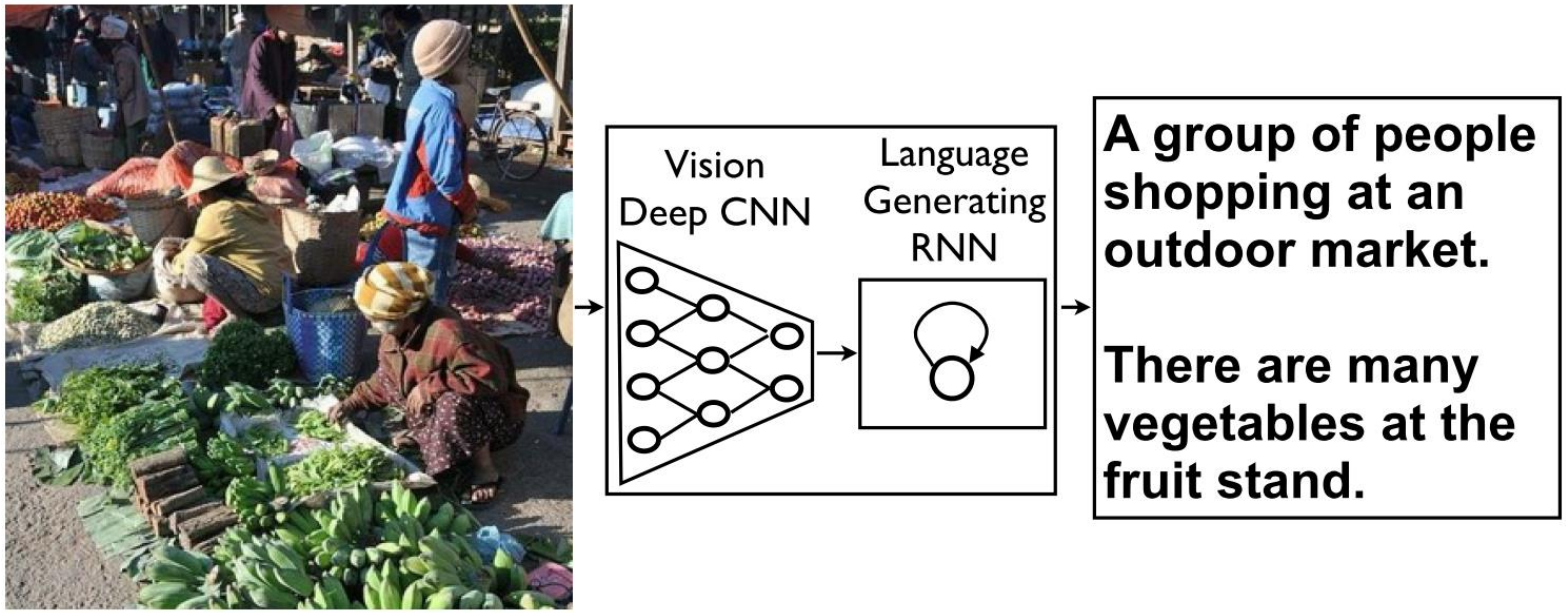
- Have pairs of correct translations
- Cross-entropy loss for predicting the right sentence
- The whole system can work end-to-end

- Optional extra for better accuracy:
  - Pre-training both the encoder and decoder on language models
  - Bidirectional LSTM

Why is this exciting?

- End to end translation between modalities!
- Overtaking handcrafted systems with lots of moving parts in accuracy
- We do not need to limit ourselves to same modality for translation!

Image captioning with RNNs



[Vinyals et al., “Show and Tell: A Neural Image Caption Generator”, CVPR 2015]

Image captioning with RNNs

- Same training as before, but now the encoder is a CNN
- We can train a system end to end using cross entropy loss
- Often use already pre-trained CNN and RNN models
  - CNN on visual object classification
  - RNN on language modeling
  - Can also train a coordinated multimodal representation space as well
- Training is done on pairs of images and captions

- Datasets
  - MS COCO
  - Flickr8k
  - Flickr30k

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.

Evaluation

- Tricky to do automatically!
- Ideally want humans to evaluate
  - What do you ask?
  - Can't use human evaluation for validating models – too slow and expensive
- Using standard machine translation metrics instead
  - BLEU, ROUGE CIDER, Meteor

State-of-the-art on MS COCO

	CIDEr-D	Meteor	ROUGE-L	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Google[4]	0.943	0.254	0.53	0.713	0.542	0.407	0.309
MSR Captivator[9]	0.931	0.248	0.526	0.715	0.543	0.407	0.308
m-RNN[15]	0.917	0.242	0.521	0.716	0.545	0.404	0.299
MSR[8]	0.912	0.247	0.519	0.695	0.526	0.391	0.291
Nearest Neighbor[11]	0.886	0.237	0.507	0.697	0.521	0.382	0.28
m-RNN (Baidu/ UCLA)[16]	0.886	0.238	0.524	0.72	0.553	0.41	0.302
Berkeley LRCN[2]	0.869	0.242	0.517	0.702	0.528	0.384	0.277
Human[5]	0.854	0.252	0.484	0.663	0.469	0.321	0.217

State-of-the-art on MS COCO

- A challenge was done with actual human evaluations of the captions (CVPR 2015)

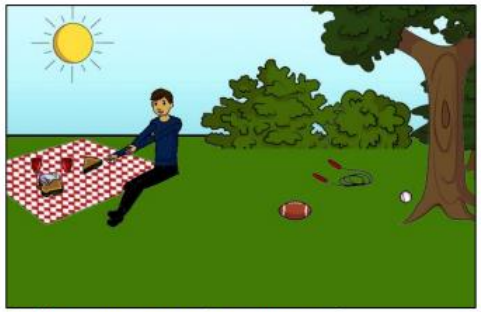
	M1	M2	M3	M4	M5
Human[5]	0.638	0.675	4.836	3.428	0.352
Google[4]	0.273	0.317	4.107	2.742	0.233
MSR[8]	0.268	0.322	4.137	2.662	0.234
Montreal/Toronto[10]	0.262	0.272	3.932	2.832	0.197
MSR Captivator[9]	0.250	0.301	4.149	2.565	0.233
Berkeley LRCN[2]	0.246	0.268	3.924	2.786	0.204
m-RNN[15]	0.223	0.252	3.897	2.595	0.202
Nearest Neighbor[11]	0.216	0.255	3.801	2.716	0.196

Visual Question Answering

- A very new and exciting task created in part to address evaluation problems with the above task
- Task - Given an image and a question answer the question (http://www.visualqa.org/)

What color are her eyes?
What is the mustache made of?

How many slices of pizza are there?
Is this a vegetarian pizza?

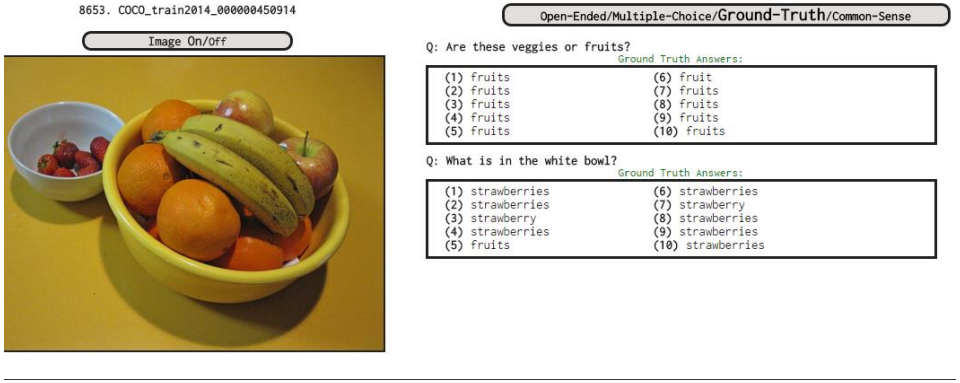


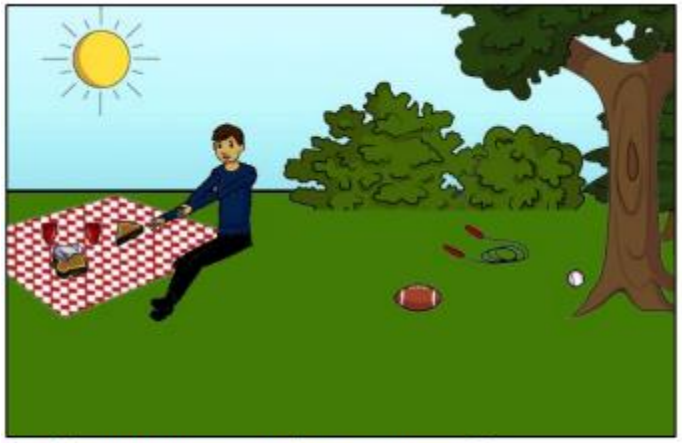
Is this person expecting company?
What is just under the tree?

Does it appear to be rainy?
Does this person have 20/20 vision?

Visual Question Answering

- Real images
  - 200k MS COCO images
  - 600k questions
  - 6M answers
  - 1.8M plausible answers
- Abstract images
  - 50k scenes
  - 150k questions
  - 1.5M answers
  - 450k plausible answers





Is this person expecting company?
What is just under the tree?

Visual Question Answering

- A workshop and a challenge next door in this CVPR
- Dominated by LSTM + CNN
- An easier and more objective task to evaluate
- Currently good at yes/no question, not so much free form and counting

Multi-modal alignment

Multimodal alignment

- Identifying relations between the elements from two or more different modalities
  - Time/Space/Instance

- Latent
  - The alignment of modalities happens as a side effect of another task – representation/translation/fusion
  - Modeling alignment allows us to solve the above problems better
- Explicit
  - The task itself is to align two or more modalities

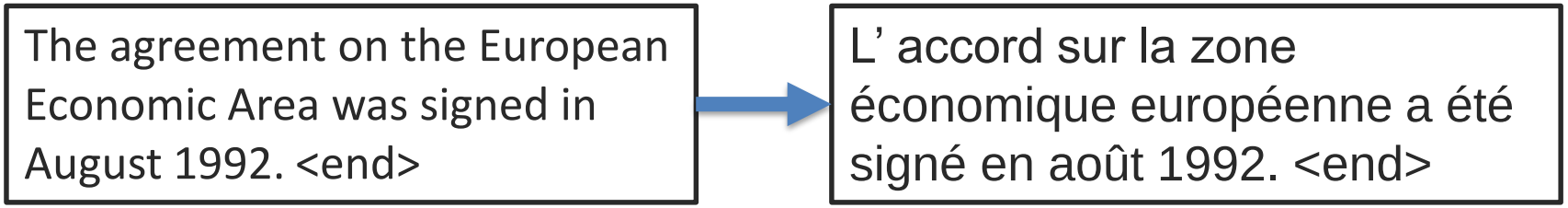
Multimodal alignment - challenges

- Challenges
  - Labeled data is rare - labeling data is really difficult and expensive
  - Temporal alignment is difficult – what are the exact event boundaries?
  - Requires joint representation or translation between modalities

Latent alignment – attention models

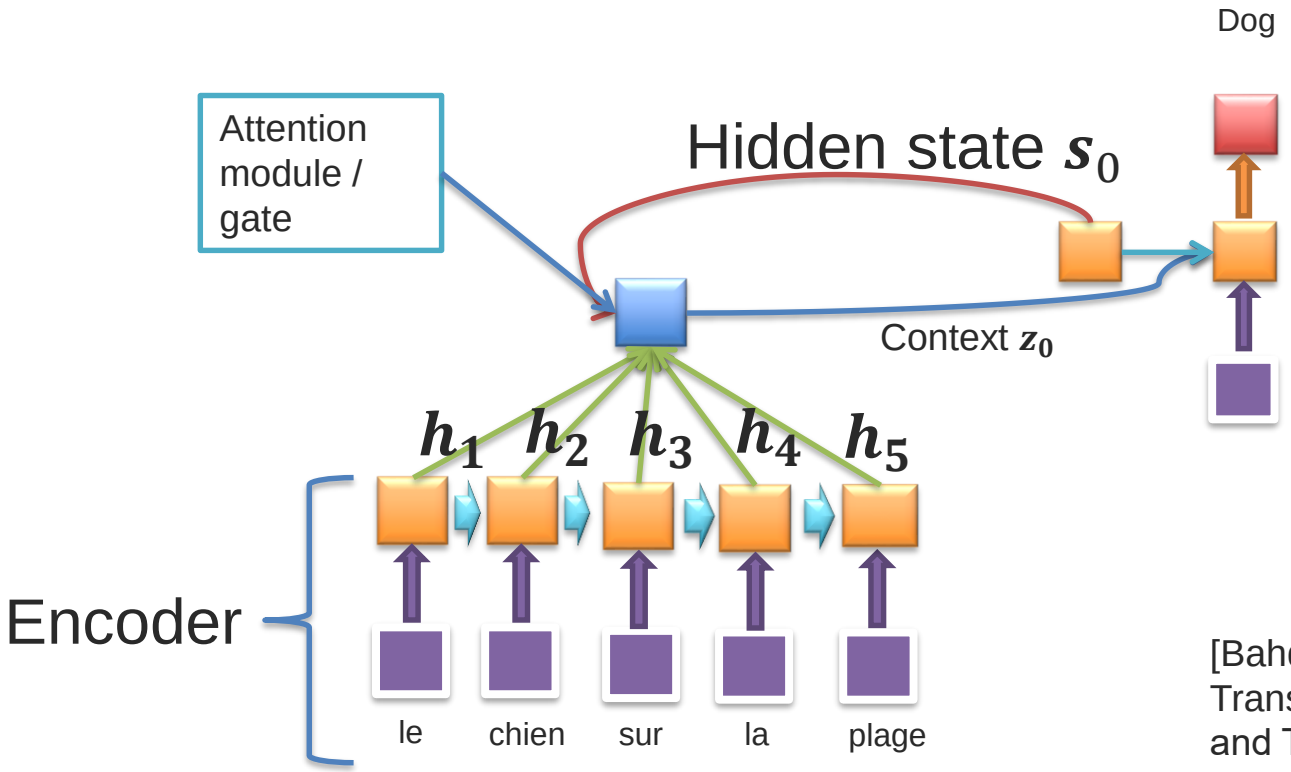
Machine Translation and image captioning with RNNs

- What is the problem with this?
- What happens when the sentences are very long or images have many objects?
- We expect the encoders hidden state to capture everything in a sentence, a very complex state in a single vector, such as



Decoder – attention model

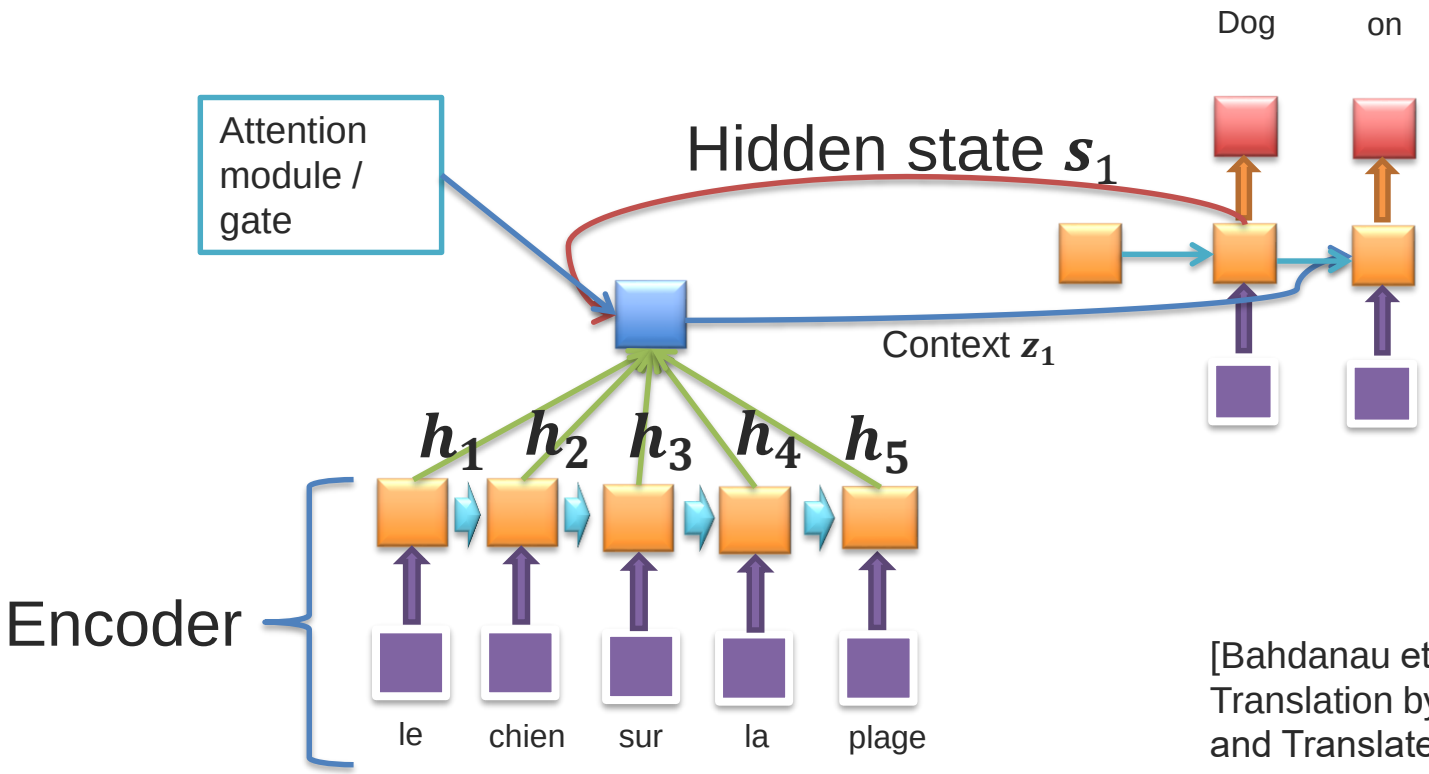
- Before encoder would just take the final hidden state, now we actually care about the intermediate hidden states



[Bahdanau et al., "Neural Machine Translation by Jointly Learning to Align and Translate", ICLR 2015]

Decoder – attention model

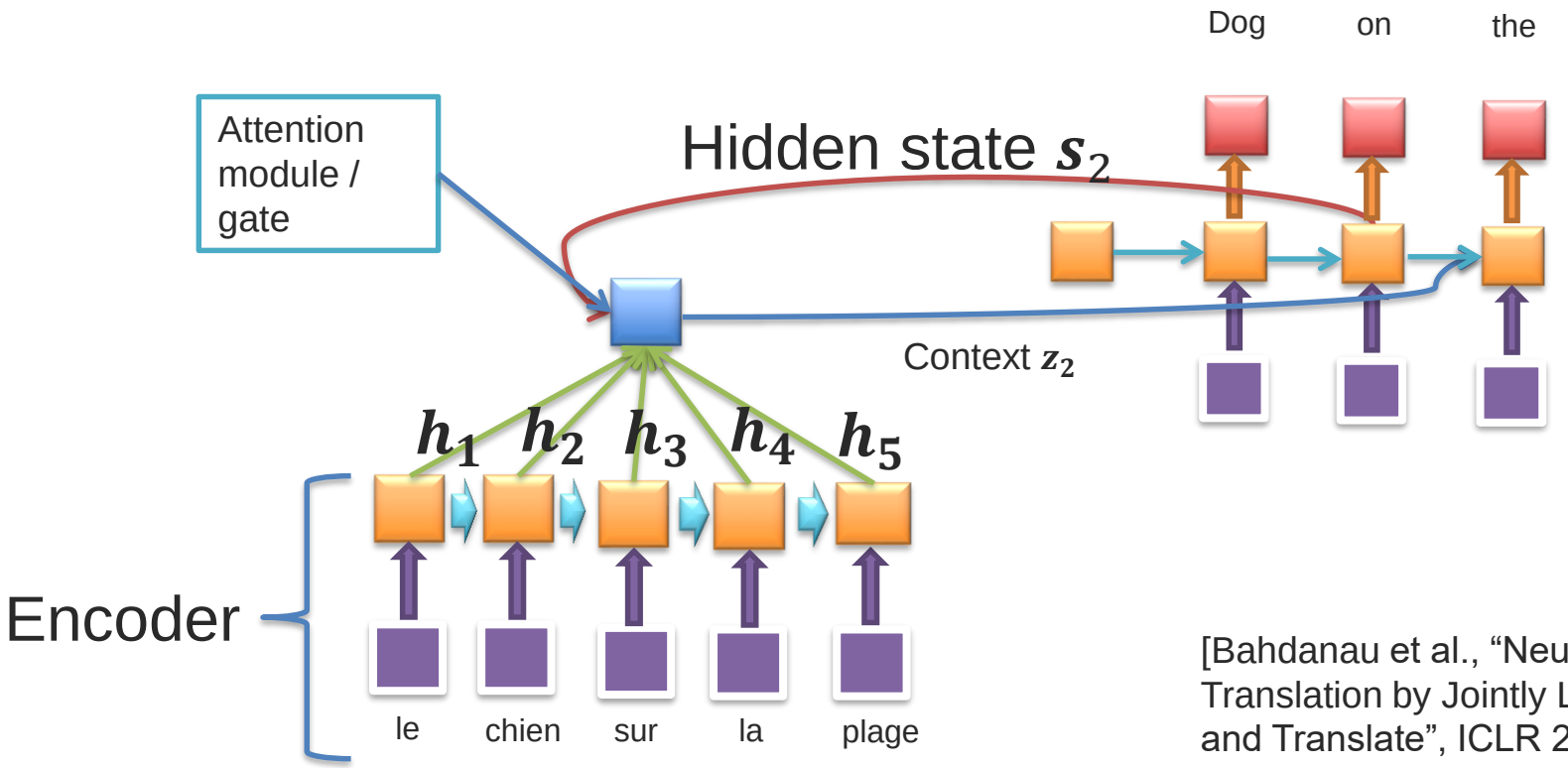
- Before encoder would just take the final hidden state, now we actually care about the intermediate hidden states



[Bahdanau et al., "Neural Machine Translation by Jointly Learning to Align and Translate", ICLR 2015]

Decoder – attention model

- Before encoder would just take the final hidden state, now we actually care about the intermediate hidden states



[Bahdanau et al., "Neural Machine Translation by Jointly Learning to Align and Translate", ICLR 2015]

How do we encode attention

- Now the generation doesn't depend on a local context but on a global one, before:
  - $p(y_i|y_1, \dots, y_{i-1}, \boldsymbol{x}) = g(y_{i-1}, s_i, z)$, where $z = h_T$, and s_i - the current state of the decoder
- Now:
  - $p(y_i|y_1, \dots, y_{i-1}, \boldsymbol{x}) = g(y_{i-1}, s_i, z_i)$
- Have an attention "gate"
  - A different context z_i used at each time step!
  - $z_i = \sum_{j=i}^{T_x} \alpha_{ij} h_j$

α_{ij} - the attention for word j at generation step i

MT with attention

- So how do we determine α_{ij},
  - $\alpha_{i,j} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T_x} \exp(e_{ik})}$ - softmax, making sure they sum to 1
- Where:
  - $e_{ij} = a(s_{i-1}, h_j) = \boldsymbol{v}^T \sigma(W s_{i-1} + U h_j)$
  - a feedforward network that can tell us given the current state of decoder how important is the current encoding is now
  - $\boldsymbol{v}, W, U$– learnable weights,
- $z_i = \sum_{j=i}^{T_x} \alpha_{ij} h_j$

expectation of the context (a fancy way to say it's a weighted average)

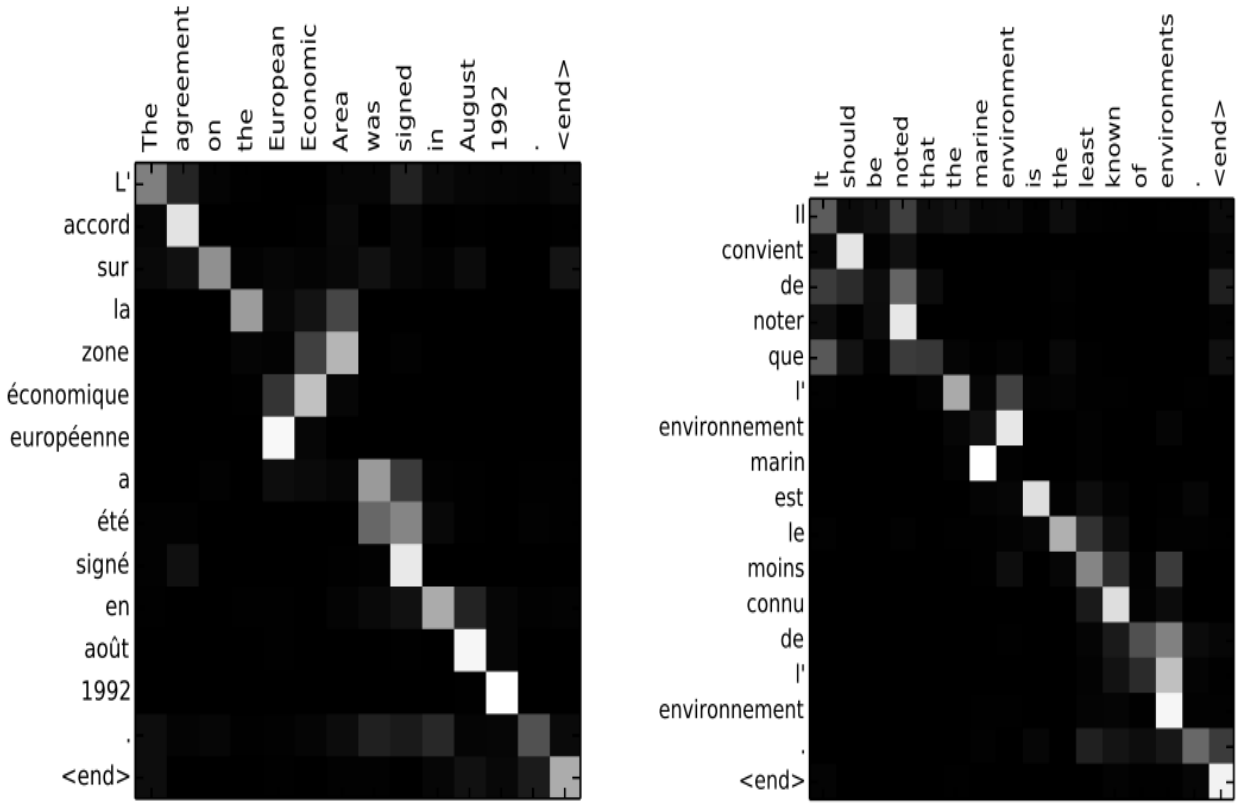
MT with attention

- Can use RNN, LSTM or GRU gates
- Encoder being used is the same structure as before
  - Can use uni-directional
  - Can use bi-directional
- Model can be trained using our regular back-propagation through time, all of the modules are differentiable

[Bahdanau et al., “Neural Machine Translation by Jointly Learning to Align and Translate”, ICLR 2015]

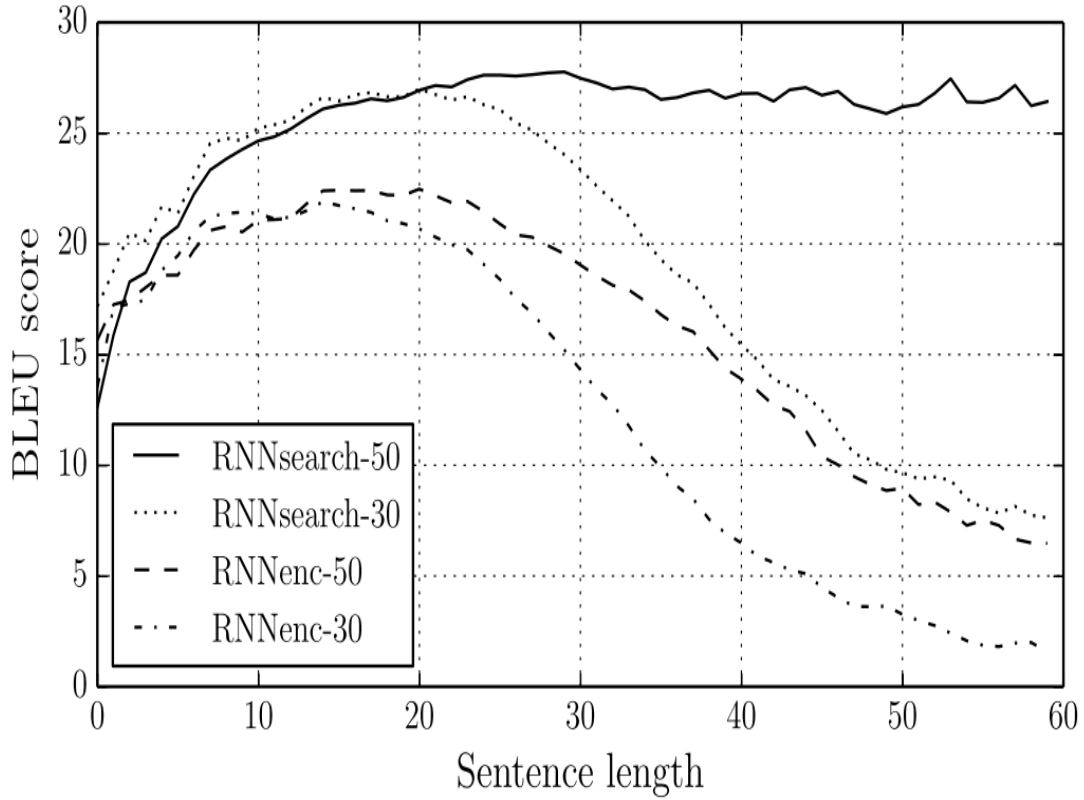
Does it work?

- Some sample alignments



Does it work?

- Especially good for long sentences



MT with attention recap

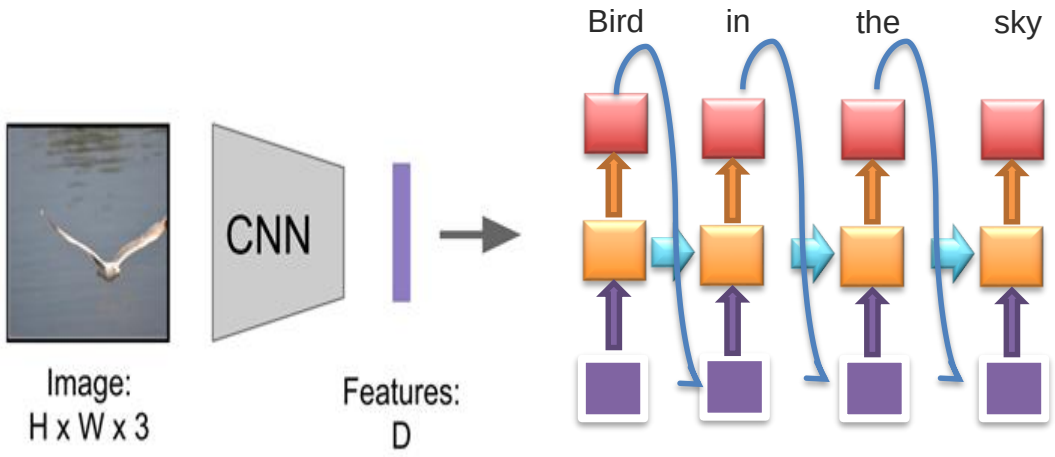
- Get good translation results (especially for long sentences)
- Also get a rough alignment of sentences in different languages – a nice way to confirm that the approach works

Visual captioning with soft attention

- A similar model but with a visual modality over which we pay attention

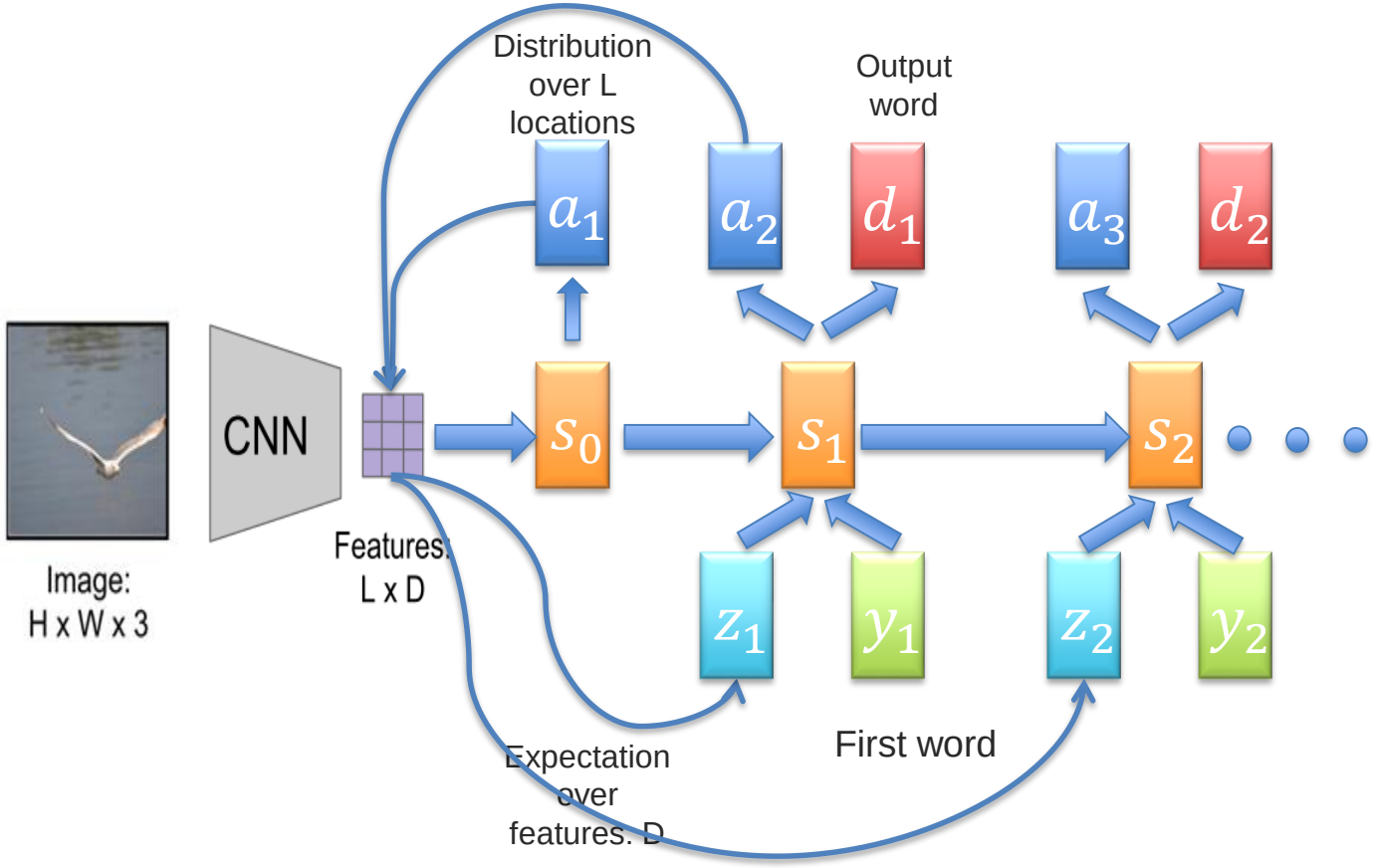
Xu et.al., ICML 2015

Image captioning with RNNs



Why might we not want to focus on the final layer?

Looking at more fine grained features



Visual captioning with soft attention

- Works pretty well – outperforms a number of baselines
- Allows us to get an idea of what the network “sees”
- A very similar model to that used for translation
- As well can be optimized using back propagation
  - Code is available https://github.com/kelvinxu/arctic-captions

Hard attention models

- Instead of allowing weighted sum allow the model to only look at one item only – Hard attention
- More difficult to implement
- Requires variational methods
- Similarities to Reinforcement learning [Mnih, 2014]

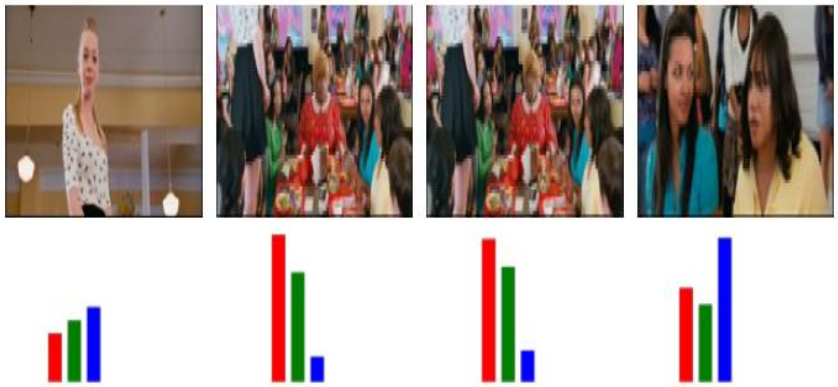
Other examples of latent alginment

Attention work - Good at paper naming

- Show, Attend and Tell (extension of Show and Tell)
- Listen, Attend and Walk
- Listen, Attend and Spell
- Ask, Attend and Answer

Video Descriptions

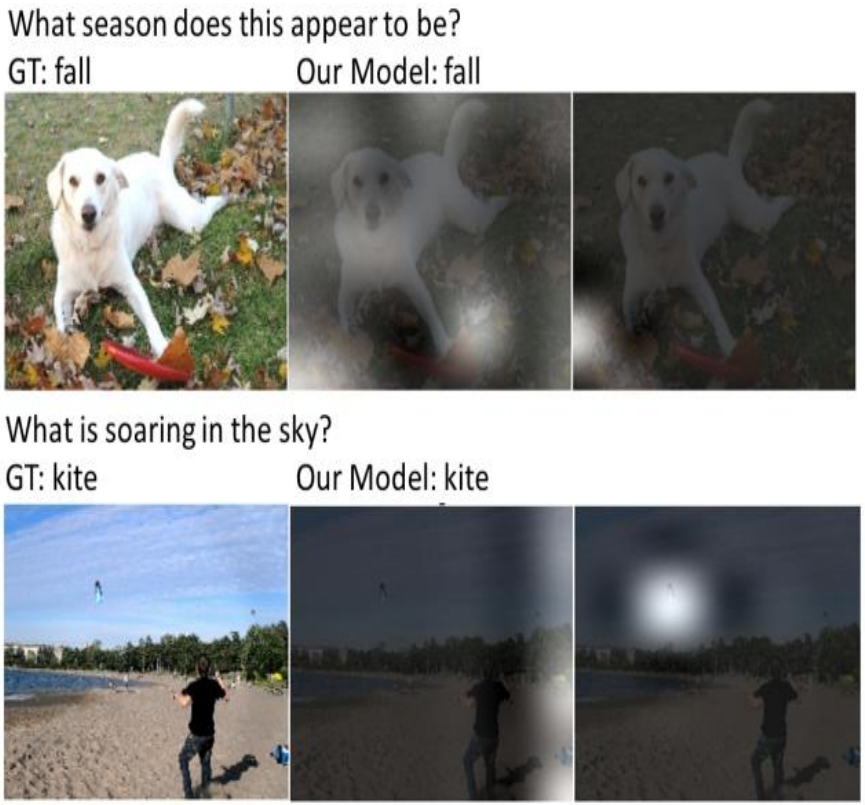
- Yao et al. 2015



- Soft attention model

VQA

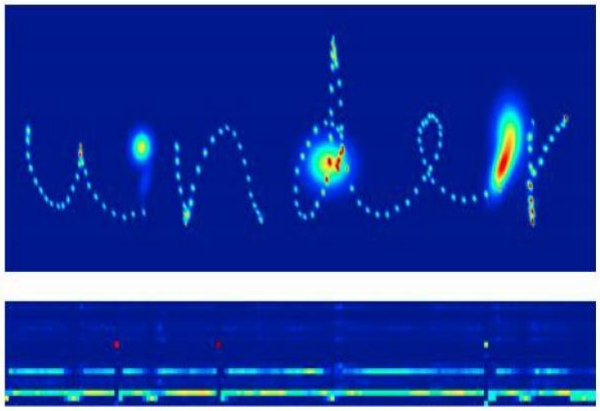
- Xu and Saenko, 2015



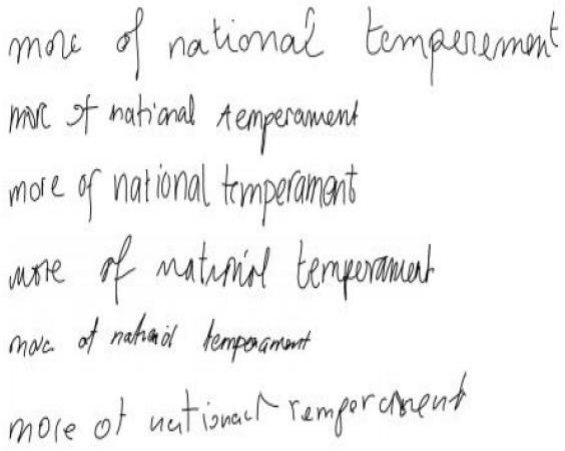
Speech

- Speech transcription – Listen, Attend and Spell [Chan et al.]
- Speech recognition - Attention-Based Models for Speech Recognition [Chorowski et al.]

Generative models with attention



Graves, "Generating Sequences with Recurrent Neural Networks", arXiv 2013



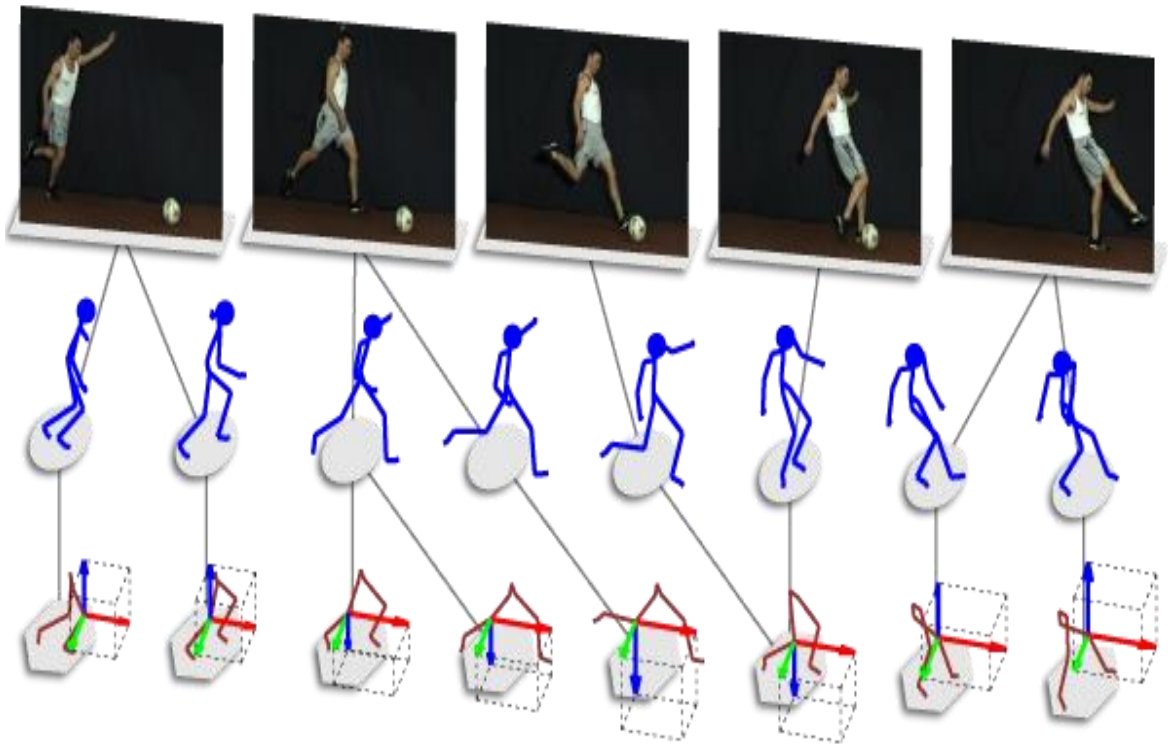
- DRAW paper from DeepMind

Explicit alignment

Other forms of alignment?

- Attention gave us a version of latent alignment it was not forced and does not always correspond to exact alignment of modalities but more where to pay attention
- We want to do proper alignment of multi-modal signals

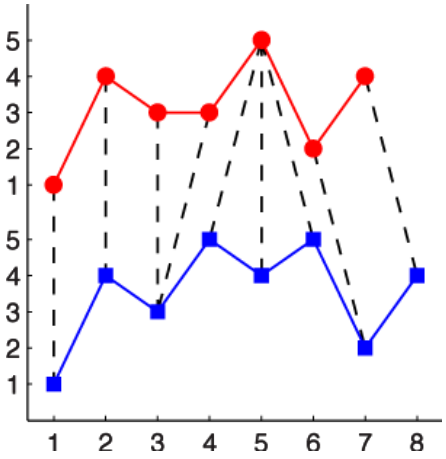
Our task



[Zhou and de la Tore, 2012]

- Re-aligning asynchronous data
- Finding similar data across modalities
- Event reconstruction from multiple sources

Let's start unimodal



$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_{n_x}] \in \mathbb{R}^{d \times n_x}$$

$$\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_{n_y}] \in \mathbb{R}^{d \times n_y}$$

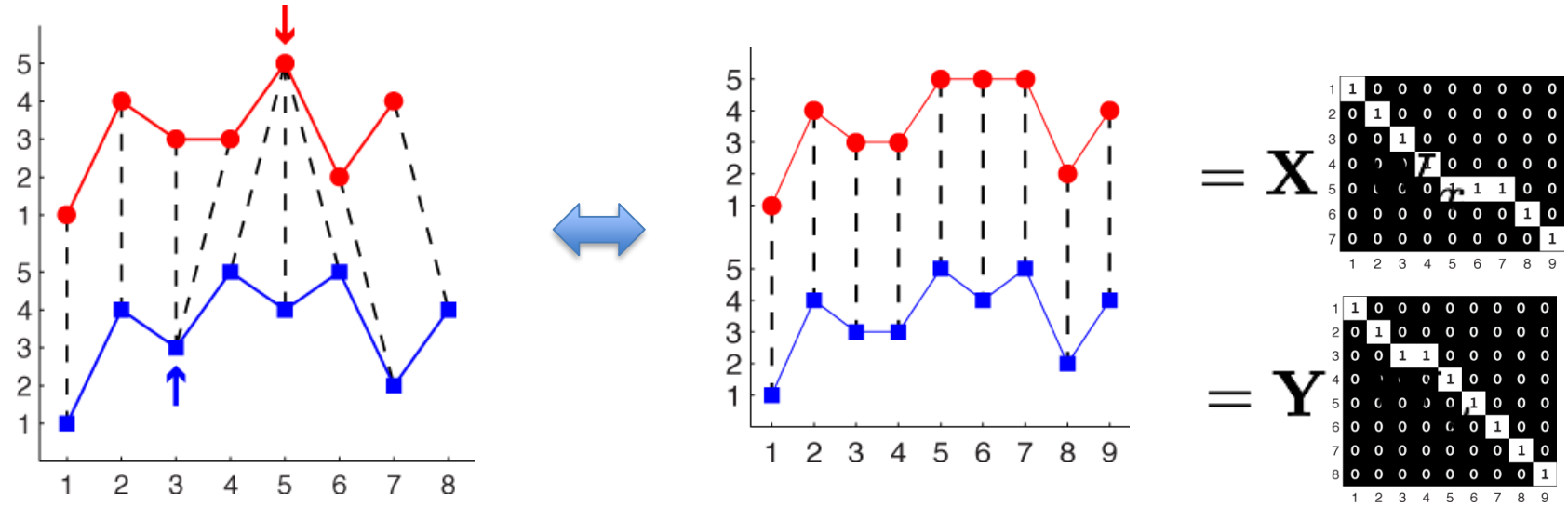
$$J_{dtw}(\mathbf{p}_x, \mathbf{p}_y) = \sum_{t=1}^{l} \|\mathbf{x}_{p_t^x} - \mathbf{y}_{p_t^y}\|_2^2$$

Dynamic Time Warping - DTW

$$J_{dtw}(\mathbf{p}_x, \mathbf{p}_y) = \sum_{t=1}^{l} \|\mathbf{x}_{p_t^x} - \mathbf{y}_{p_t^y}\|_2^2$$

$$J_{dtw}(\mathbf{W}_x, \mathbf{W}_y) = \|\mathbf{X}\mathbf{W}_x - \mathbf{Y}\mathbf{W}_y\|_F^2$$

Replication doesn't change the objective.



Dynamic Time Warping - DTW

same #rows, different #columns

$$\mathbf{X} \in \mathbb{R}^{d \times n_x}, \mathbf{Y} \in \mathbb{R}^{d \times n_y}$$

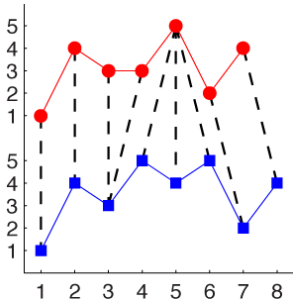
$$J_{dtw}(\mathbf{W}_x, \mathbf{W}_y) = \|\mathbf{X}\mathbf{W}_x - \mathbf{Y}\mathbf{W}_y\|_F^2$$

$$\mathbf{W}_x \in \{0,1\}^{n_x \times l}, \mathbf{W}_y \in \{0,1\}^{n_y \times l}$$

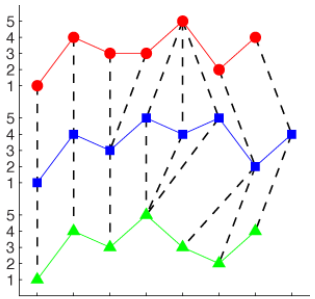
temporal alignment

DTW - limitations

- Complexity



$O(n_x n_y)$



$O(n_x n_y n_z)$

m sequences

$$O(\prod_{i=1}^{m} n_i)$$

- Unimodality

Canonical Correlation Analysis

- When data is normalized it is actually equivalent to smallest RMSE under certain restrictions [Zhou and de la Tore, 2012]

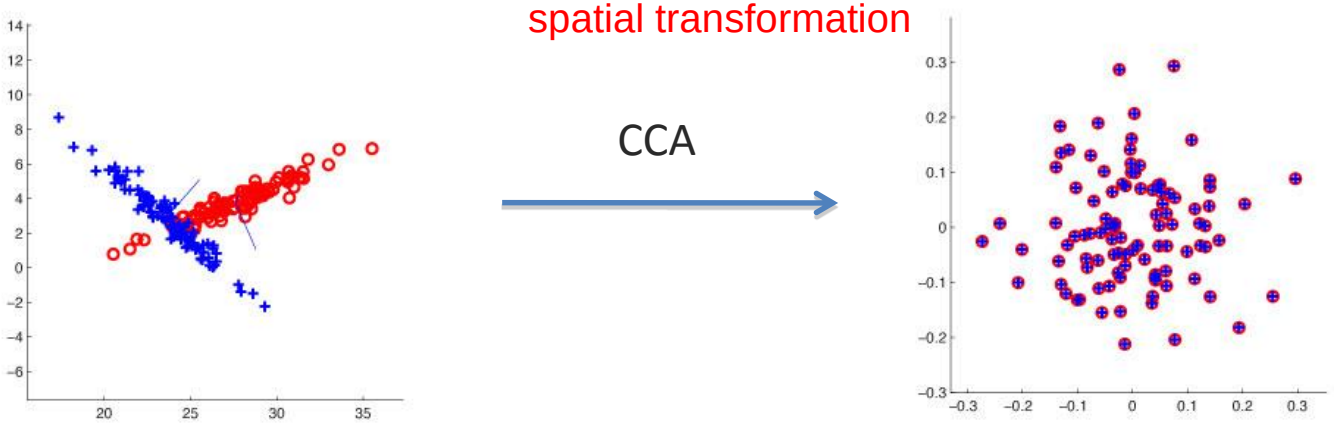
different #rows, same #columns

$$\mathbf{X} \in \mathbb{R}^{d_x \times n}, \mathbf{Y} \in \mathbb{R}^{d_y \times n}$$

$$J_{cca}(\mathbf{V}_x, \mathbf{V}_y) = \|\mathbf{V}_x^T \mathbf{X} - \mathbf{V}_y^T \mathbf{Y}\|_F^2$$

$$\mathbf{V}_x \in \mathbb{R}^{d_x \times b}, \mathbf{V}_y \in \mathbb{R}^{d_y \times b}$$

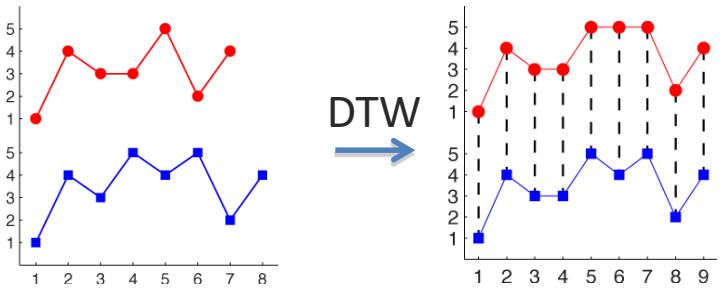
spatial transformation



Combining DTW and CCA

- Dynamic Time Warping (DTW)

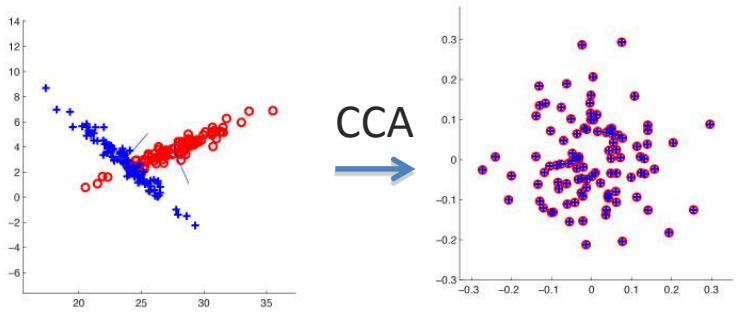
$$J_{dtw}(\mathbf{W}_x, \mathbf{W}_y) = \|\mathbf{X}\mathbf{W}_x - \mathbf{Y}\mathbf{W}_y\|_F^2$$



same feature, different #frames

- Canonical Correlation Analysis (CCA)

$$J_{cca}(\mathbf{V}_x, \mathbf{V}_y) = \|\mathbf{V}_x^T\mathbf{X} - \mathbf{V}_y^T\mathbf{Y}\|_F^2$$

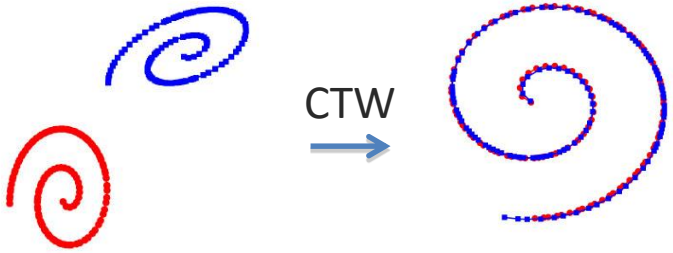


different feature, same #frames

Combining DTW and CCA

- Canonical Time Warping (CTW)

$$J_{ctw}(\mathbf{W}_x, \mathbf{W}_y, \mathbf{V}_x, \mathbf{V}_y) = \|\mathbf{V}_x^T \mathbf{X} \mathbf{W}_x - \mathbf{V}_y^T \mathbf{Y} \mathbf{W}_y\|_F^2$$



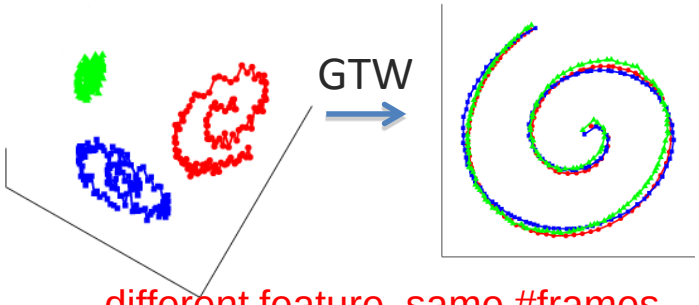
different feature, different #frames

[Zhou and de la Tore, 2012]

Combining DTW and CCA

- Generalized Time Warping (GTW)

$$J_{gtw}(\mathbf{W}_i, \mathbf{V}_i) = \sum_{i=1}^{m}\sum_{j=1}^{m} \frac{1}{2}\|\mathbf{V}_i^T\mathbf{X}_i\mathbf{W}_i - \mathbf{V}_j^T\mathbf{X}_j\mathbf{W}_j\|_F^2$$

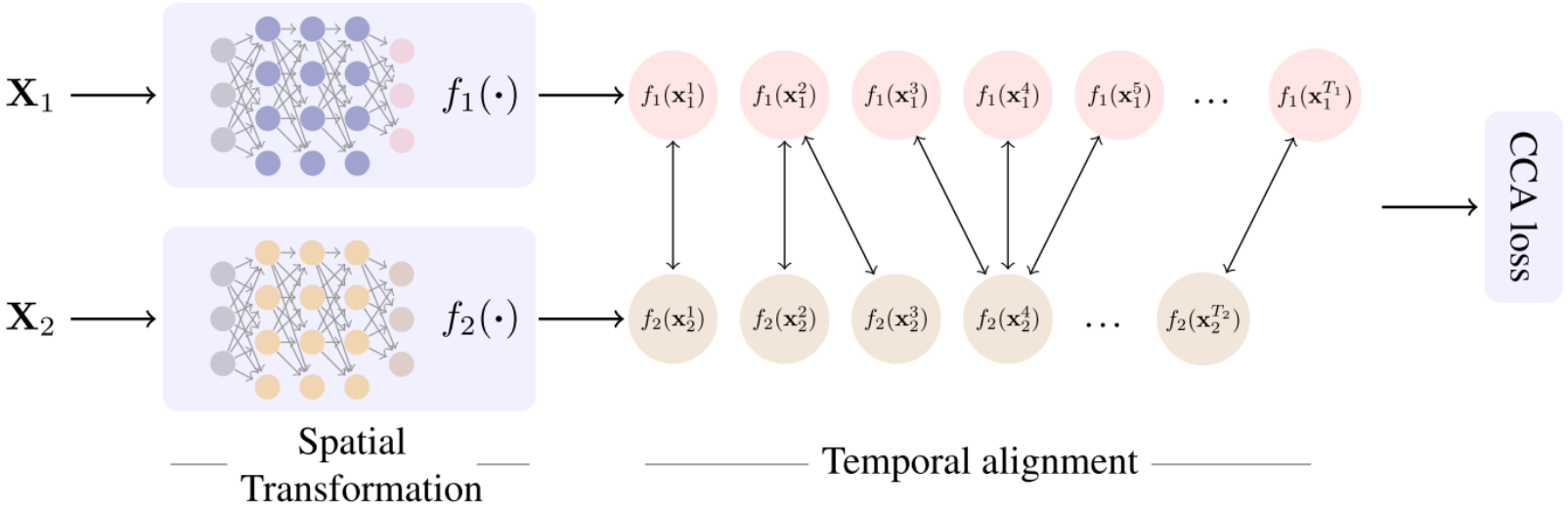


different feature, same #frames,
multiple time series

[Zhou and de la Tore, 2012]

Have a deep version as well

- Deep Canonical Time Warping



[Trigeorgis et al., CVPR 2016]

Multimodal fusion

Multimodal fusion

- Process of joining information from two or more modalities to perform a prediction
- Examples
  - Audio-visual speech recognition
  - Audio-visual emotion recognition
  - Multimodal biometrics
  - Speaker identification and diarization

(a) answer-phone (a) get-out-car (a) fight-person

Benefits

- Complementary information - McGurk effect
  - The sum is greater than the parts
- Robustness in presence of noise in one modality
- Dealing with missing or unobserved data in one of the modalities

Challenges and pitfalls

- Different sampling rates
- Different amounts of noise in modalities
- Potentially missing data in one modality
- One of the modalities not being informative
- Different predictive power of each modality
- Modalities only providing redundancy

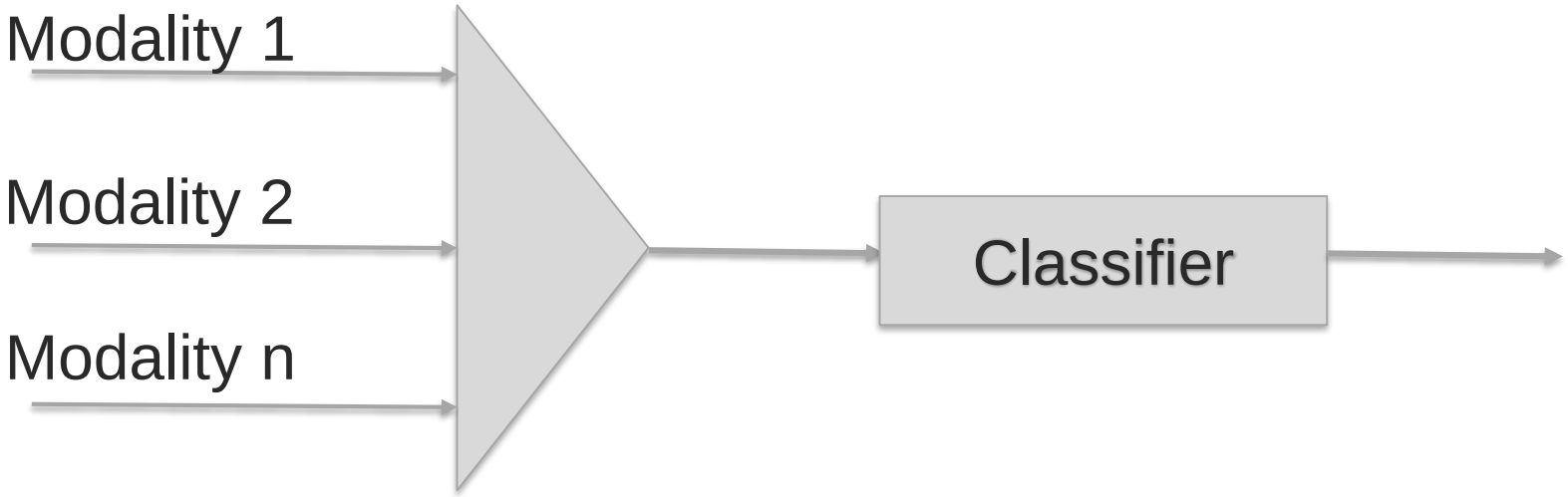
What happens when signals don't match?

[Aviezer, 2012]

Model free approaches

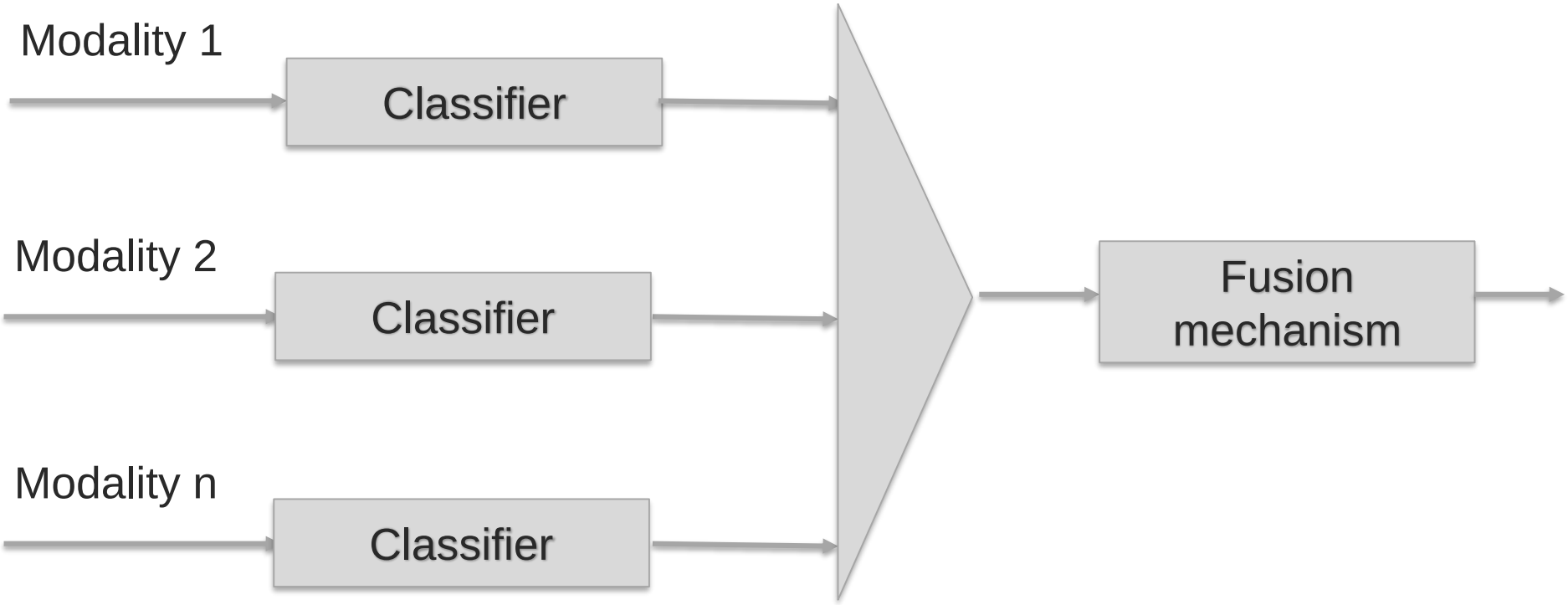
Model free approaches – early fusion

- Easy to implement – just concatenate the features
- More difficult to use if features have different framerates

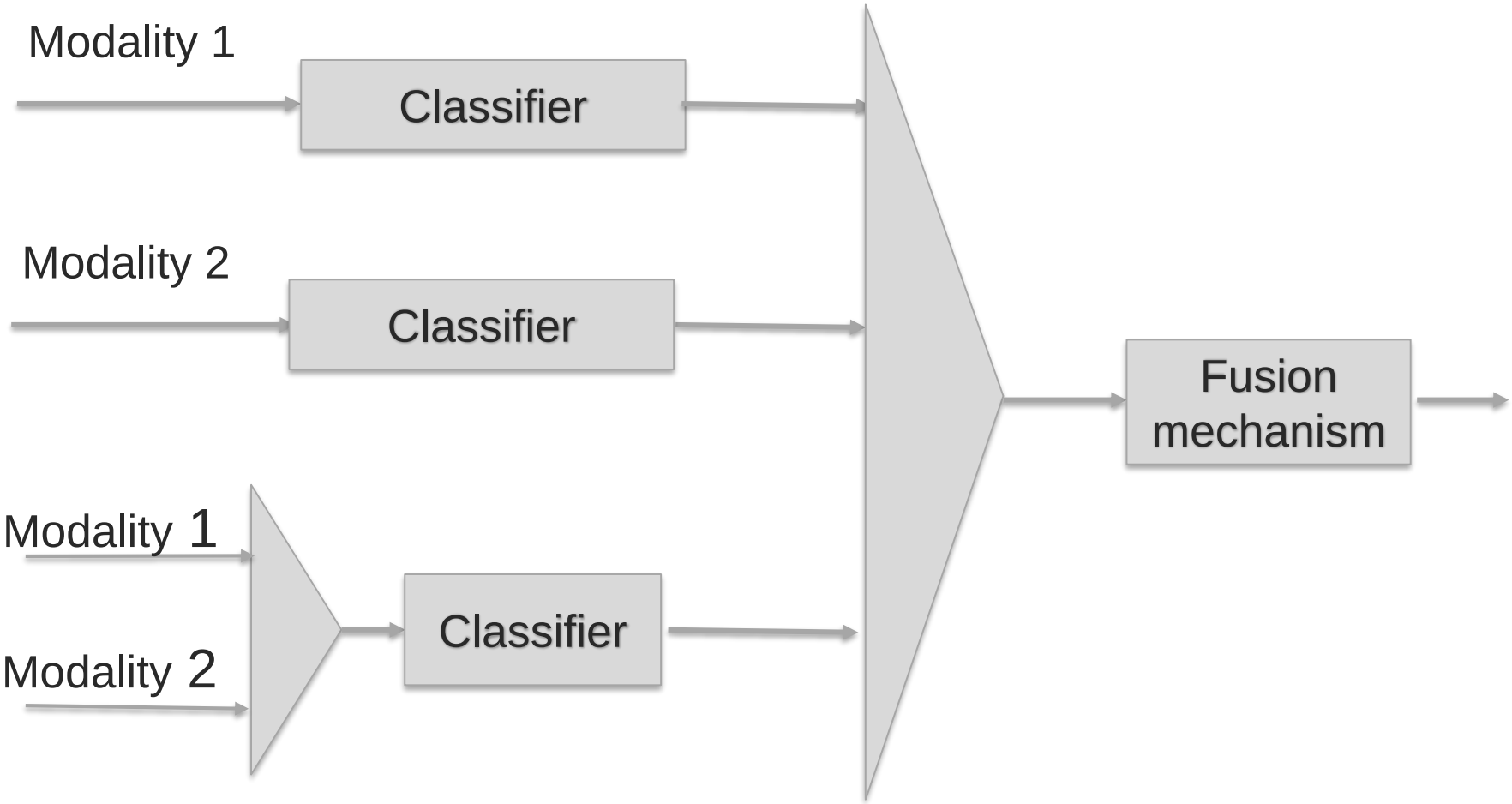


Model free approaches – late fusion

- Requires multiple training stages
- Do not model low level interactions between modalities
- Fusion mechanism can be voting, weighted sum or an ML approach



Model free approaches – hybrid fusion



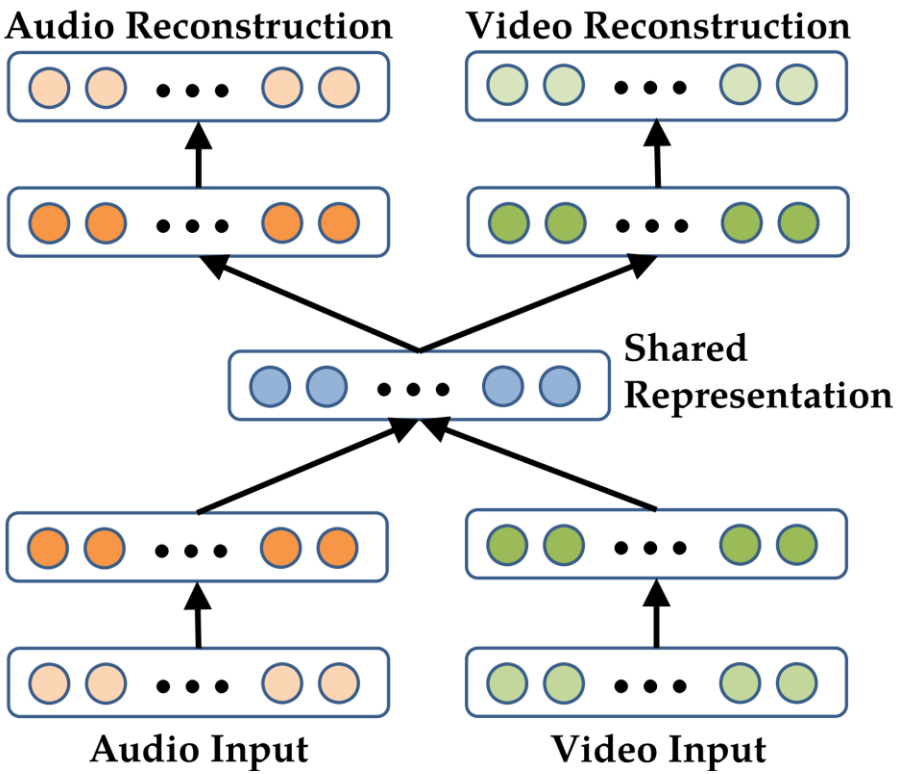
Model based fusion - Neural networks

Multimodal fusion using neural networks

- The fusion happens at some point in the neural network in an intermediate stage
- Line between representation and fusion is fuzzy

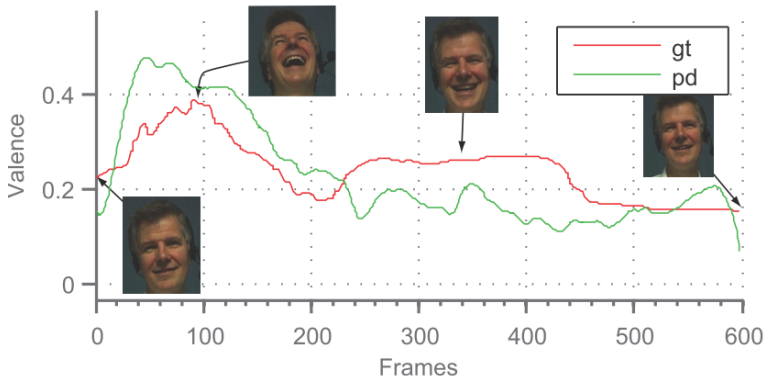
AVSR using neural networks

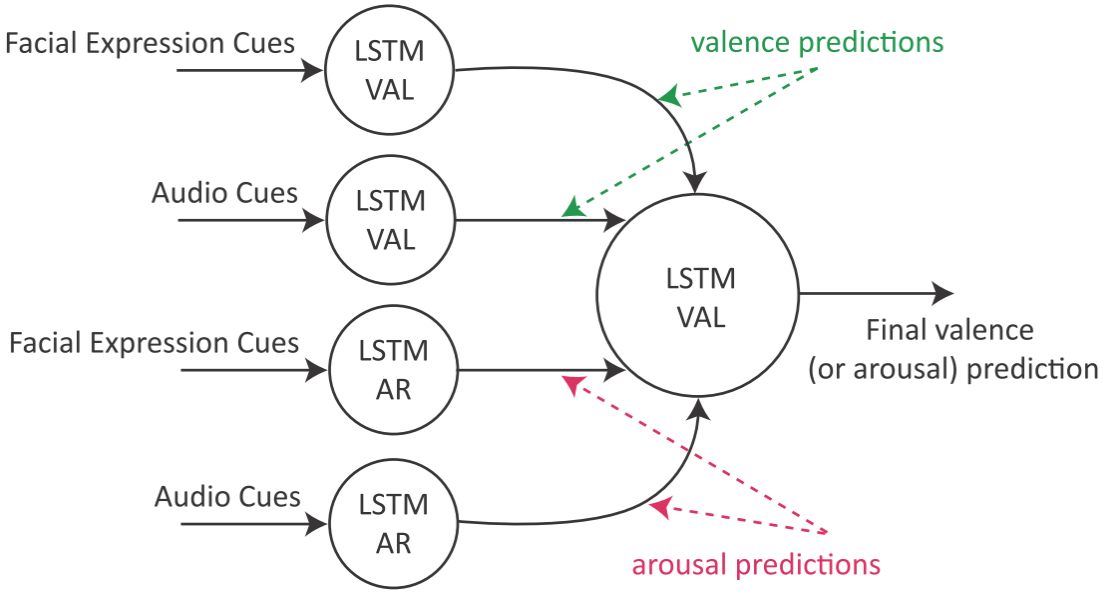
- Interestingly some late 80s early 90s work on this
- A more modern approach multimodal autoencoder [Ngiam et al., Multimodal Deep Learning, 2011]
- Fine-tuning an autoencoder learned representation using an AVSR task
- Where does the fusion actually happen?



Emotion recognition using LSTMs

- Using LSTM based fusion for audio-visual emotion recognition at each time step [Nicolaou 2011]

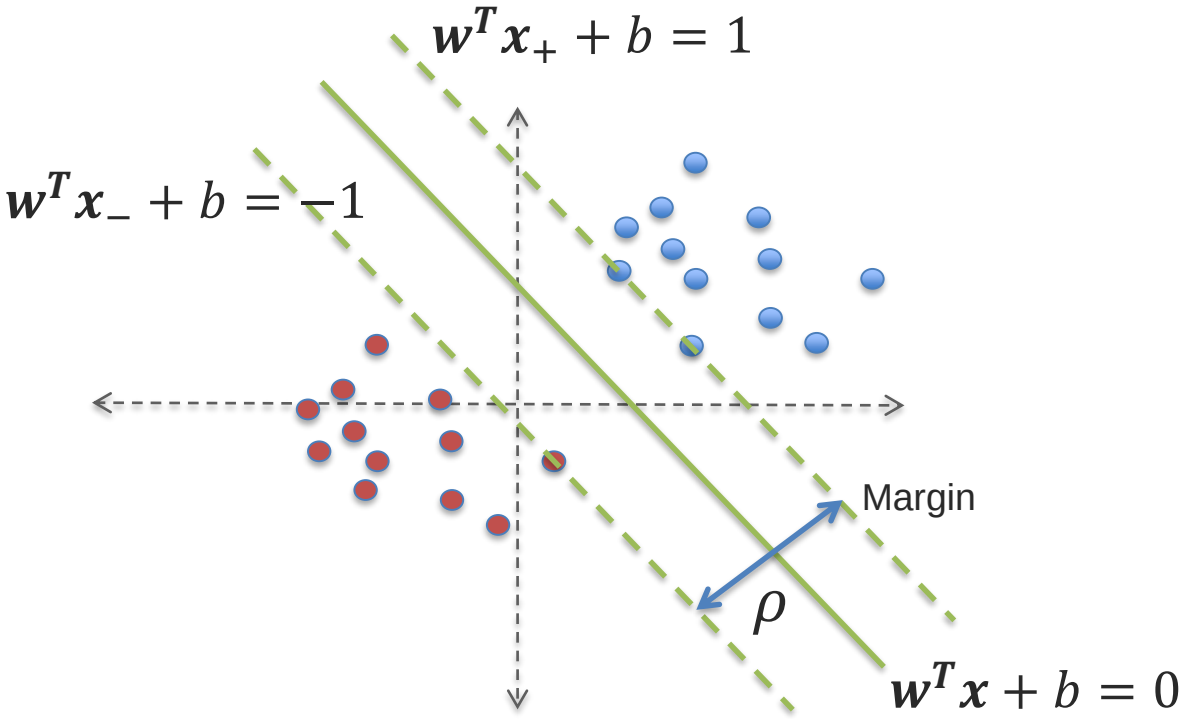




Kernel based approaches

Kernel support vector machines - refresher

- A crash course into kernel SVMs!



Primal form of Kernel SVM

minimize: $\phi(\boldsymbol{w}), \xi$

$$\frac{1}{2}\|\phi(\boldsymbol{w})\|^2 + C\sum_{i=1}^{M}\xi_i$$

$\phi(\boldsymbol{w})$ – a kernel function

subject to:

$$y_i(\phi(\boldsymbol{w})^T\phi(\boldsymbol{x}_i) + b) \geq 1 - \xi_i$$
$$\xi_i \geq 0$$

where: $\boldsymbol{x} \in \mathbb{R}^D, \boldsymbol{w} \in \mathbb{R}^D,$
$\phi(\boldsymbol{w}) \in \mathbb{R}^Z, \xi \in \mathbb{R}^M$

Careful, in some papers $\mathbf{w} \in \mathbb{R}^Z$, in others $\mathbf{w} \in \mathbb{R}^D$, and $\phi(\boldsymbol{w}) \in \mathbb{R}^Z$

prediction: $y = \text{sign}\left(\phi(\boldsymbol{w})^T\phi(\boldsymbol{x}) + b\right)$

1. Potentially too many parameters to optimize
2. Going to the Z space might be too computationally expensive

Dual form of Kernel SVM

minimize: $\min_{\boldsymbol{\alpha}} \frac{1}{2}\sum_i \sum_j \alpha_i \alpha_j y_i y_j \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_j) - \sum_i \alpha_i$

subject to: $\sum_i \alpha_i y_i = 0, \alpha_i \in [0, C]$

where: $b = y_i - \sum_j \alpha_j y_j \phi(\boldsymbol{x}_j)^T \phi(\boldsymbol{x}_i)$, for i where $\alpha_i > 0$

$\xi \in \mathbb{R}^M, \boldsymbol{\alpha} \in \mathbb{R}^M$

prediction: $y = \text{sign}\left(\sum_i \alpha_i y_i \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}) + b\right)$

What is a Kernel function

- What is a kernel function

  $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_j) = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle$, where $\phi: D \rightarrow Z$
- Kernel function performs an inner product in feature map space ϕ
- Inner product (a generalization of the dot product) is often denoted as $\langle ., . \rangle$ in SVM papers
- $\boldsymbol{x} \in \mathbb{R}^D$(but not necessarily), but $\phi(\boldsymbol{x})$ can be in any space – same, higher, lower or even in an infinite dimensional space
- Acts as a similarity metric between data points

Kernel properties

- Fortunately our dual only requires an inner product so we never need to go to the other space, we just need to be able to compute a dot-product in it
- We need a kernel value for all possible input combinations (across all $\boldsymbol{x}_i, \boldsymbol{x}_j$), we can store that in a matrix
- It's called a Gramian (or Gram) matrix, which by definition is positive semi definite, with non-negative eigenvalues

$$K_{i,j} = K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle$$

$$K = \begin{bmatrix} \langle \phi(\boldsymbol{x}_1), \phi(\boldsymbol{x}_1) \rangle & \langle \phi(\boldsymbol{x}_1), \phi(\boldsymbol{x}_2) \rangle & \cdots\cdots & \langle \phi(\boldsymbol{x}_1), \phi(\boldsymbol{x}_n) \rangle \\ \vdots & & \ddots & \vdots \\ \langle \phi(\boldsymbol{x}_n), \phi(\boldsymbol{x}_1) \rangle & \langle \phi(\boldsymbol{x}_n), \phi(\boldsymbol{x}_2) \rangle & \cdots\cdots & \langle \phi(\boldsymbol{x}_n), \phi(\boldsymbol{x}_n) \rangle \end{bmatrix} = \begin{bmatrix} \phi(\boldsymbol{x}_1)^T \\ \vdots \\ \phi(\boldsymbol{x}_n)^T \end{bmatrix} [\phi(\boldsymbol{x}_i) \quad \cdots\cdots \quad \phi(\boldsymbol{x}_n)]$$

Radial Basis Function Kernel (RBF)

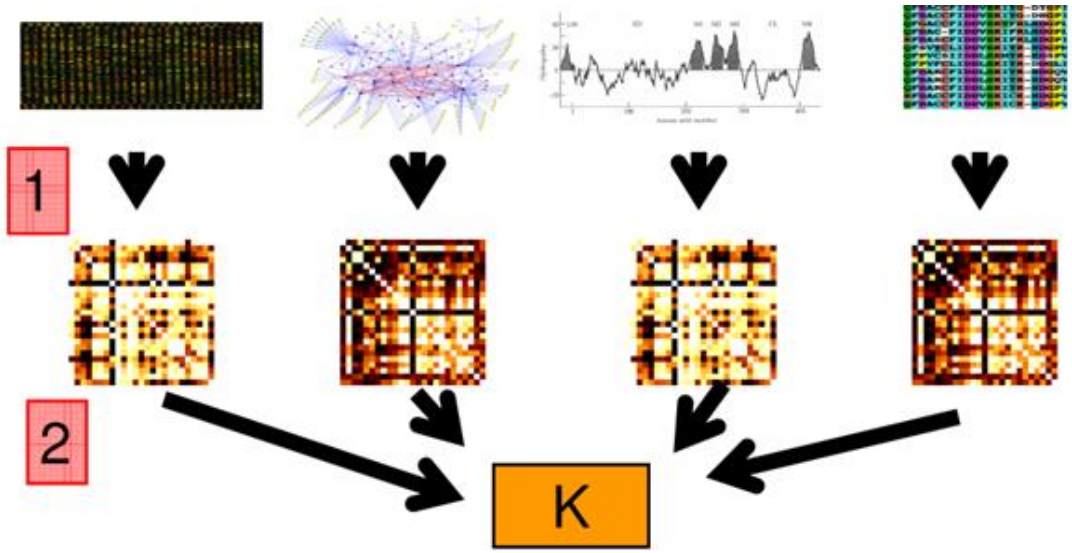
- Arguably the most popular SVM kernel
- $K(x_i, x_j) = \exp -\frac{1}{2\sigma^2}\|x_i - x_j\|^2$
- $\phi(\boldsymbol{x}) =$? It is infinite dimensional and fairly involved, no easy way to actually perform the mapping to this space, but we know what an inner product looks like in it
- $K = \exp\left(-\frac{1}{2\sigma^2} diag(X^T X)\mathbf{1}^T + \mathbf{1} diag(X^T X)^T - 2X^T X\right)$
- σ – a hyperparameter
- With a really low sigma the model becomes close to a KNN approach (potentially very expensive)

Different properties of different signals

- How do we deal with heterogeneous or multimodal data?
- The data of interest is not in a joint space so appropriate kernels might be different
- Multiple Kernel Learning (MKL) was popular for image classification and retrieval before deep learning approaches came around (winner of 2010 VOC challenge, ImageClef 2011 challenge) – not strictly multimodal but fusing different visual features in a single framework
- MKL - fell slightly out of favor when deep learning approaches became popular
- Still useful when large datasets are not available

Multiple Kernel Learning

- Instead of providing a single kernel and validating which one works optimize in a family of kernels (or different families for different modalities)
- Works as well for unimodal and multimodal data, very little adaptation is needed



[Lanckriet 2004]

Multiple Kernel Learning

- Allows to reduce amount of cross validation, (e.g. instead of using σ as a hyperparameter in RBF learn which values are important)
- Instead of feature selection throw all of them at MKL and let the kernels learn which ones are important
- Dealing with different format and scale data (real, ordinal, nominal)
- A technically sound way of combining features
- Feature combination and classifier training is done simultaneously
- Good learning bounds

MKL formulation primal

minimize: $\phi(\boldsymbol{w}), \boldsymbol{\beta}, \xi$

$$\frac{1}{2}\sum_{l=1}^{p}\frac{1}{\beta_l}\|\phi_l(\boldsymbol{w})\|^2 + C\sum_{i=1}^{M}\xi_i$$

subject to:

$$y_i\left(\sum_{l=1}^{p}\frac{1}{\beta_l}\langle\phi_l(\boldsymbol{w}),\phi_l(\boldsymbol{x}_i)\rangle + b\right) \geq 1-\xi_i$$

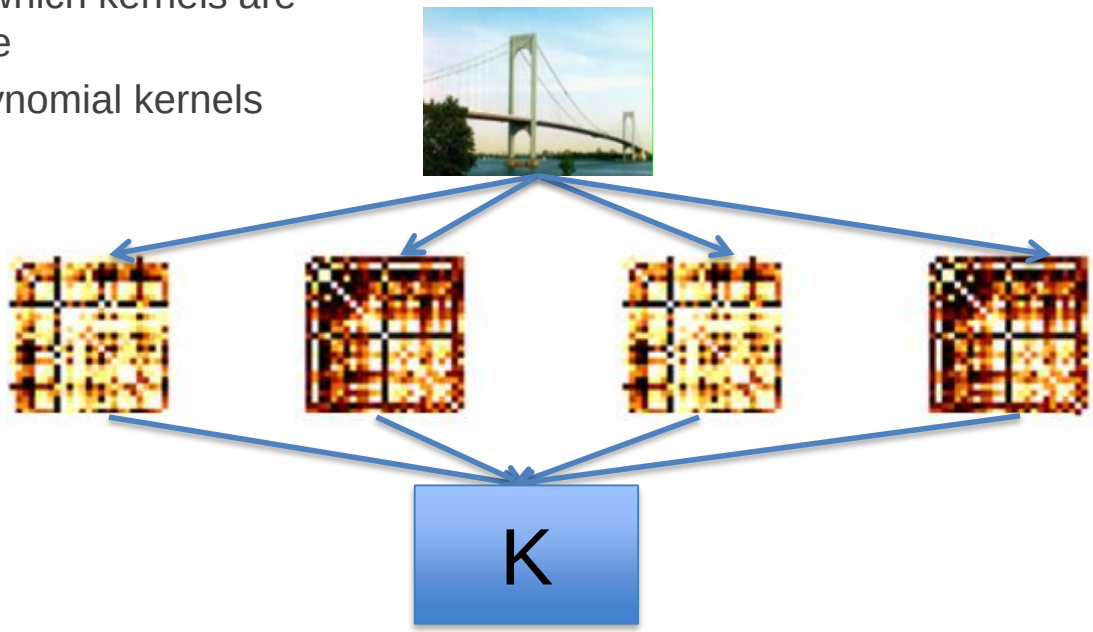
$$\xi_i \geq 0, \boldsymbol{\beta} \geq \mathbf{0}$$

where: $\boldsymbol{x} \in \mathbb{R}^D, \boldsymbol{w} \in \mathbb{R}^D, \xi \in \mathbb{R}^M$

prediction: $$y = \text{sign}\left(\sum_{l=1}^{p}\frac{1}{\beta_l}\langle\phi_l(\boldsymbol{w}),\phi_l(\boldsymbol{x})\rangle + b\right)$$

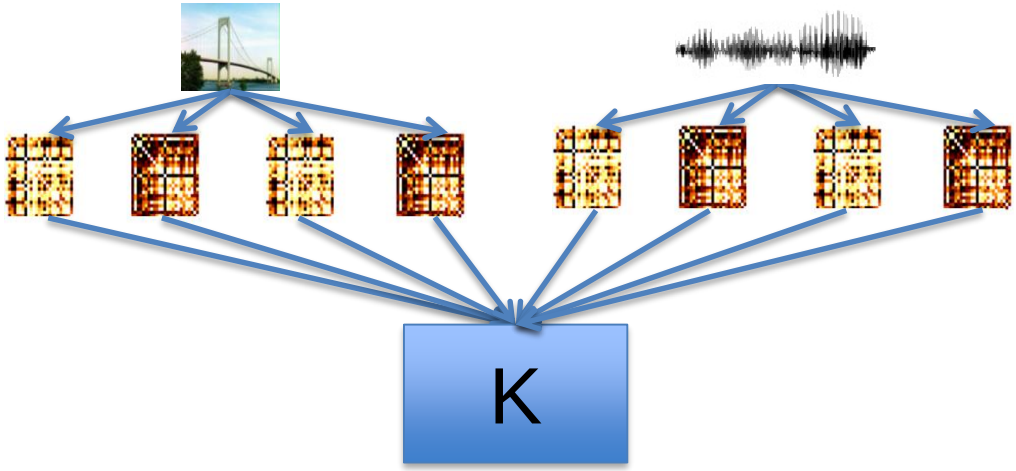
MKL in unimodal case

- Pick a family of kernels and learn which kernels are important for the classification case
- For example a set of RBF and polynomial kernels

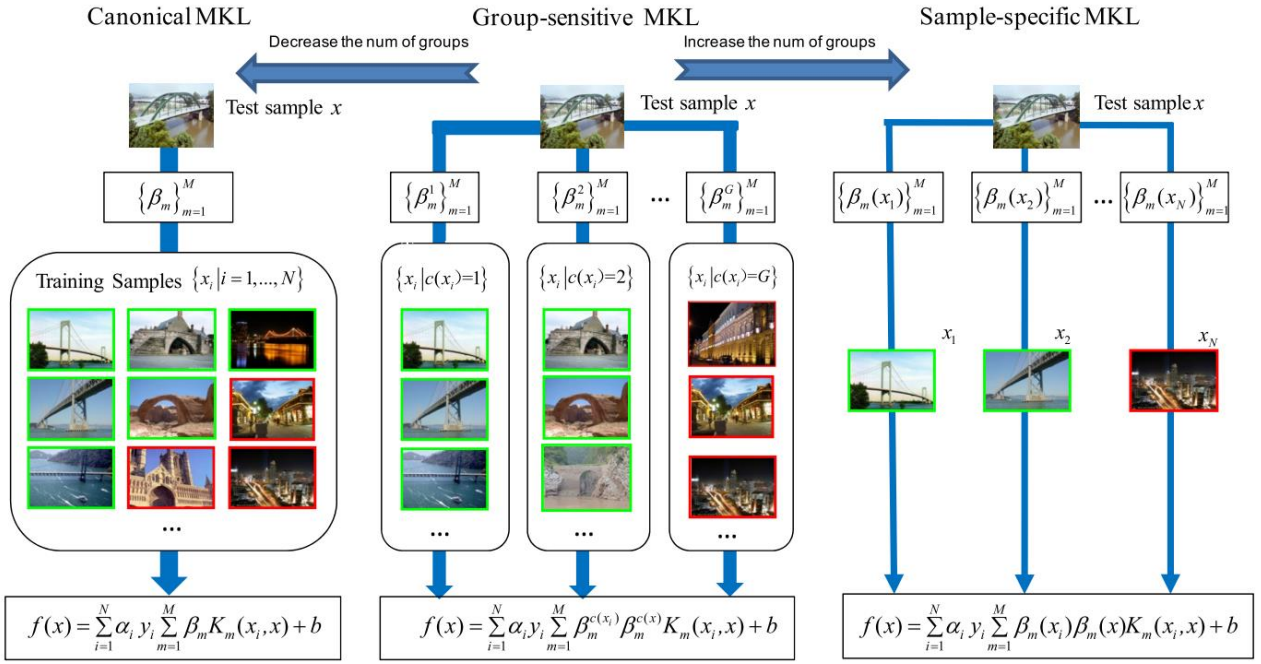


MKL in multimodal case

- Pick a family of kernels for each modality and learn which kernels are important for the classification case
- Does not need to be different modalities, often we use different views of the same modality (HOG, SIFT, etc.)



Different ways of picking kernels



Yang, 2009

Kernel CCA

Kernel CCA

- If we remember CCA it used only inner products in definitions when dealing with data, that means we can again use kernels
- We can now map into a high-dimensional non-linear space instead

$$(w_1^*, w_2^*) = \underset{w_1, w_2}{\operatorname{argmax}} \frac{w_1' \Sigma_{12} w_2}{\sqrt{w_1' \Sigma_{11} w_1 w_2' \Sigma_{22} w_2}} = \underset{w_1' \Sigma_{11} w_1 = w_2' \Sigma_{22} w_2 = 1}{\operatorname{argmax}} w_1' \Sigma_{12} w_2$$

$$(\alpha_1^*, \alpha_2^*) = \underset{\alpha_1, \alpha_2}{\operatorname{argmax}} \frac{\alpha_1' K_1 K_2 \alpha_2}{\sqrt{(\alpha_1' K_1^2 \alpha_2)\ (\alpha_1' K_2^2 \alpha_2)}} = \underset{\alpha_1' K_1^2 \alpha_1 = \alpha_2' K_2^2 \alpha_2 = 1}{\operatorname{argmax}} \alpha_1' K_1 K_2 \alpha_2,$$

[Lai et al. 2000]

Graphical models

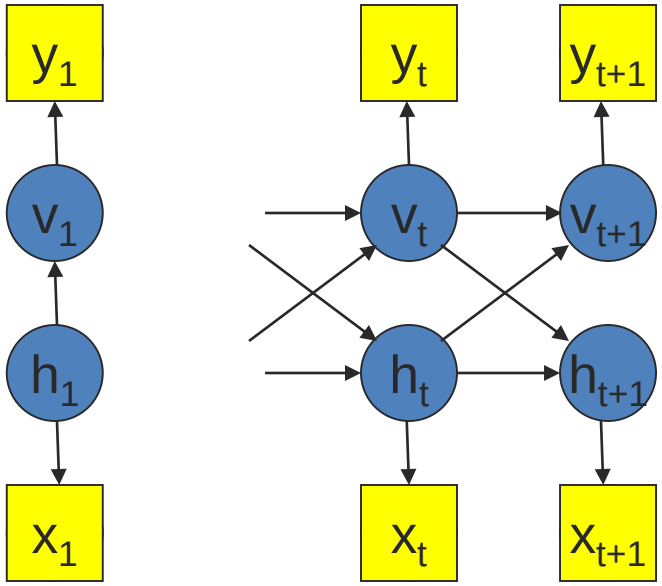
Multimodal fusion using graphical models

- Graphical models lend themselves nicely to multimodal fusion
- We can easily create links between modalities and outputs
- Can model latent dynamics between modalities as well
- Allows to incorporate expert knowledge

Generative and discriminative models

- Generative models model joint probability
  - Hidden Markov Models
  - Dynamic Bayesian Networks
  - Boltzmann Machines
- Discriminative models model conditional probability
  - Conditional Random Fields
  - Hidden Conditional Random Fields

The Coupled HMM



Video observations

Viseme states

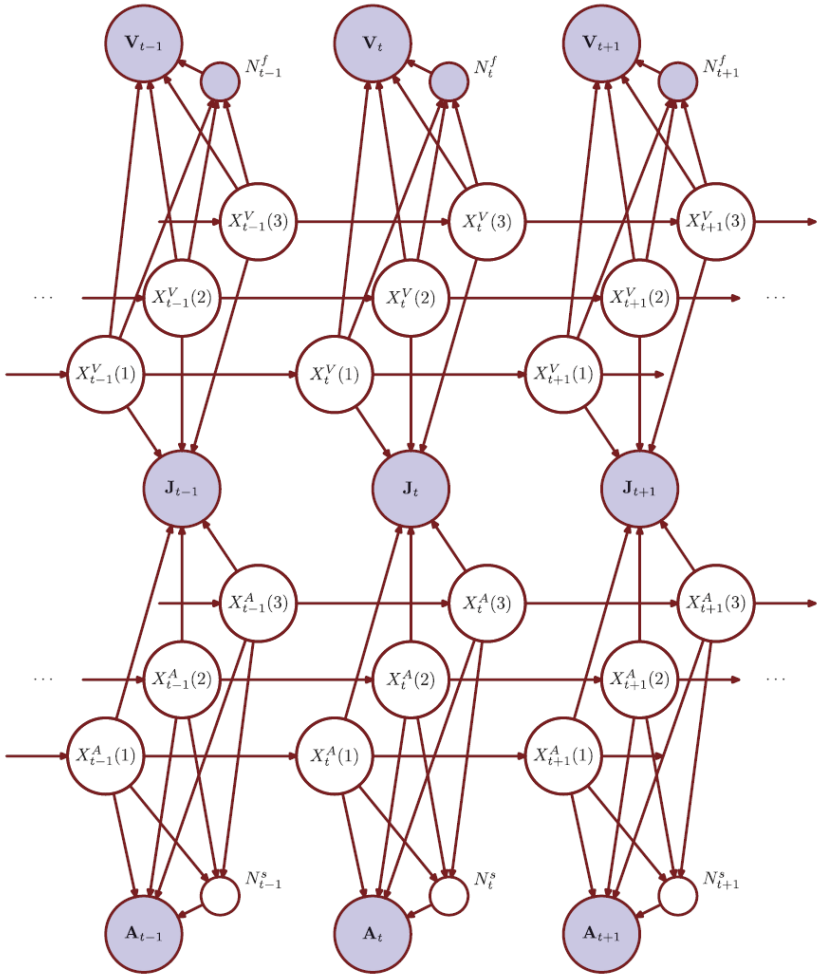
Audio phoneme states

Audio spectral observations

[Brand et al. 1997]

DBN for Audio-Visual Speaker Diarization

[Noulas et al., 2012, TPAMI]



The Overall Speaker Diarization Accuracy Achieved by Different Input Modalities

Method	IDIAP *A*	IDIAP *B*	Edinburgh	News
Wooters *et al.*	70%	70%	76%	77%
Audio Only	67%	67%	80%	72%
Multimodal	**84%**	**77%**	**89%**	**94%**

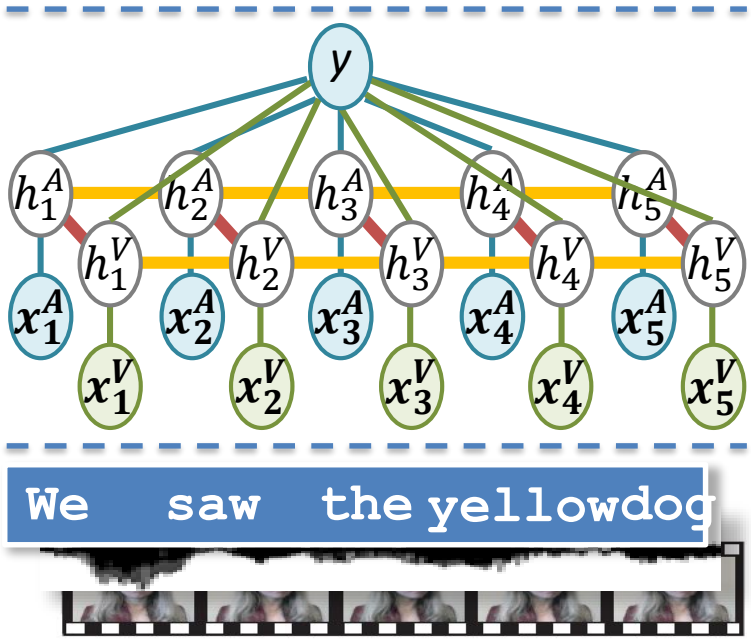
Multi-view Latent Variable Discriminative Models

Modality-*private* structure

- Internal grouping of observations

Modality-*shared* structure

- Interaction and synchrony



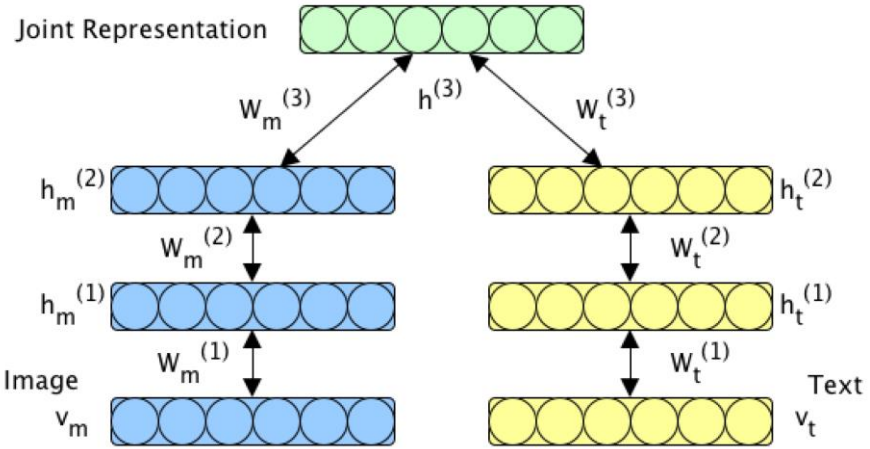
$$p(y|\, \boldsymbol{x}^{\boldsymbol{A}}, \boldsymbol{x}^{\boldsymbol{V}}; \boldsymbol{\theta}) = \sum_{\boldsymbol{h}^{\boldsymbol{A}}, \boldsymbol{h}^{\boldsymbol{V}}} p\left(y, \boldsymbol{h}^{\boldsymbol{A}}, \boldsymbol{h}^{\boldsymbol{V}} \middle| \boldsymbol{x}^{\boldsymbol{A}}, \boldsymbol{x}^{\boldsymbol{V}}; \boldsymbol{\theta}\right)$$

[Song, 2012]

➢ Approximate inference using loopy-belief

Deep Multimodal Boltzmann machines

- Generative model
- Individual modalities trained like a DBN
- Multimodal representation trained using Variational approaches
- Used for image tagging and cross-media retrieval
- Reconstruction of one modality from another is a bit more “natural” than in autoencoder representation
- Can actually sample text and images

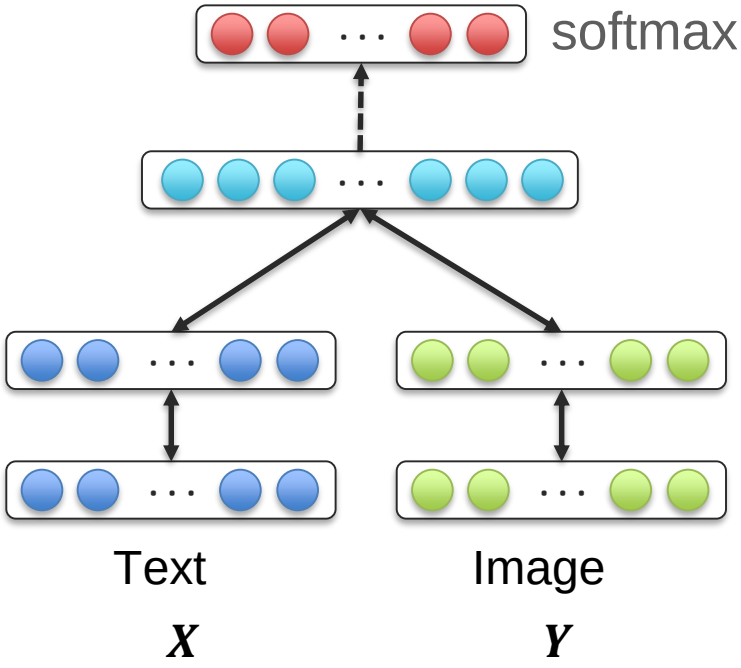


[Srivastava and Salakhutdinov, Multimodal Learning with Deep Boltzmann Machines, 2012, 2014]

Recap - Multimodal Representation Learning

Learn (unsupervised) a joint representation between multiple modalities where similar unimodal concepts are closely projected.

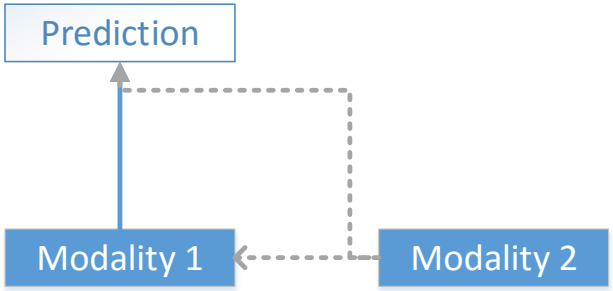
- Deep Multimodal Boltzmann machines



Co-learning

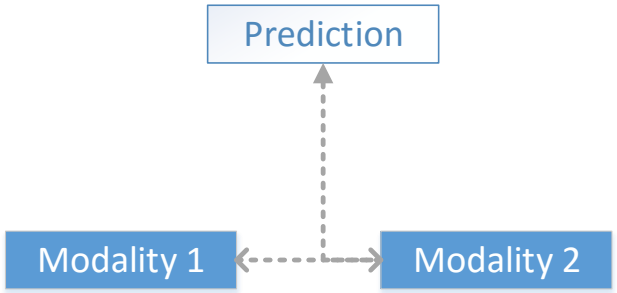
Co-Learning

A Unidirectional co-learning:



- One modality may have more resources
- Related to bootstrapping and domain adaptation

B Bidirectional co-learning:



- Famous example: co-training [Blum & Mitchell]
- Strong correspondence requirement to be succesful

Co-learning

- Revival of this with resurgence of deep learning
- Transfer learning
- Pre-training and fine tuning

Concluding remarks

Multimodal machine learning

- Talked about the past and the present of multimodal machine learning
- Introduced five challenges facing it
  - Representation
  - Translation
  - Alignment
  - Fusion
  - Co-training

Future of Multimodal machine learning

- What are the challenges that need to be addressed?
- What datasets are required for that?
- Any new and interesting application areas?