Language Technologies Institute

Carnegie Mellon University

Tutorial on Multimodal Machine Learning

Louis-Philippe (LP) Morency
Tadas Baltrusaitis

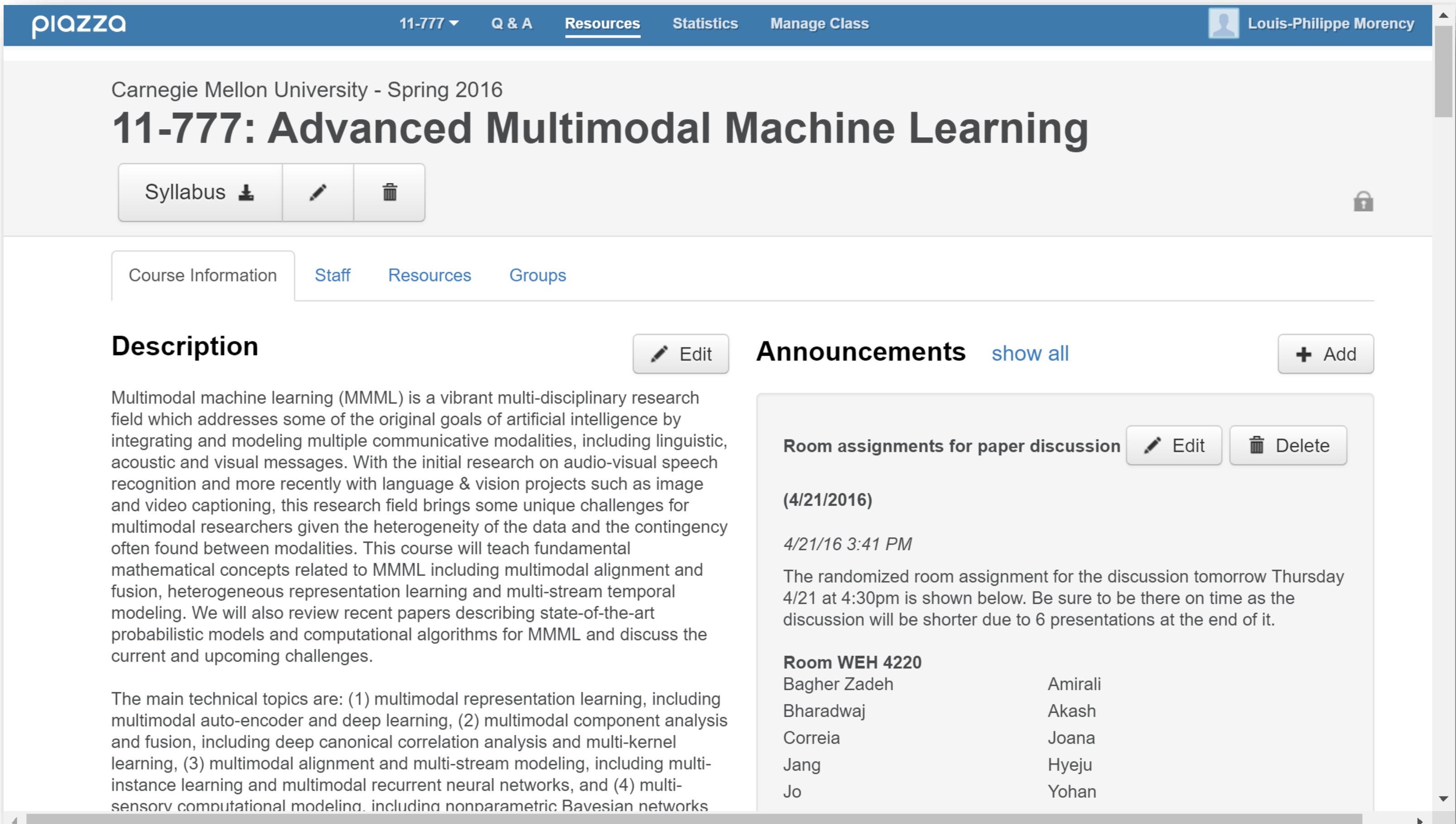
CMU Multimodal Communication and Machine Learning Laboratory [MultiComp Lab]

Your Instructors

Louis-Philippe Morency
morency@cs.cmu.edu

Tadas Baltrusaitis
tbaltrus@cs.cmu.edu

CMU Course 11-777: Multimodal Machine Learning



Tutorial Schedule

- **Introduction**
  - What is Multimodal?
    - Historical view, multimodal vs multimedia
  - Why multimodal
    - Multimodal applications: image captioning, video description, AVSR,…
  - Core technical challenges
    - Representation learning, translation, alignment, fusion and co-learning

Tutorial Schedule

- **Basic concepts – Part 1**
  - Linear models
    - Score and loss functions, regularization
  - Neural networks
    - Activation functions, multi-layer perceptron
  - Optimization
    - Stochastic gradient descent, backpropagation

Tutorial Schedule

- **Unimodal representations**
  - Visual representations
    - Convolutional neural networks
  - Acoustic representations
    - Spectrograms, autoencoders

Tutorial Schedule

- **Multimodal representations**
  - Joint representations
    - Visual semantic spaces, multimodal autoencoder
    - Tensor fusion representation
  - Coordinated representations
    - Similarity metrics, canonical correlation analysis
- Coffee break [20 mins]

Tutorial Schedule

- **Basic concepts – Part 2**
  - Recurrent neural networks
    - Long Short-Term Memory models
  - Optimization
    - Backpropagation through time

Tutorial Schedule

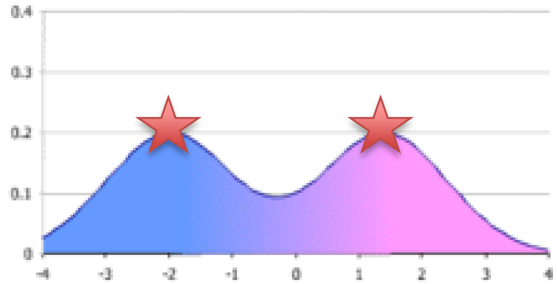
- **Translation and alignment**
  - Translation applications
    - Machine translation, image captioning
  - Explicit alignment
    - Dynamic time warping, deep canonical time warping
  - Implicit alignment
    - Attention models, multi instance learning
    - Temporal attention-gated model

Tutorial Schedule

- **Multimodal fusion**
  - Model free approaches
    - Early and late fusion, hybrid models
  - Kernel-based fusion
    - Multiple kernel learning
  - Multimodal graphical models
    - Factorial HMM, Multi-view Hidden CRF
    - Multi-view LSTM model

What is Multimodal?

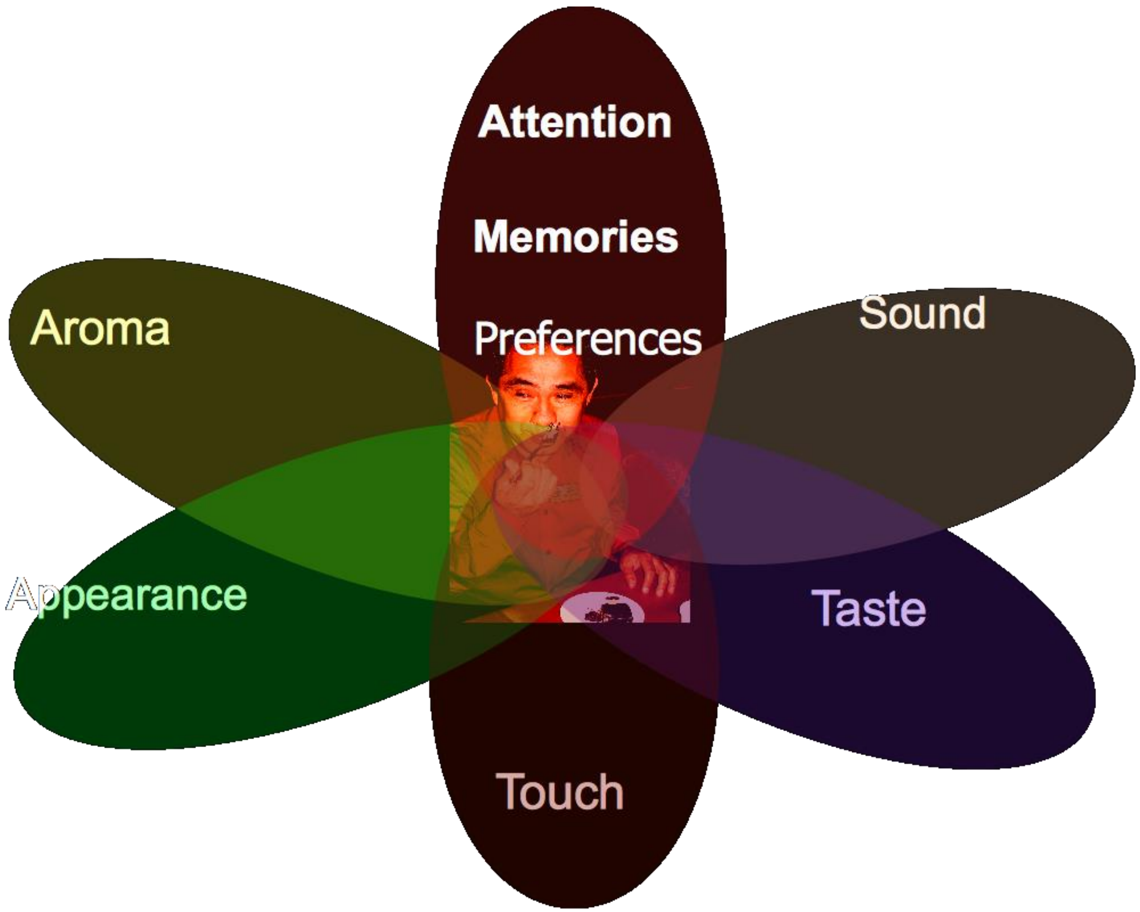
What is Multimodal?



Multimodal distribution

- Multiple modes, i.e., distinct "peaks" (local maxima) in the probability density function

What is Multimodal?



Sensory Modalities

What is Multimodal?

Modality

The way in which something happens or is experienced.

- Modality refers to a certain type of information and/or the representation format in which information is stored.
- Sensory modality: one of the primary forms of sensation, as vision or touch; channel of communication.

Medium ("middle")

A means or instrumentality for storing or communicating information; system of communication/transmission.

- Medium is the means whereby this information is delivered to the senses of the interpreter.

Examples of Modalities

- Natural language (both spoken or written)
- Visual (from images or videos)
- Auditory (including voice, sounds and music)
- Haptics / touch
- Smell, taste and self-motion
- Physiological signals
  - Electrocardiogram (ECG), skin conductance
- Other modalities
  - Infrared images, depth images, fMRI

Multimodal Communicative Behaviors

Verbal

Lexicon
- Words

Syntax
- Part-of-speech
- Dependencies

Pragmatics
- Discourse acts

Vocal

Prosody
- Intonation
- Voice quality

Vocal expressions
- Laughter, moans

Visual

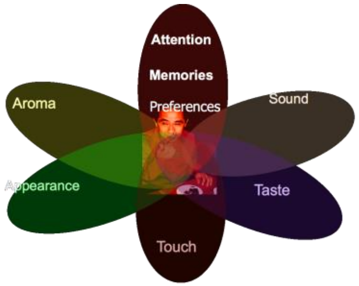
Gestures
- Head gestures
- Eye gestures
- Arm gestures

Body language
- Body posture
- Proxemics

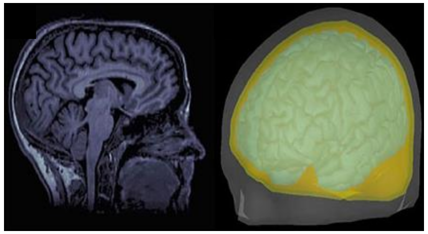
Eye contact
- Head gaze
- Eye gaze

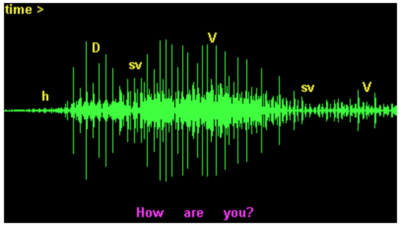
Facial expressions
- FACS action units
- Smile, frowning

Multiple Communities and Modalities



Psychology


Medical

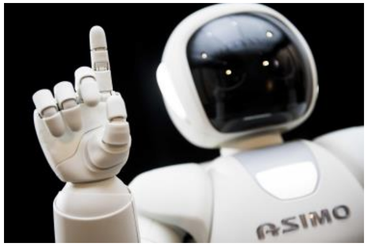


Speech

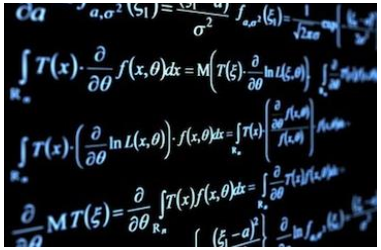

Vision

Language


Multimedia



Robotics

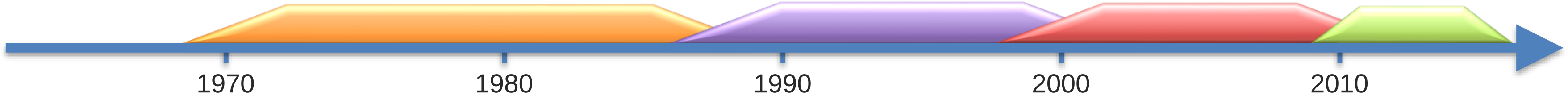

Learning

A Historical View

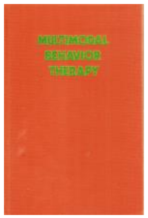
Prior Research on “Multimodal”

Four eras of multimodal research

- The “behavioral” era (1970s until late 1980s)
- The “computational” era (late 1980s until 2000)
- The “interaction” era (2000 - 2010)
- The “deep learning” era (2010s until …)
  - Main focus of this tutorial



The "Behavioral" Era (1970s until late 1980s)



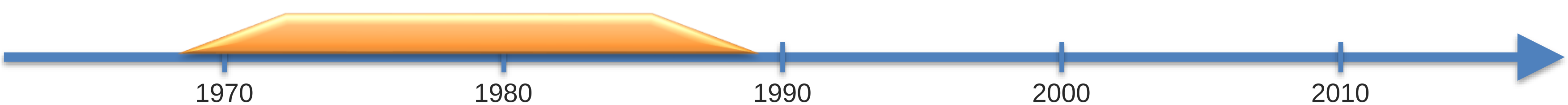
Multimodal Behavior Therapy by Arnold Lazarus [1973]

➢ 7 dimensions of personality (or modalities)

Multi-sensory integration (in psychology):

- Multimodal signal detection: Independent decisions vs. integration [1980]
- Infants' perception of substance and temporal synchrony in multimodal events [1983]
- A multimodal assessment of behavioral and cognitive deficits in abused and neglected preschoolers [1984]

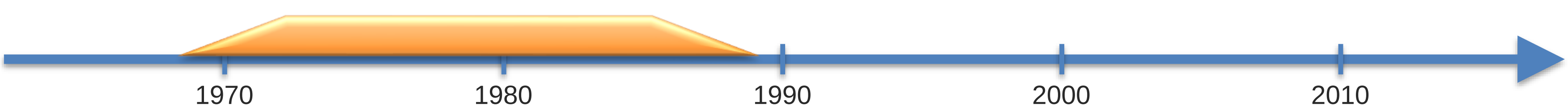
❑ TRIVIA: Geoffrey Hinton received his B.A. in Psychology ☺



Language and Gestures

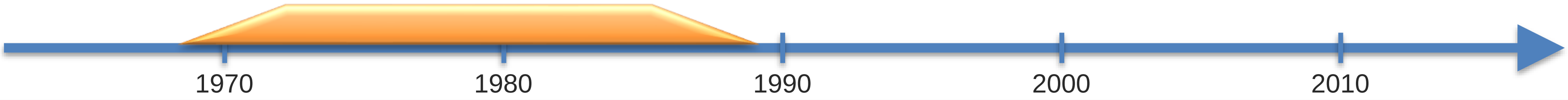
David McNeill
University of Chicago
Center for Gesture and Speech Research

“For McNeill, gestures are in effect the speaker’s thought in action, and integral components of speech, not merely accompaniments or additions.”



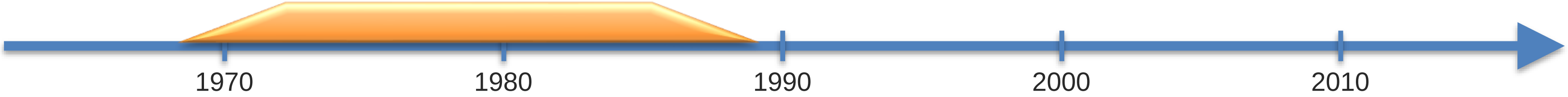
The McGurk Effect (1976)

[Hearing lips and seeing voices – Nature]



The McGurk Effect (1976)

[Hearing lips and seeing voices – Nature]



➢ The “Computational” Era(Late 1980s until 2000)

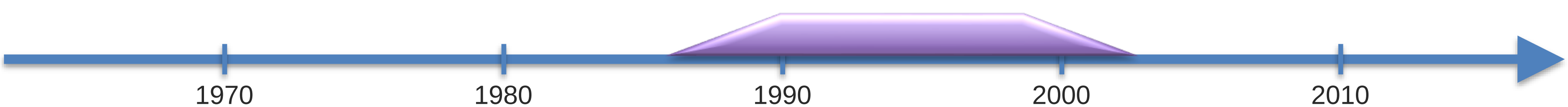
1) Audio-Visual Speech Recognition (AVSR)

- Motivated by the McGurk effect
- First AVSR System in 1986

  “Automatic lipreading to enhance speech recognition”
- Good survey paper [2002]

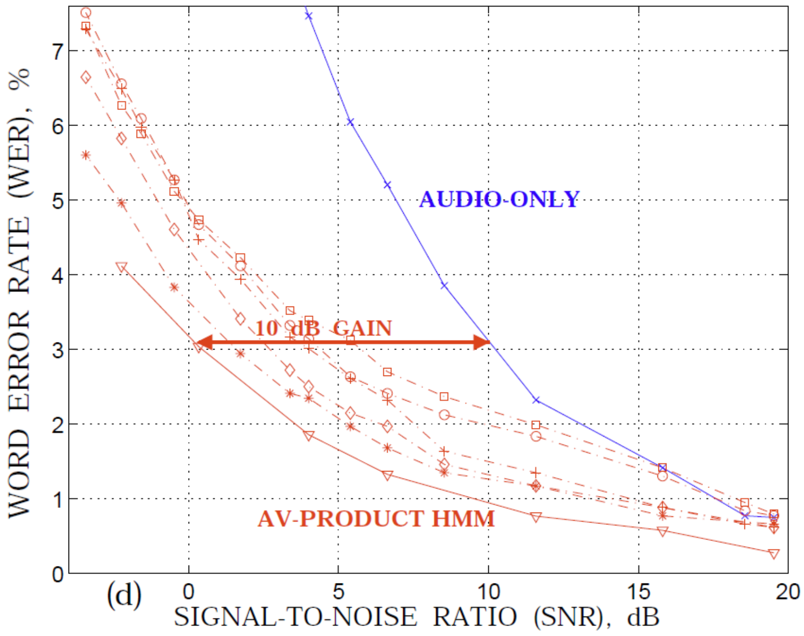
  “Recent Advances in the Automatic Recognition of Audio-Visual Speech”

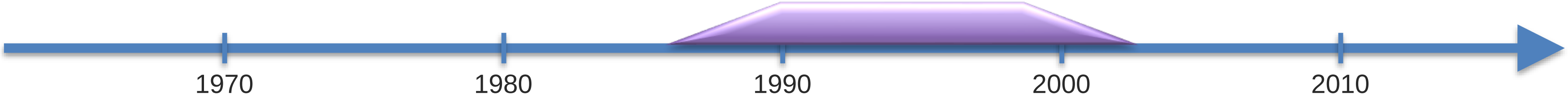
❑ TRIVIA: The first multimodal deep learning paper was about audio-visual speech recognition [ICML 2011]



➢ The "Computational" Era(Late 1980s until 2000)

1) Audio-Visual Speech Recognition (AVSR)



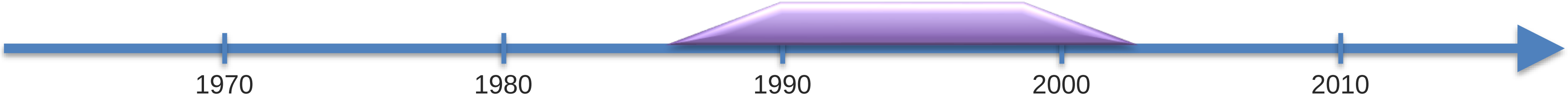


➢ The “Computational” Era (Late 1980s until 2000)

2) Multimodal/multisensory interfaces

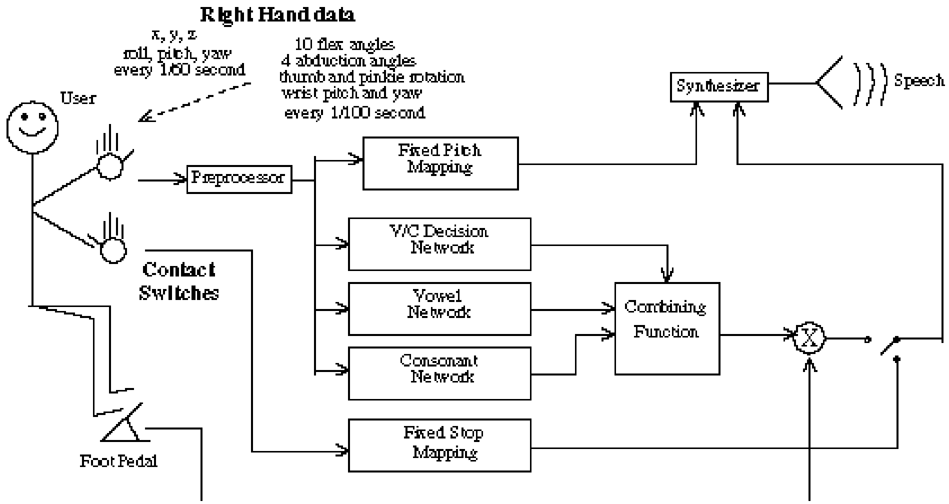
- Multimodal Human-Computer Interaction (HCI)

“Study of how to design and evaluate new computer systems where human interact through multiple *modalities, including both input and output modalities.”*

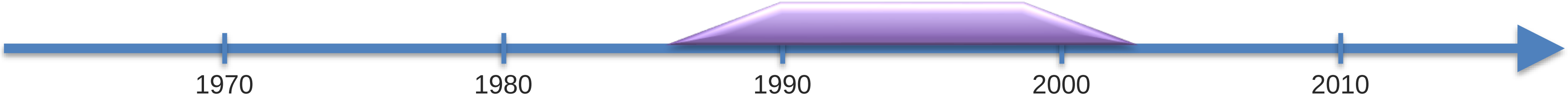


➢ The "Computational" Era (Late 1980s until 2000)

2) Multimodal/multisensory interfaces

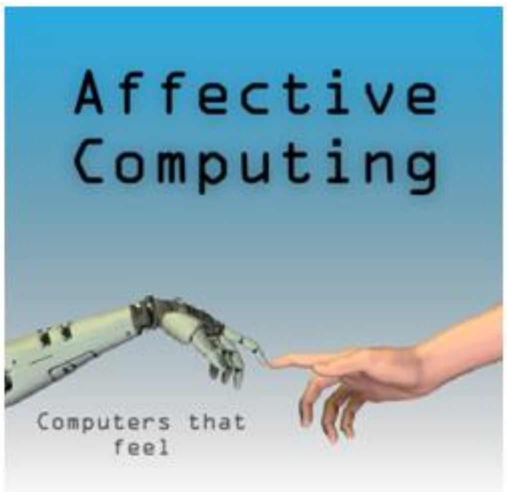


[Glove-talk: A neural network interface between a data-glove and a speech synthesizer](#) By Sidney Fels & *Geoffrey Hinton [CHI'95]*



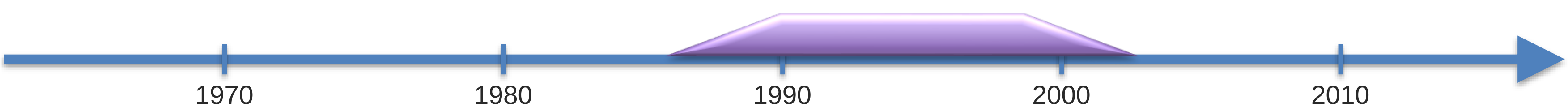
➢ The "Computational" Era (Late 1980s until 2000)

2) Multimodal/multisensory interfaces



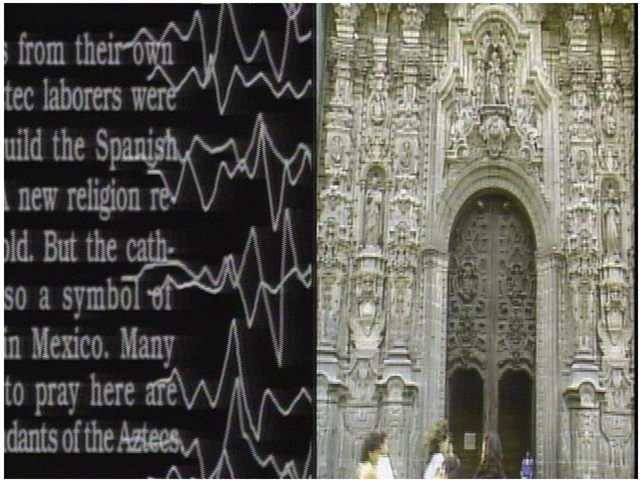
Rosalind Picard

Affective Computing *is computing that relates to, arises from, or deliberately influences emotion or other affective phenomena.*



➢ The "Computational" Era (Late 1980s until 2000)

3) Multimedia Computing

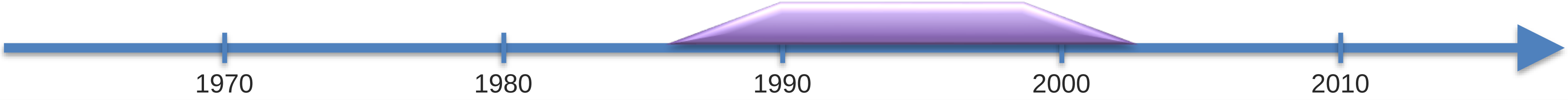


Carnegie Mellon University

informedia digital video understanding

[1994-2010]

"The Informedia Digital Video Library Project automatically combines speech, image and natural language understanding to create a full-content searchable digital video library."

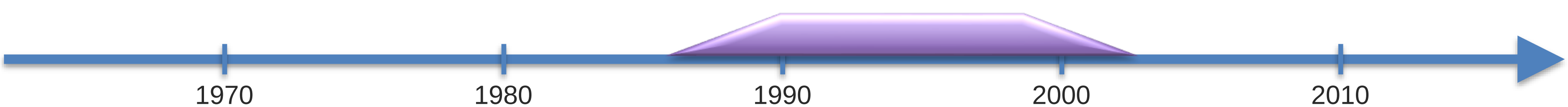


➢ The "Computational" Era (Late 1980s until 2000)

3) Multimedia Computing

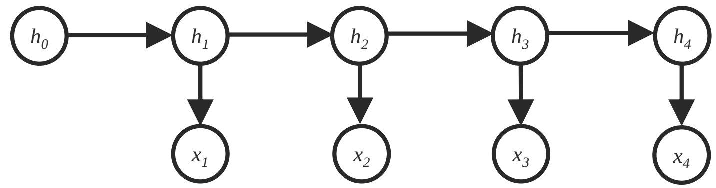
Multimedia content analysis

- **Shot-boundary detection (1991 -)**
  - Parsing a video into continuous camera shots
- **Still and dynamic video abstracts (1992 -)**
  - Making video browsable via representative frames (keyframes)
  - Generating short clips carrying the essence of the video content
- **High-level parsing (1997 -)**
  - Parsing a video into semantically meaningful segments
- **Automatic annotation (indexing) (1999 -)**
  - Detecting prespecified events/scenes/objects in video

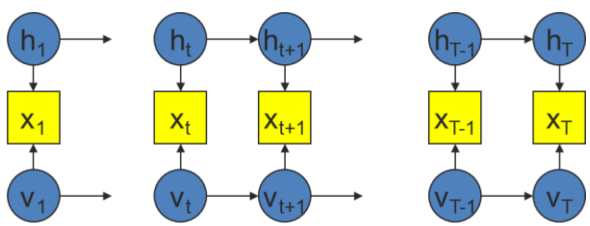


Multimodal Computation Models

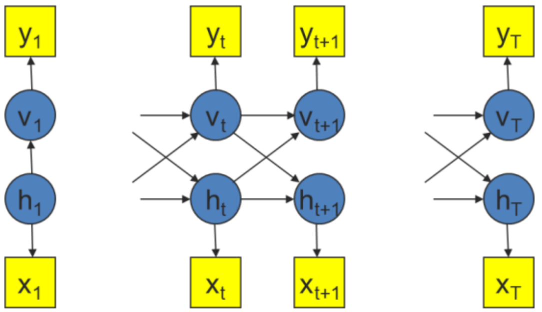
- Hidden Markov Models [1960s]

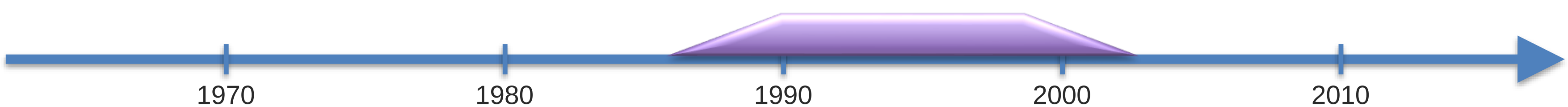


- ❑ Factorial Hidden Markov Models [1996]



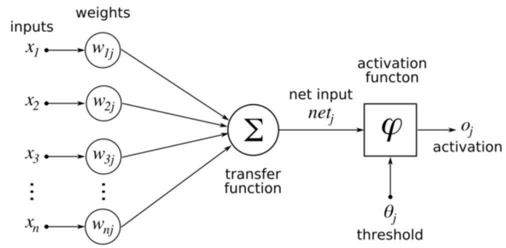
- ❑ Coupled Hidden Markov Models [1997]



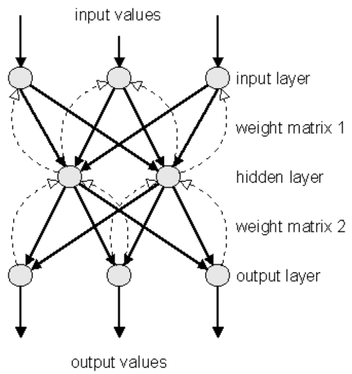


Multimodal Computation Models

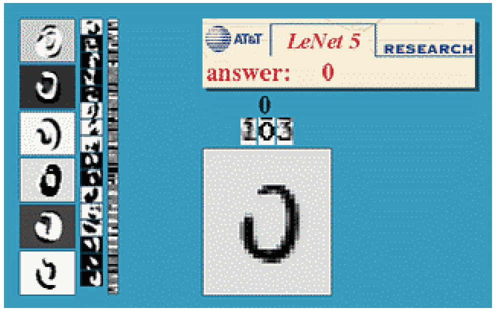
- Artificial Neural Networks [1940s]

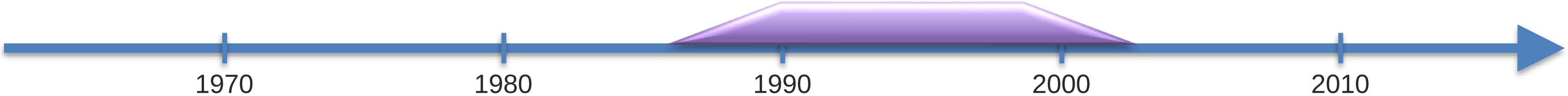


❑ Backpropagation [1975]



❑ Convolutional neural networks [1980s]





➢ The “Interaction” Era (2000s)

1) Modeling Human Multimodal Interaction

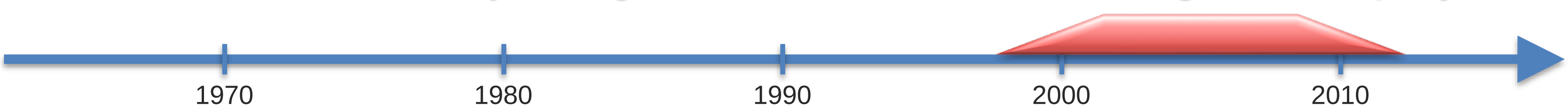
AMI Project [2001-2006, IDIAP]

- 100+ hours of meeting recordings
- Fully synchronized audio-video
- Transcribed and annotated

CHIL Project [Alex Waibel]

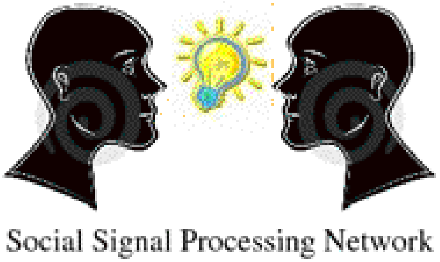
- Computers in the Human Interaction Loop
- Multi-sensor multimodal processing
- Face-to-face interactions

❑ TRIVIA: Samy Bengio started at IDIAP working on AMI project



➢ The "Interaction" Era (2000s)

1) Modeling Human Multimodal Interaction



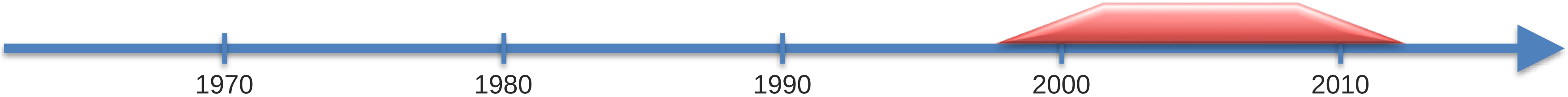
CALO Project [2003-2008, SRI]

- Cognitive Assistant that Learns and Organizes
- Personalized Assistant that Learns (PAL)
- Siri was a spinoff from this project

SSP Project [2008-2011, IDIAP]

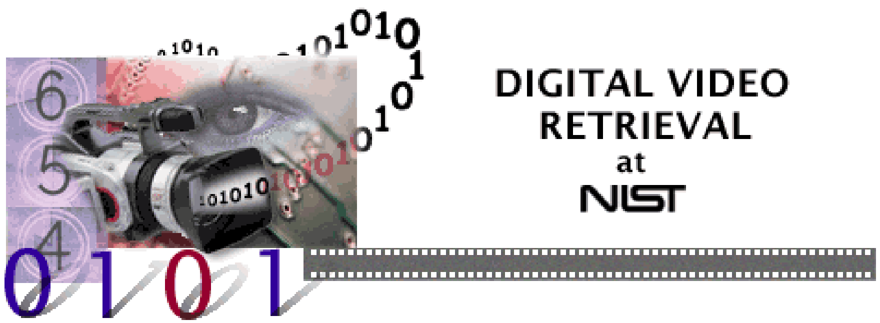
- Social Signal Processing
- First coined by Sandy Pentland in 2007
- Great dataset repository: http://sspnet.eu/

❑ TRIVIA: LP's PhD research was partially funded by CALO ☺



➢ The “Interaction” Era (2000s)

2) Multimedia Information Retrieval

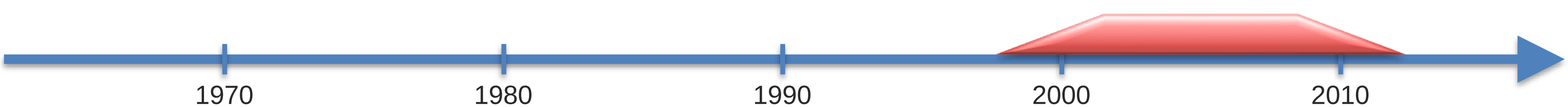


“Yearly competition to promote progress in content-based retrieval from digital video via open, metrics-*based evaluation”*

[Hosted by NIST, 2001-2016]

Research tasks and challenges:

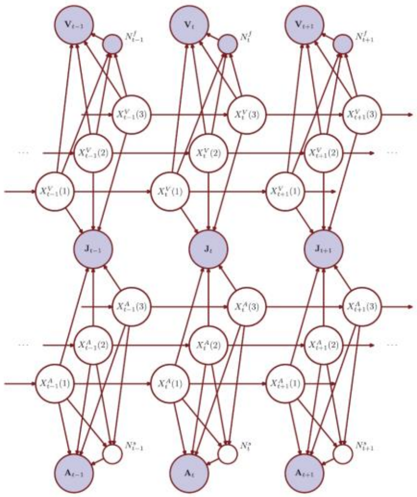
- Shot boundary, story segmentation, search
- “High-level feature extraction”: semantic event detection
- Introduced in 2008: copy detection and surveillance events
- Introduced in 2010: Multimedia event detection (MED)

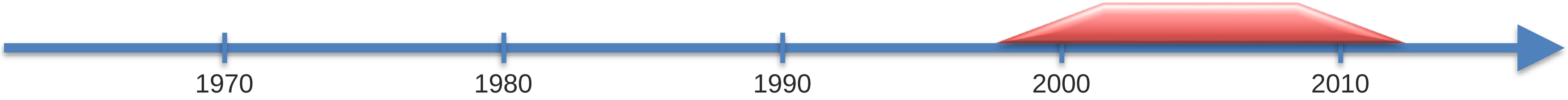


Multimodal Computational Models

- Dynamic Bayesian Networks
  - Kevin Murphy's PhD thesis and Matlab toolbox
  - Asynchronous HMM for multimodal [Samy Bengio, 2007]

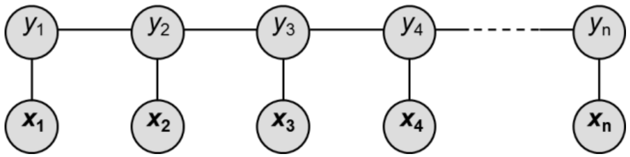
Audio-visual speech segmentation



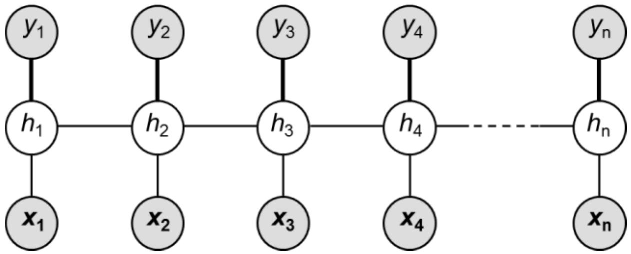


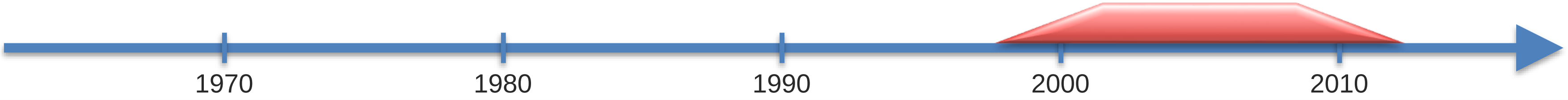
Multimodal Computational Models

- Discriminative sequential models
  - Conditional random fields [Lafferty et al., 2001]



  - Latent-dynamic CRF [Morency et al., 2007]





➢ The "deep learning" era (2010s until …)

Representation learning (a.k.a. deep learning)

- Multimodal deep learning [ICML 2011]
- Multimodal Learning with Deep Boltzmann Machines [NIPS 2012]
- Visual attention: Show, Attend and Tell: Neural Image Caption Generation with Visual Attention [ICML 2015]

Key enablers for multimodal research:

- New large-scale multimodal datasets
- Faster computer and GPUS
- High-level visual features
- "Dimensional" linguistic features

Our tutorial focuses on this era!

➢ The “deep learning” era (2010s until …)

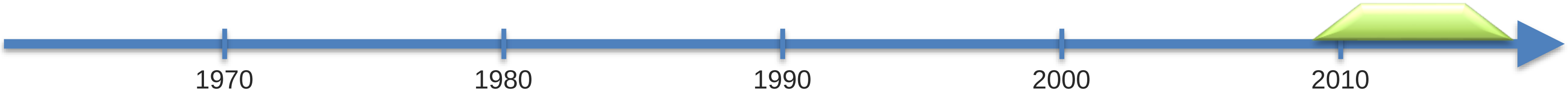
Many new challenges and multimodal corpora !!

Audio-Visual Emotion Challenge (AVEC, 2011-)

- Emotional dimension estimation
- Standardized training and test sets
- Based on the SEMAINE dataset

Emotion Recognition in the Wild Challenge (EmotiW 2013-)

- Emotional dimension estimation
- Standardized training and test sets
- Based on the SEMAINE dataset



➢ The "deep learning" era (2010s until …)

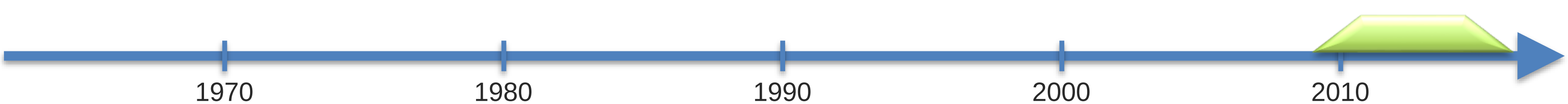
Renew of multimedia content analysis !

- Image captioning

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.

- Video description
- Visual Question-Answer



Real-World Tasks Tackled by Multimodal Research

- Affect recognition
  - Emotion
  - Persuasion
  - Personality traits
- Media description
  - Image captioning
  - Video captioning
  - Visual Question Answering
- Event recognition
  - Action recognition
  - Segmentation
- Multimedia information retrieval
  - Content based/Cross-media

"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."

(a) answer-phone (a) get-out-car (a) fight-person (b) push-up (b) cartwheel

Core Technical Challenges

Core Challenges in “Deep” Multimodal ML

Representation

Alignment

Fusion

Translation

Co-Learning

Multimodal Machine Learning: A Survey and Taxonomy

By Tadas Baltrusaitis, Chaitanya Ahuja, and Louis-Philippe Morency

https://arxiv.org/abs/1705.09406

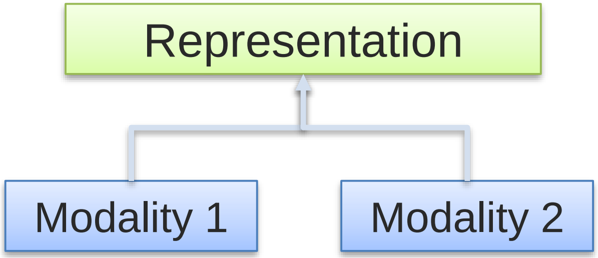
- ☑ **5 core challenges**
- ☑ **37 taxonomic classes**
- ☑ **253 referenced citations**

These challenges are non-exclusive.

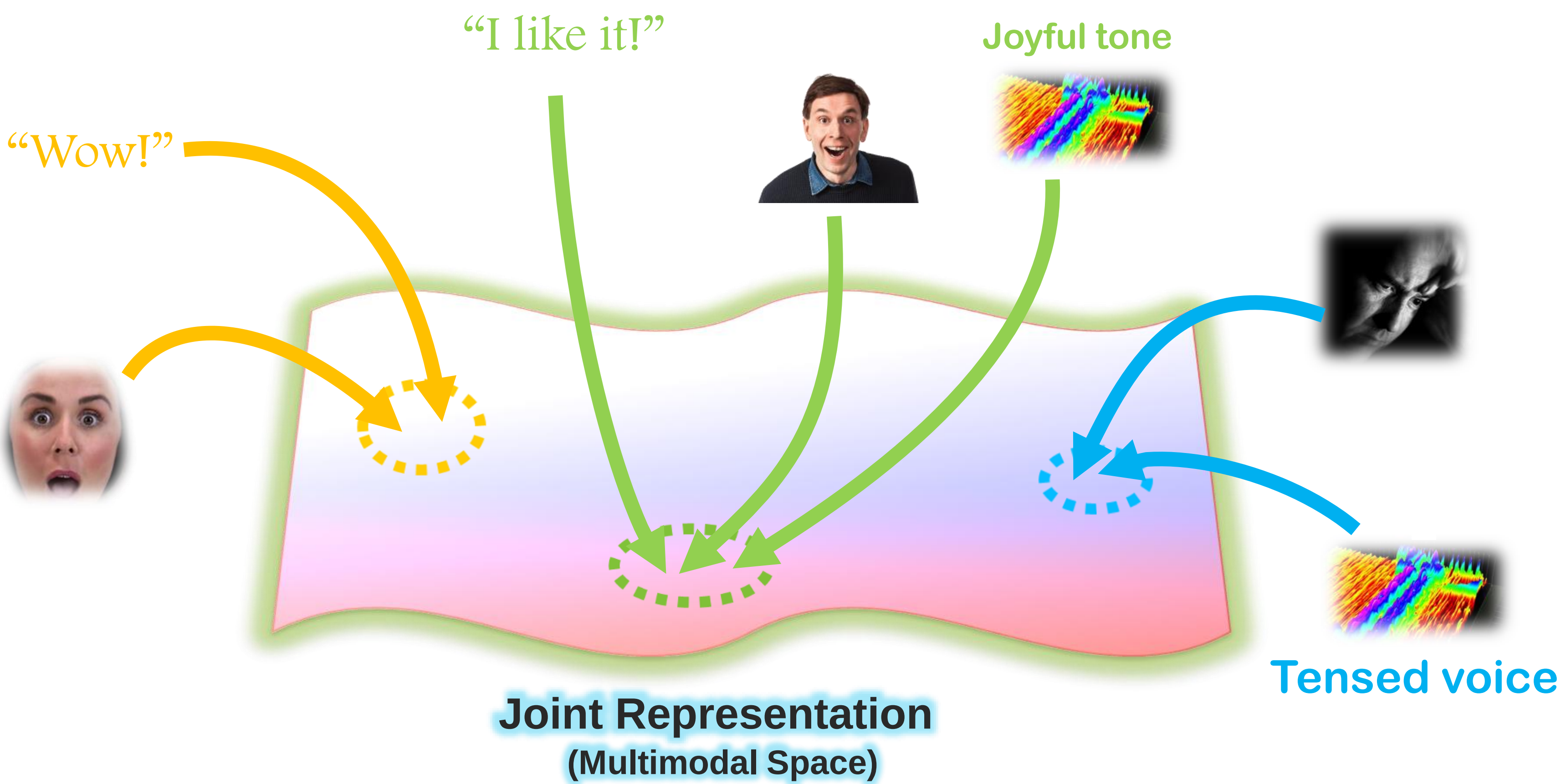
Core Challenge 1: Representation

Definition: Learning how to represent and summarize multimodal data in away that exploits the complementarity and redundancy.

Ⓐ Joint representations:



Joint Multimodal Representation



Joint Multimodal Representations

Audio-visual speech recognition
[Ngiam et al., ICML 2011]

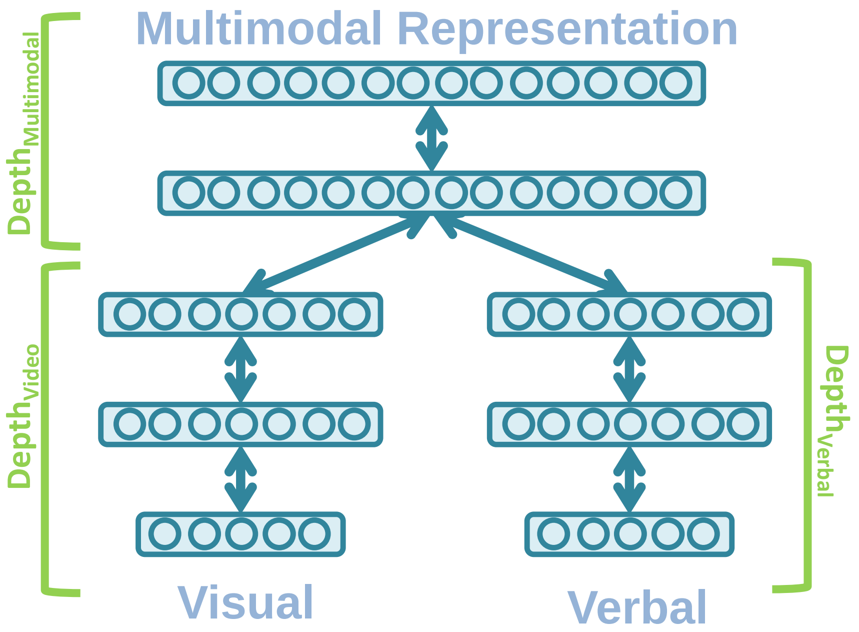
- Bimodal Deep Belief Network

Image captioning
[Srivastava and Salahutdinov, NIPS 2012]

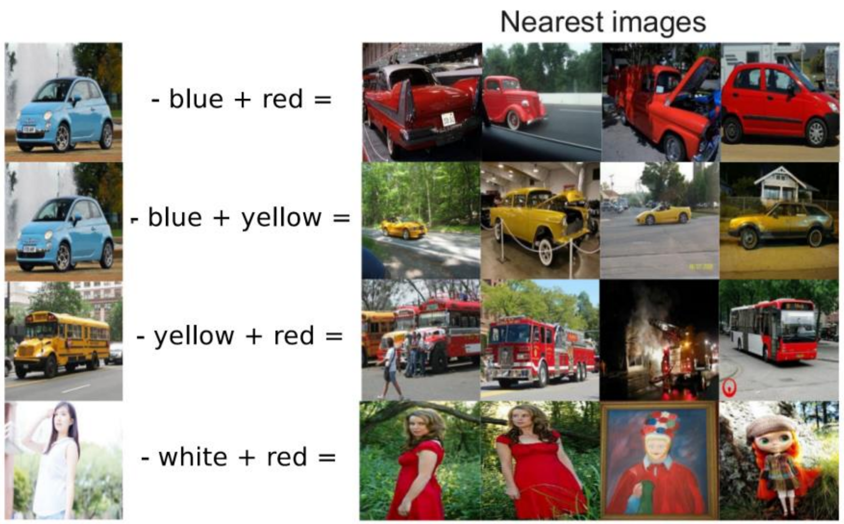
- Multimodal Deep Boltzmann Machine

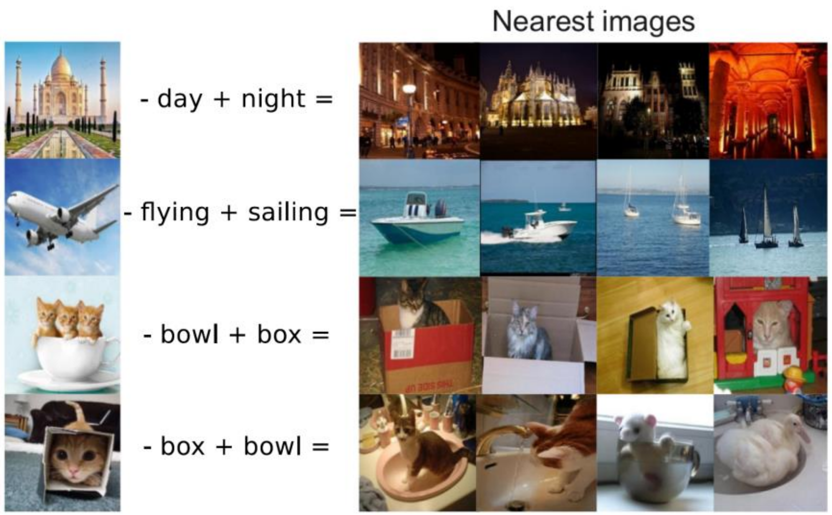
Audio-visual emotion recognition
[Kim et al., ICASSP 2013]

- Deep Boltzmann Machine



Multimodal Vector Space Arithmetic



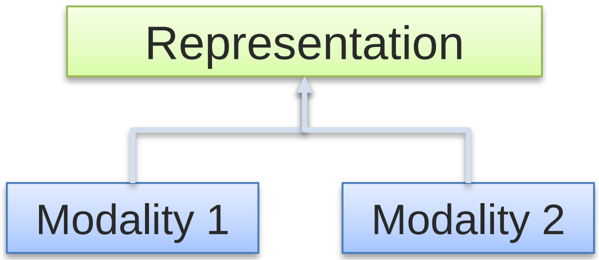


[Kiros et al., Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models, 2014]

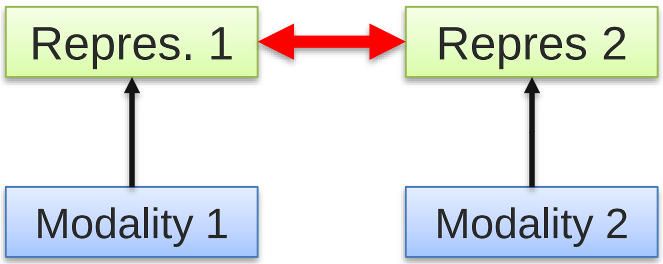
Core Challenge 1: Representation

Definition: Learning how to represent and summarize multimodal data in away that exploits the complementarity and redundancy.

A Joint representations:



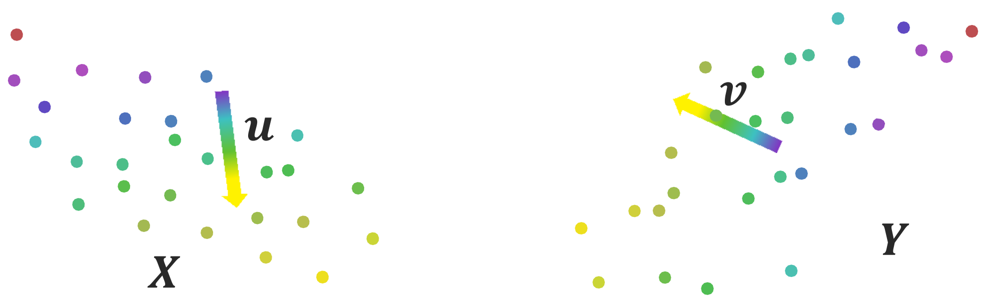
B Coordinated representations:

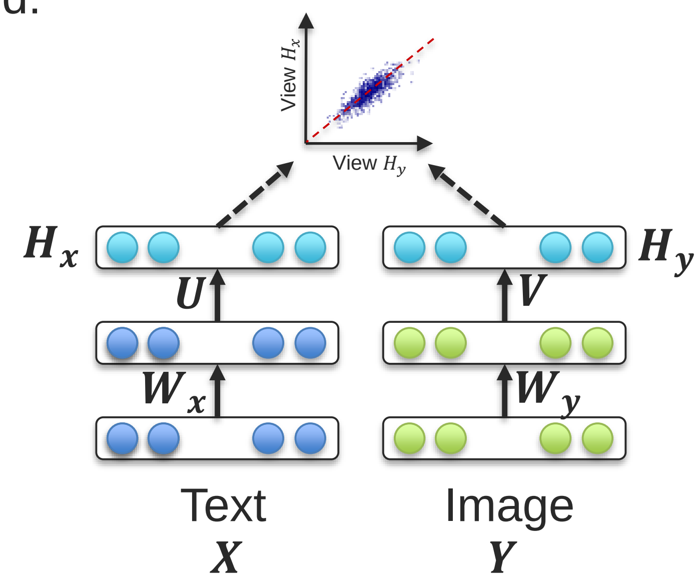


Coordinated Representation: Deep CCA

Learn linear projections that are maximally correlated:

$$(\boldsymbol{u}^*, \boldsymbol{v}^*) = \underset{\boldsymbol{u},\boldsymbol{v}}{\operatorname{argmax}}\, corr\left(\boldsymbol{u}^T\boldsymbol{X}, \boldsymbol{v}^T\boldsymbol{Y}\right)$$

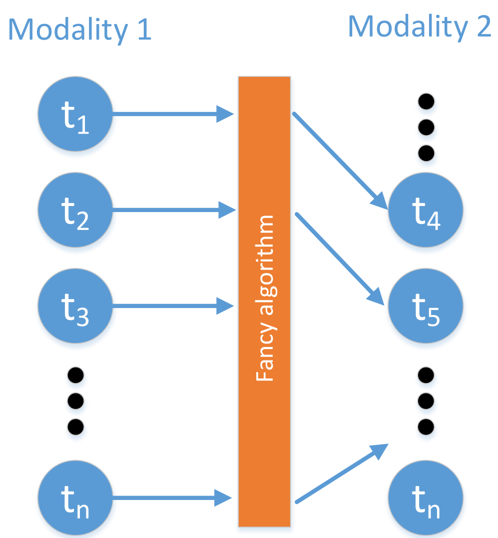




Andrew et al., ICML 2013

Core Challenge 2: Alignment

Definition: Identify the direct relations between (sub)elements from two or more different modalities.



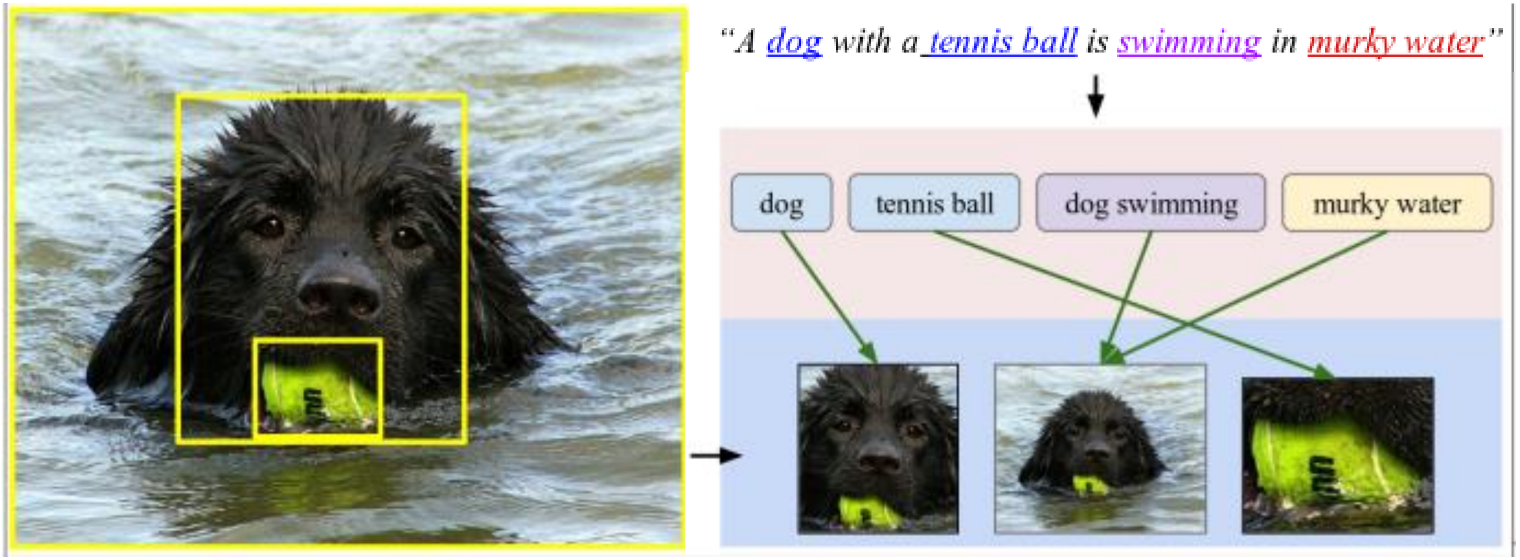
A Explicit Alignment

The goal is to directly find correspondences between elements of different modalities

B Implicit Alignment

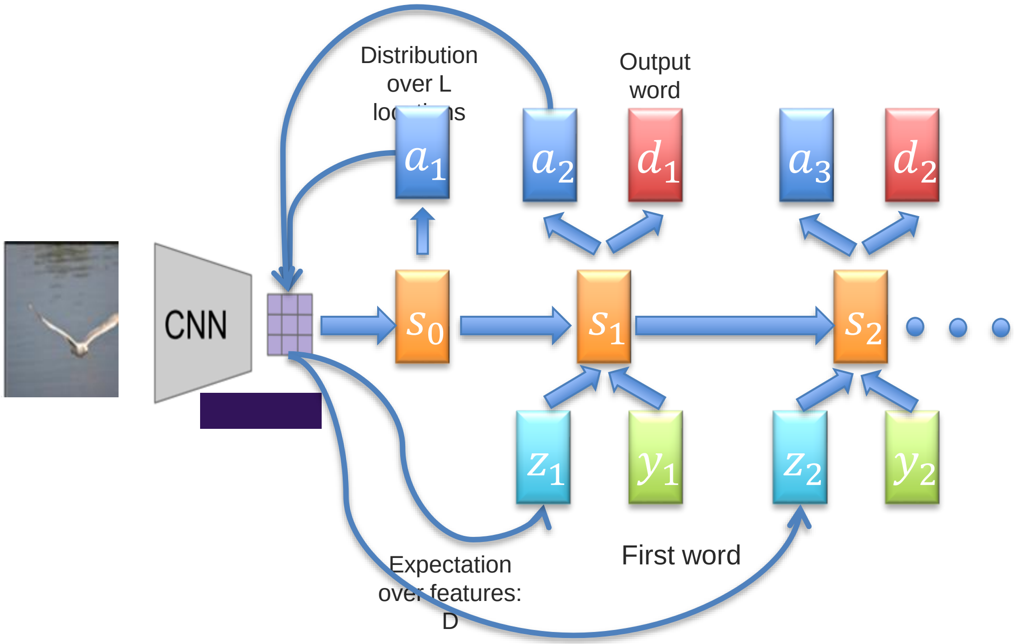
Uses internally latent alignment of modalities in order to better solve a different problem

Implicit Alignment



Karpathy et al., Deep Fragment Embeddings for Bidirectional Image Sentence Mapping, https://arxiv.org/pdf/1406.5679.pdf

Attention Models for Image Captioning



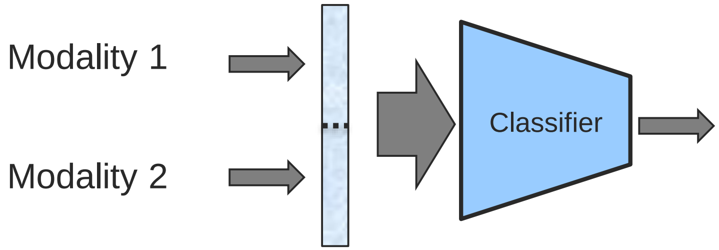
Xu et.al., ICML 2015

Core Challenge 3: Fusion

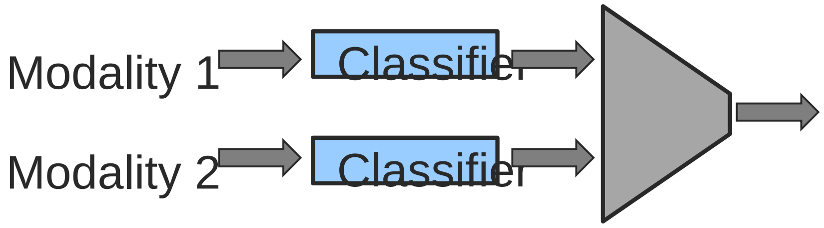
Definition: To join information from two or more modalities to perform a prediction task.

Ⓐ Model-Agnostic Approaches

1) Early Fusion



2) Late Fusion

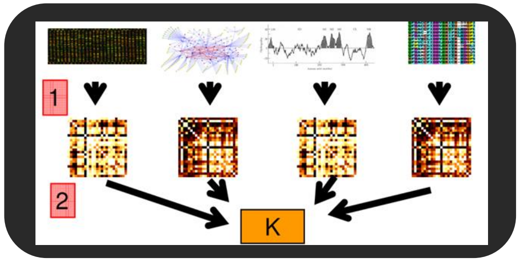


Core Challenge 3: Fusion

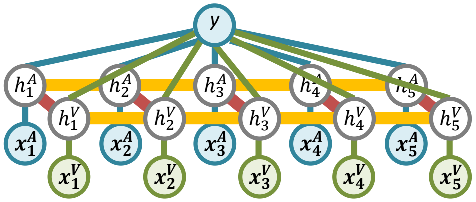
Definition: To join information from two or more modalities to perform a prediction task.

B Model-Based (Intermediate) Approaches

1) Deep neural networks
2) Kernel-based methods
3) Graphical models



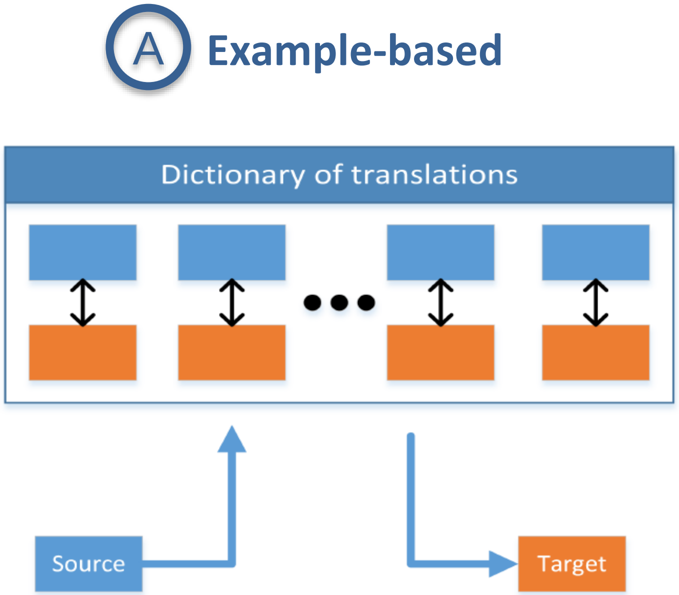
Multiple kernel learning

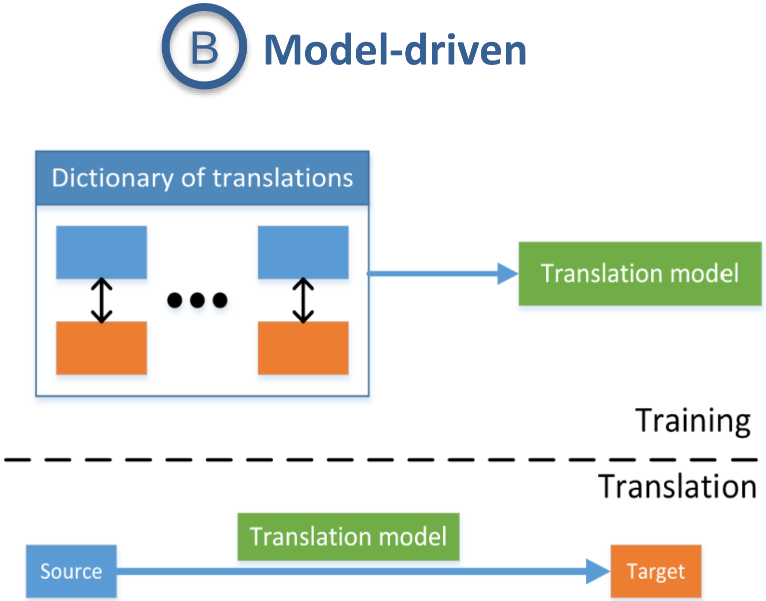


Multi-View Hidden CRF

Core Challenge 4: Translation

Definition: Process of changing data from one modality to another, where the translation relationship can often be open-ended or subjective.





Core Challenge 4 – Translation

Visual gestures
(both speaker and listener gestures)

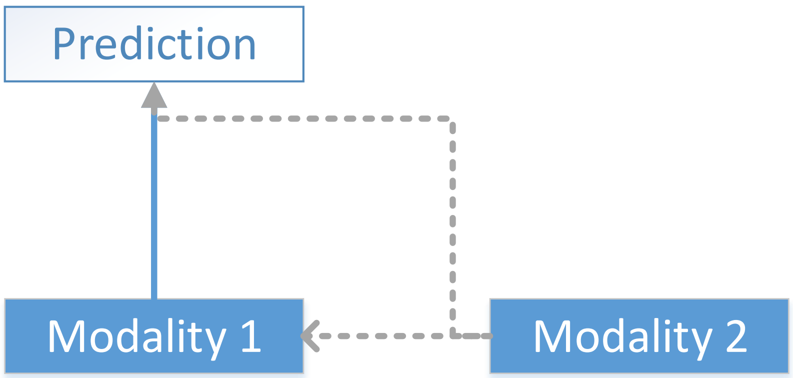
←

Transcriptions
+
Audio streams

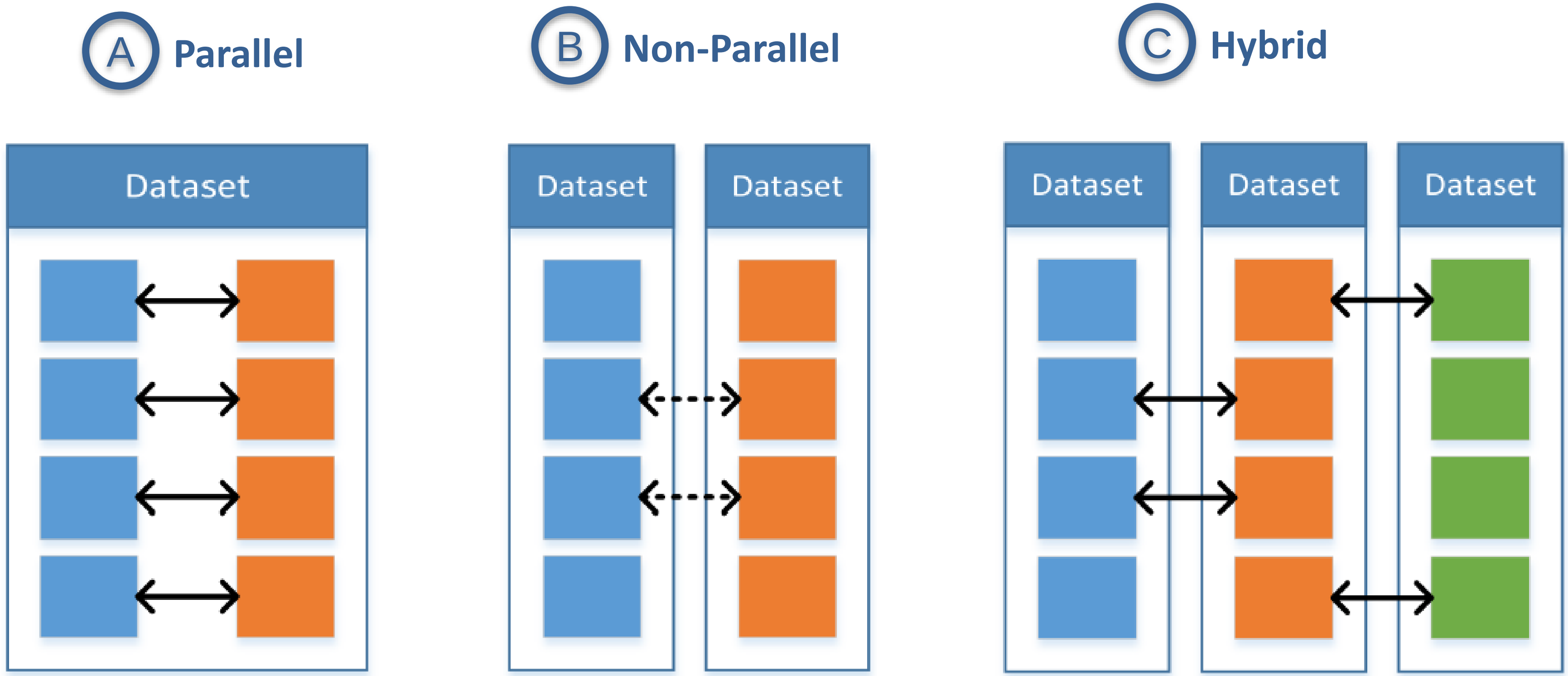
Marsella et al., Virtual character performance from speech, SIGGRAPH/Eurographics Symposium on Computer Animation, 2013

Core Challenge 5: Co-Learning

Definition: Transfer knowledge between modalities, including their representations and predictive models.



Core Challenge 5: Co-Learning



Taxonomy of Multimodal Research [https://arxiv.org/abs/1705.09406]

Representation

- Joint
  - Neural networks
  - Graphical models
  - Sequential
- Coordinated
  - Similarity
  - Structured

Translation

- Example-based
  - Retrieval
  - Combination
- Model-based
  - Grammar-based
  - Encoder-decoder
  - Online prediction

Alignment

- Explicit
  - Unsupervised
  - Supervised
- Implicit
  - Graphical models
  - Neural networks

Fusion

- Model agnostic
  - Early fusion
  - Late fusion
  - Hybrid fusion
- Model-based
  - Kernel-based
  - Graphical models
  - Neural networks

Co-learning

- Parallel data
  - Co-training
  - Transfer learning
- Non-parallel data
  - Zero-shot learning
  - Concept grounding
  - Transfer learning
- Hybrid data
  - Bridging

Tadas Baltrusaitis, Chaitanya Ahuja, and Louis-Philippe Morency, Multimodal Machine Learning: A Survey and Taxonomy

Multimodal Applications

[https://arxiv.org/abs/1705.09406]

	Challenges				
Applications	Representation	Translation	Fusion	Alignment	Co-learning
Speech Recognition and Synthesis					
Audio-visual Speech Recognition	✓		✓	✓	✓
(Visual) Speech Synthesis	✓	✓			
Event Detection					
Action Classification	✓		✓		✓
Multimedia Event Detection	✓		✓		✓
Emotion and Affect					
Recognition	✓		✓	✓	✓
Synthesis	✓	✓			
Media Description					
Image Description	✓	✓		✓	✓
Video Description	✓	✓	✓	✓	✓
Visual Question-Answering	✓		✓	✓	✓
Media Summarization	✓	✓	✓		
Multimedia Retrieval					
Cross Modal retrieval	✓	✓		✓	✓
Cross Modal hashing	✓				✓

Tadas Baltrusaitis, Chaitanya Ahuja, and Louis-Philippe Morency, Multimodal Machine Learning: A Survey and Taxonomy

Basic Concepts: Score and Loss Functions

Linear Classification (e.g., neural network)

Image

(Size: 32*32*3)

?

1. **Define a (linear) score function**
2. **Define the loss function (possibly nonlinear)**
3. **Optimization**

1) Score Function

Image

(Size: 32*32*3)

Duck ?
Cat ?
Dog ?
Pig ?
Bird ?

What should be the prediction score for each label class?

<u>For linear classifier:</u>

Input observation (ith element of the dataset)
[3072x1]

$$f(x_i; W, b) = W x_i + b$$

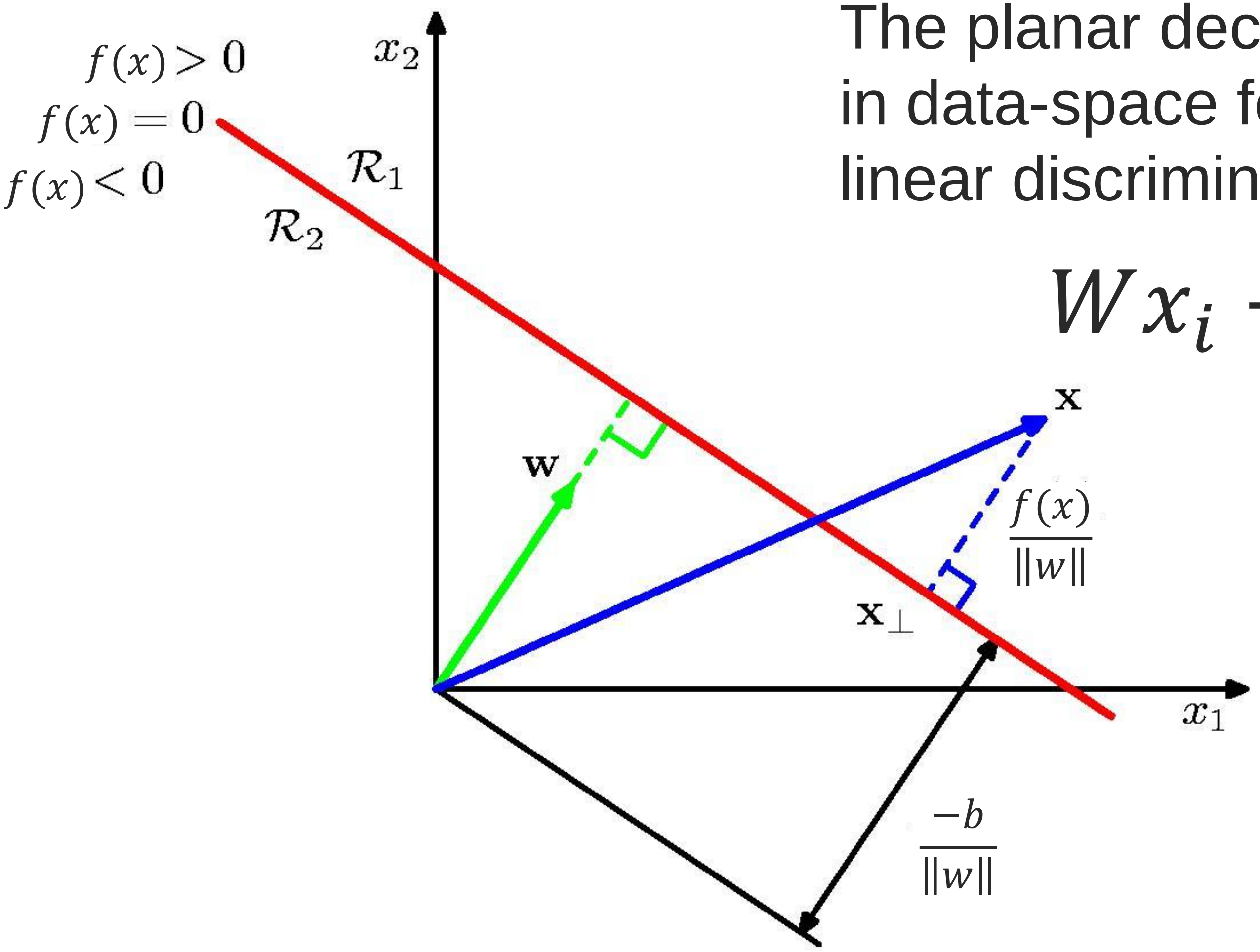
Class score
[10x1]

Weights [10x3072]

Bias vector [10x1]

Parameters [10x3073]

Interpreting a Linear Classifier



The planar decision surface in data-space for the simple linear discriminant function:

$$W x_i + b > 0$$

Some Notation Tricks – Multi-Label Classification

$$W = [W_1 \quad W_2 \quad \dots \quad W_N]$$

$$f(x_i; W, b) = Wx_i + b \longrightarrow f(x_i; W) = Wx_i$$

Weights x Input + Bias

[10x3072] [3072x1] [10x1]

Weights x Input

[10x3073] [3073x1]

The bias vector will be the last column of the weight matrix

Add a "1" at the end of the input observation vector

Some Notation Tricks

General formulation of linear classifier: $f(x_i; W, b)$

"dog" linear classifier:

$$f\left(x_i; W_{dog}, b_{dog}\right) \quad \text{or}$$

$$f(x_i; W, b)_{dog} \quad \text{or} \quad f_{dog}$$

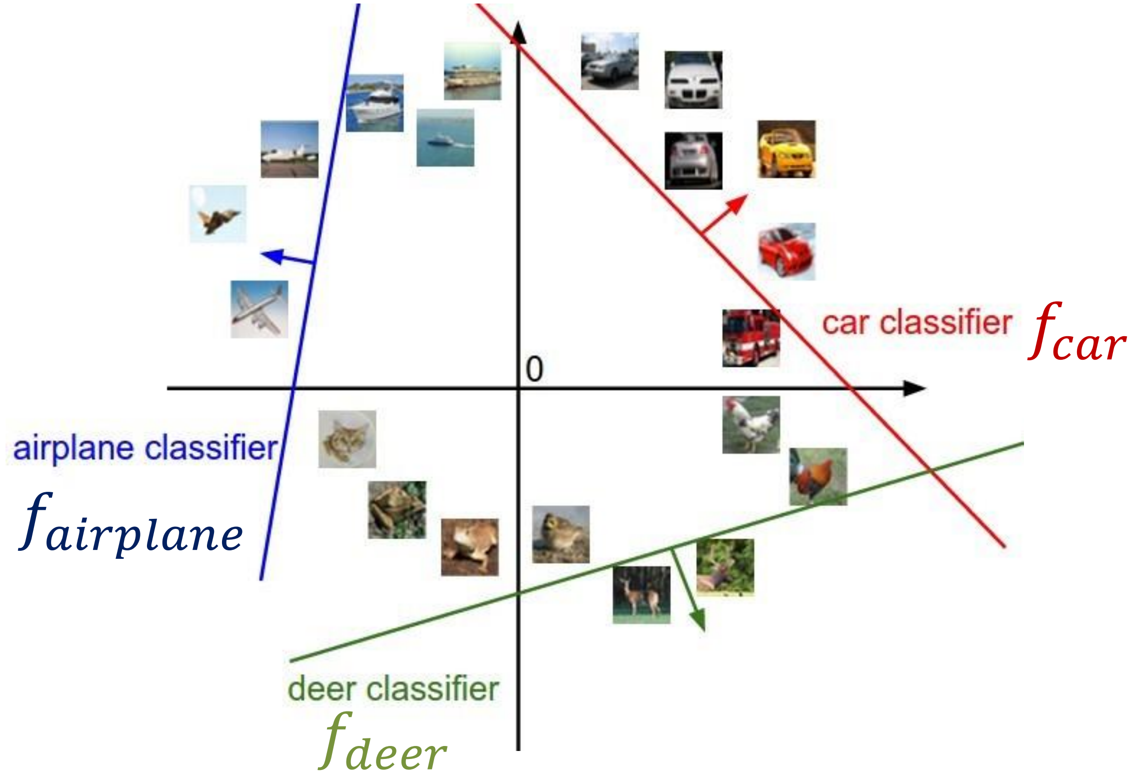
Linear classifier for label j:

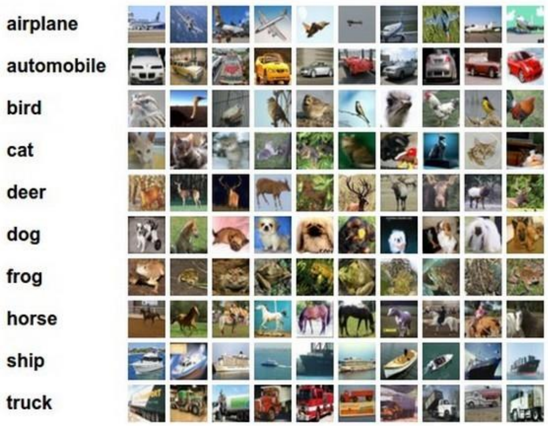
$$f\left(x_i; W_j, b_j\right) \quad \text{or}$$

$$f(x_i; W, b)_j \quad \text{or} \quad f_j$$

Interpreting Multiple Linear Classifiers

$$f(x_i; W_j, b_j) = W_j x_i + b_j$$





CIFAR-10 object recognition dataset

Linear Classification: 2) Loss Function

(or cost function or objective)

Scores $f(x_i; W)$ | **Label** $y_i = 2\ (dog)$ → **Loss** $L_i = ?$

Image x_i

(Size: 32*32*3)

0 (duck) ?	-12.3
1 (cat) ?	45.6
2 (dog) ?	98.7
3 (pig) ?	12.2
4 (bird) ?	-45.3

How to assign only one number representing how "unhappy" we are about these scores?

Multi-class problem

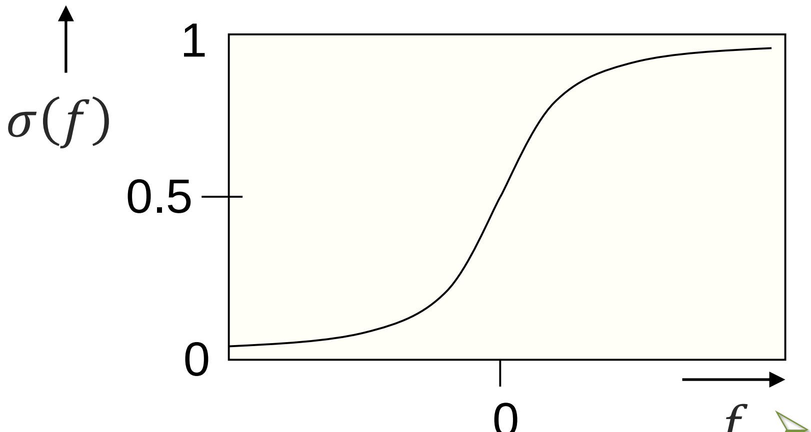
The loss function quantifies the amount by which the prediction scores deviate from the actual values.

A first challenge: how to normalize the scores?

First Loss Function: Cross-Entropy Loss

(or logistic loss)

Logistic function: $$\sigma(f) = \frac{1}{1 + e^{-f}}$$



➢ **Score function**

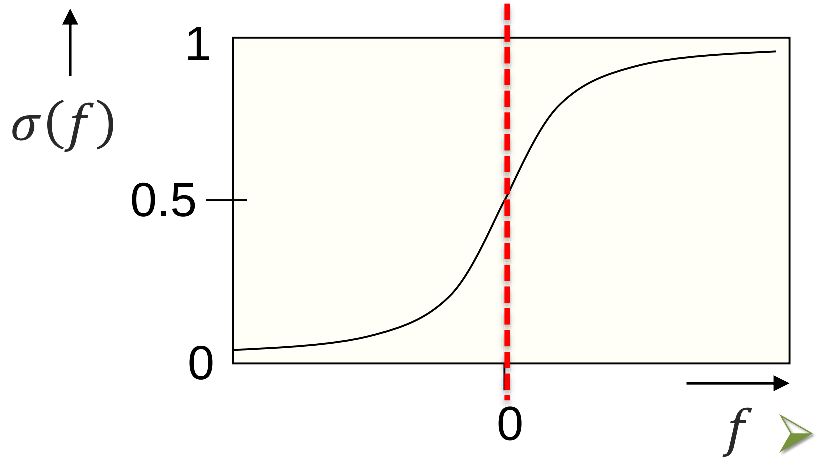
First Loss Function: Cross-Entropy Loss

(or logistic loss)

Logistic function: $$\sigma(f) = \frac{1}{1 + e^{-f}}$$

Logistic regression: $$p(y_i = "dog" | x_i; w) = \sigma(w^T x_i)$$
(two classes)

= true
for two-class problem



➢ Score function

First Loss Function: Cross-Entropy Loss

(or logistic loss)

Logistic function: $$\sigma(f) = \frac{1}{1+e^{-f}}$$

Logistic regression:
(two classes)

$$p(y_i = \text{"}dog\text{"} | x_i; w) \;\; = \sigma(w^T x_i)$$

= true
for two-class problem

Softmax function:
(multiple classes)

$$p(y_i | x_i; W) = \frac{e^{f_{y_i}}}{\sum_j e^{f_j}}$$

First Loss Function: Cross-Entropy Loss

(or logistic loss)

Cross-entropy loss:

$$L_i = \boxed{-\log\left(\boxed{\frac{e^{f_{y_i}}}{\sum_j e^{f_j}}}\right)}$$

Softmax function

Minimizing the negative log likelihood.

Second Loss Function: Hinge Loss

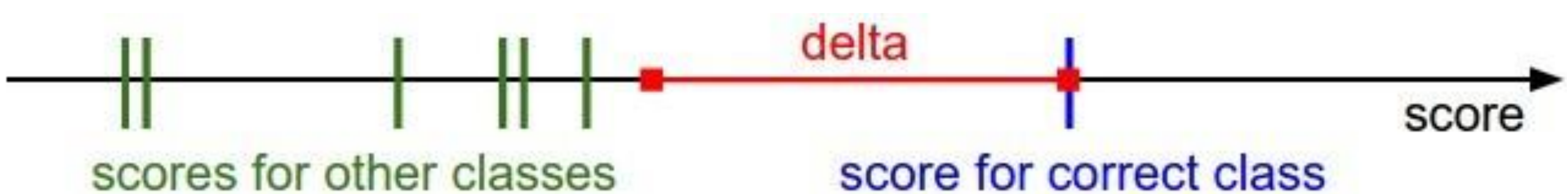
(or max-margin loss or Multi-class SVM loss)

$$\boxed{L_i} = \boxed{\sum_{j\neq y_i}} \max(0, \boxed{f(x_i,W)_j - f(x_i,W)_{y_i}} + \Delta)$$

loss due to example i

sum over all incorrect labels

difference between the correct class score and incorrect class score



Second Loss Function: Hinge Loss

(or max-margin loss or Multi-class SVM loss)

$$L_i = \sum_{j \neq y_i} \max(0, f(x_i, W)_j - f(x_i, W)_{y_i} + \Delta)$$

e.g. 10

Example: $f(x_i, W) = [13, -7, 11]$

$y_i = 0$

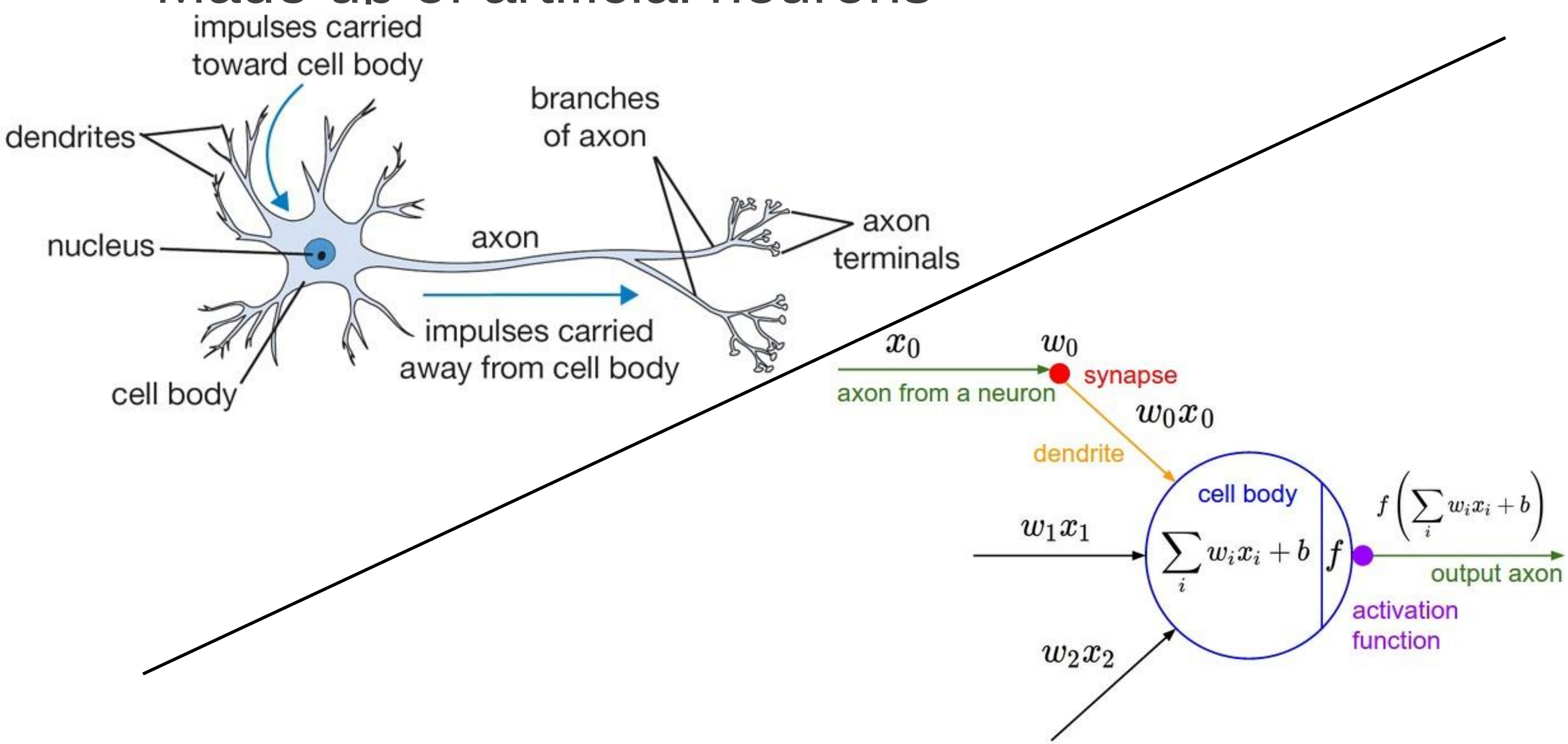
$$L_i = \max(0, -7 - 13 + 10) + \max(0, 11 - 13 + 10)$$

How to find the optimal W?

Basic Concepts: Neural Networks

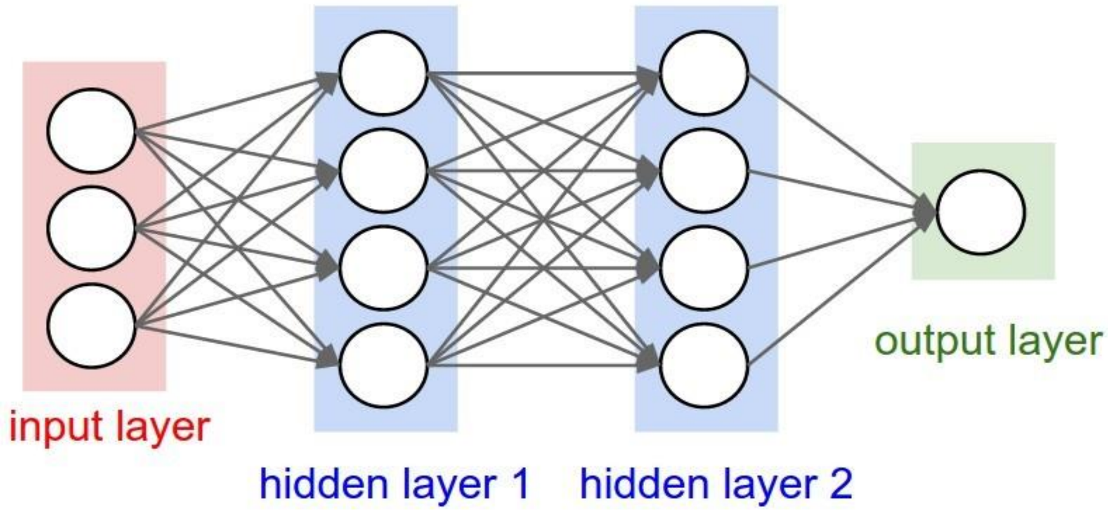
Neural Networks – inspiration

- Made up of artificial neurons



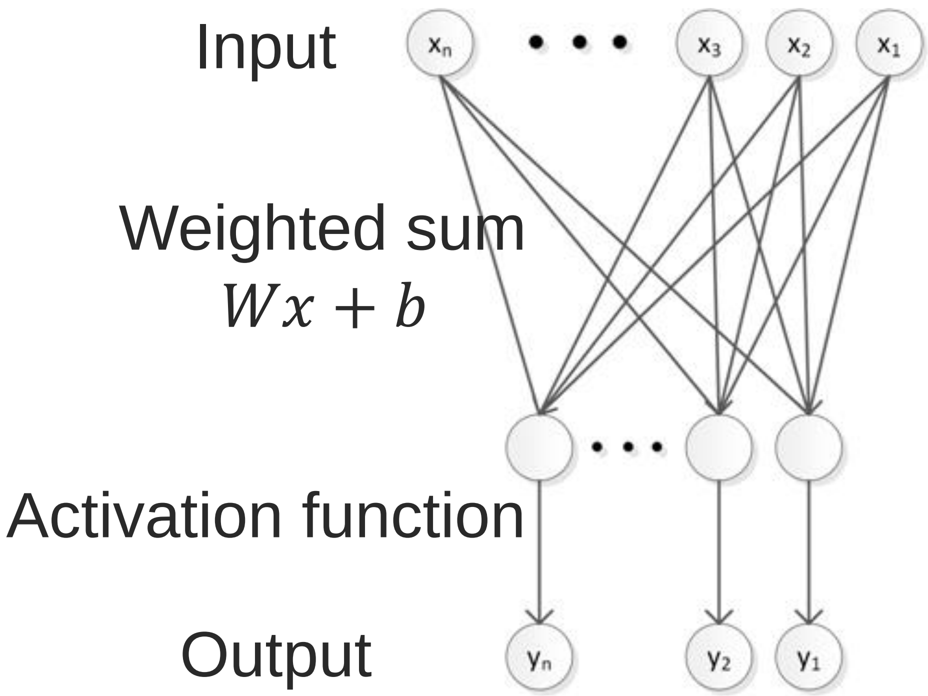
Neural Networks – score function

- Made up of artificial neurons
  - Linear function (dot product) followed by a nonlinear activation function
- Example a Multi Layer Perceptron



Basic NN building block

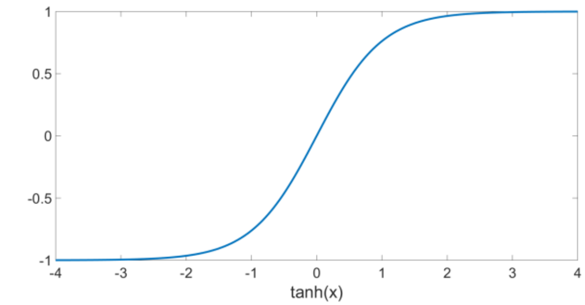
- Weighted sum followed by an activation function



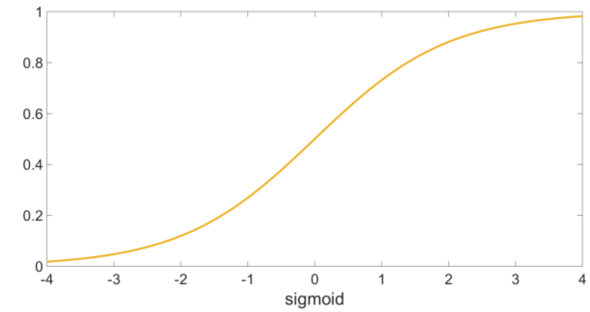
$$y = f(Wx + b)$$

Neural Networks – activation function

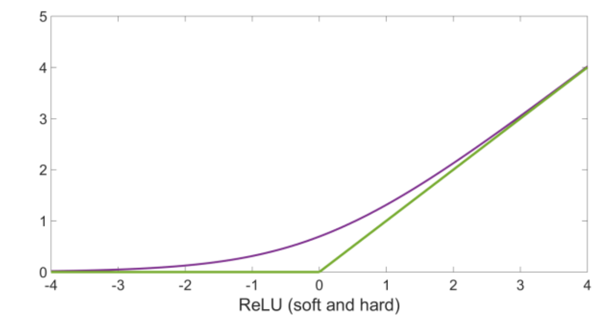
- $f(x) = \tanh(x)$



- Sigmoid - $f(x) = (1 + e^{-x})^{-1}$



- Linear – $f(x) = ax + b$
- ReLU $f(x) = \max(0, x) \sim \log(1 + \exp(x))$
  - Rectifier Linear Units
  - Faster training - no gradient vanishing
  - Induces sparsity



Neural Networks – loss function

- Already discussed it – cross-entropy, Euclidean loss, cosine similarity, etc.
- Combine it with the score function to have an end-to-end training objective
- As example use Euclidean loss for data-point I

$$L_i = (f(x_i) - y_i)^2 = (f_{3;W_3}(f_{2;W_2}(f_{1;W_1}(x_i))))^2$$

- Full loss is computed across all training samples

$$L = \sum_i (f(x_i) - y_i)^2$$

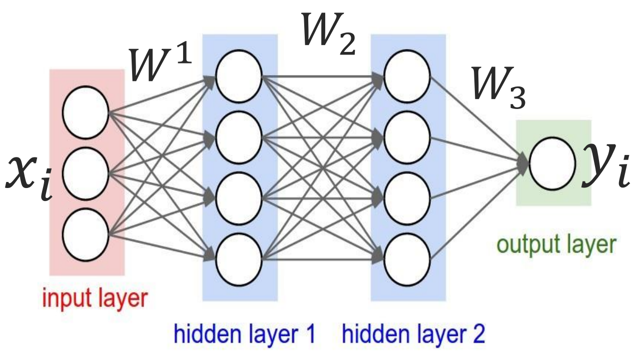
Multi-Layer Feedforward Network

Activation functions (individual layers)

$$f_{1;W_1}(x) = \sigma(W_1 x + b_1)$$

$$f_{2;W_2}(x) = \sigma(W_2 x + b_2)$$

$$f_{3;W_3}(x) = \sigma(W_3 x + b_3)$$



Score function

$$y_i = f(x_i) = f_{3;W_3}(f_{2;W_2}(f_{1;W_1}(x_i)))$$

Loss function (e.g., Euclidean loss)

$$L_i = (f(x_i) - y_i)^2 = (f_{3;W_3}(f_{2;W_2}(f_{1;W_1}(x_i))))^2$$

Basic Concepts: Optimization

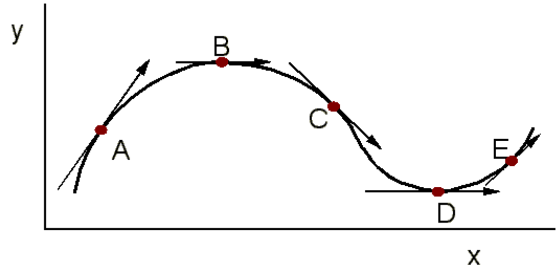
Optimizing a generic function

- We want to find a minimum (or maximum) of a generic function
- How do we do that?
  - Searching everywhere (global optimum) is computationally infeasible
  - We could search randomly from our starting point (mostly picked at random) – impractical and not accurate
  - Instead we can follow the gradient

What is a gradient?

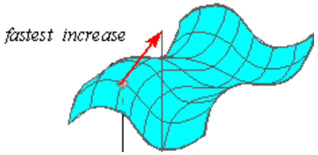
- Geometrically
  - Points in the direction of the greatest rate of increase of the function and its magnitude is the slope of the graph in that direction
- More formally in 1D

$$\frac{df(x)}{dx} = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$



- In higher dimensions

$$\frac{\partial f}{\partial x_i}(a_1, \dots, a_n) = \lim_{h \to 0} \frac{f(a_1, \dots, a_i + h, \dots, a_n) - f(a_1, \dots, a_i, \dots, a_n)}{h}$$

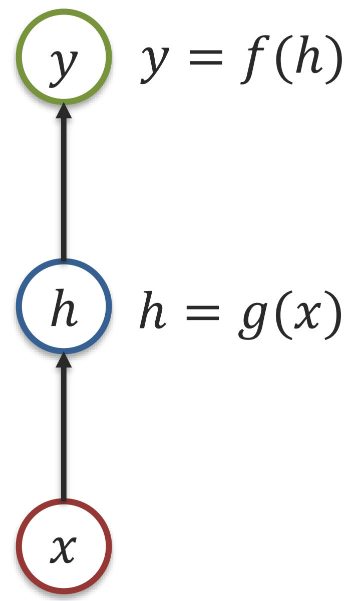


➢ In multiple dimension, the **gradient** is the vector of (partial derivatives) and is called a **Jacobian**.

Gradient Computation

Chain rule:

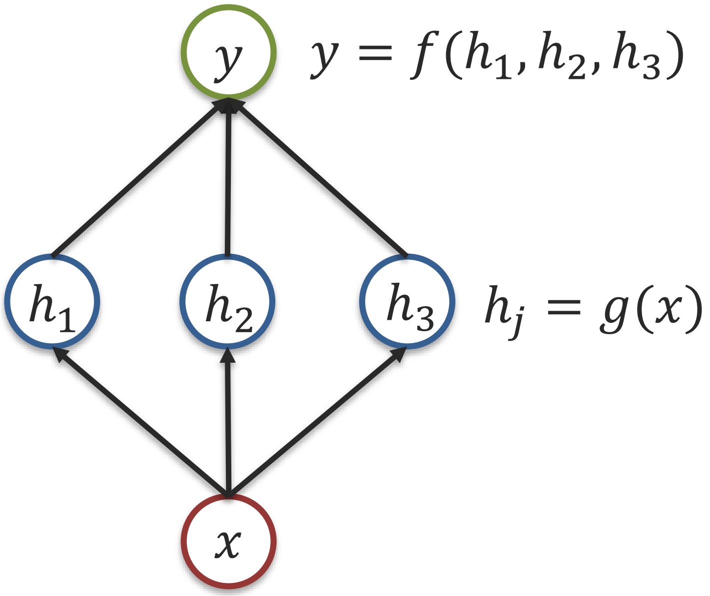
$$\frac{\partial y}{\partial x} = \frac{\partial y}{\partial h}\frac{\partial h}{\partial x}$$



Optimization: Gradient Computation

Multiple-path chain rule:

$$\frac{\partial y}{\partial x} = \sum_j \frac{\partial y}{\partial h_j} \frac{\partial h_j}{\partial x}$$



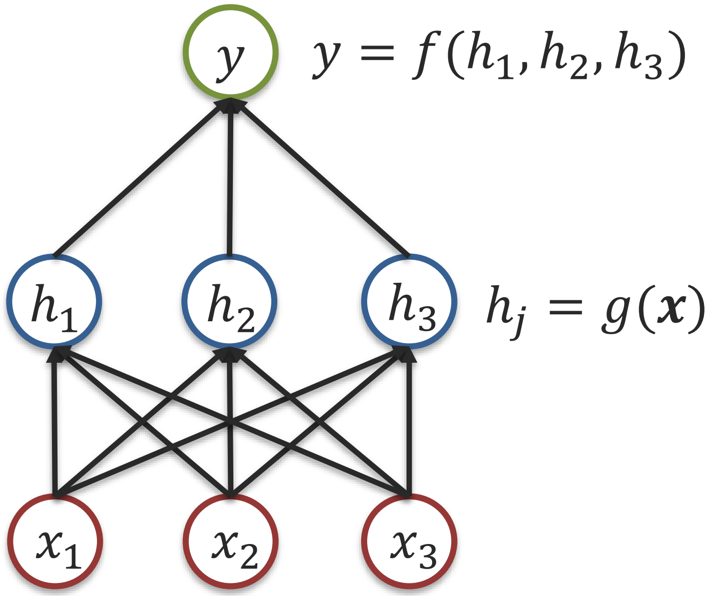
Optimization: Gradient Computation

Multiple-path chain rule:

$$\frac{\partial y}{\partial x_1} = \sum_j \frac{\partial y}{\partial h_j} \frac{\partial h_j}{\partial x_1}$$

$$\frac{\partial y}{\partial x_2} = \sum_j \frac{\partial y}{\partial h_j} \frac{\partial h_j}{\partial x_1}$$

$$\frac{\partial y}{\partial x_3} = \sum_j \frac{\partial y}{\partial h_j} \frac{\partial h_j}{\partial x_1}$$



Optimization: Gradient Computation

Vector representation:

$$\nabla_{\boldsymbol{x}}\, y = \left[\frac{\partial y}{\partial x_1}, \frac{\partial y}{\partial x_2}, \frac{\partial y}{\partial x_3}\right]$$

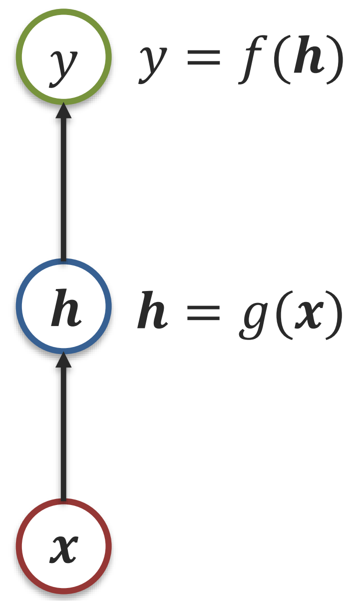
Gradient

$$\nabla_{\boldsymbol{x}}\, y = \left(\boxed{\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}}\right)^T \nabla_{\boldsymbol{h}}\, y$$

"backprop" Gradient

"local" Jacobian

(matrix of size $|h| \times |x|$ computed using partial derivatives)



Backpropagation Algorithm (efficient gradient)

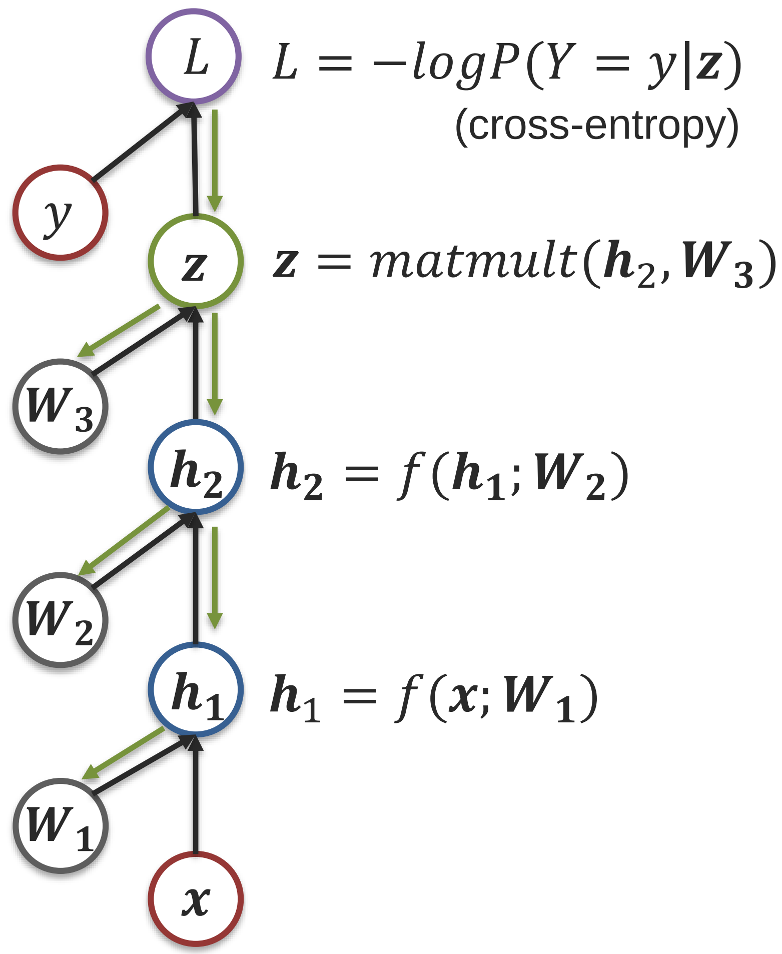
Forward pass

- Following the graph topology, compute value of each unit

Backpropagation pass

- Initialize output gradient = 1
- Compute "local" Jacobian matrix using values from forward pass
- Use the chain rule:

Gradient = "local" Jacobian x "backprop" gradient



How to follow the gradient

- Many methods for optimization
  - **Gradient Descent (actually the "simplest" one)**
  - Newton methods (use Hessian – second derivative)
  - Quasi-Newton (use approximate Hessian)
    - BFGS
    - LBFGS
    - Don't require learning rates (fewer hyperparameters)
    - But, do not work with stochastic and batch methods so rarely used to train modern Neural Networks
- **All of them look at the gradient**
  - Very few non gradient based optimization methods

Parameter Update Strategies

Gradient descent:

$$\theta^{(t+1)} = \theta^{t} - \epsilon_k \boxed{\nabla_\theta L}$$

Gradient of our loss function

New model parameters

Previous parameters

Learning rate at iteration k

$$\epsilon_k = (1-\alpha)\epsilon_0 + \alpha \boxed{\epsilon_\tau}$$

Decay learning rate linearly until iteration τ

Learning rate at iteration k

Decay

Initial learning rate

Extensions:

- Stochastic (“batch”)
- with momentum
- AdaGrad
- RMSProp

Unimodal representations: Language Modality

Unimodal Classification – Language Modality

Written language

★★★★★ **Masterful!**

By Antony Witheyman - January 12, 2006

Ideal for anyone with an interest in disguises who likes to see the subject tackled in a $\boxed{\text{humourous}}$ manner.

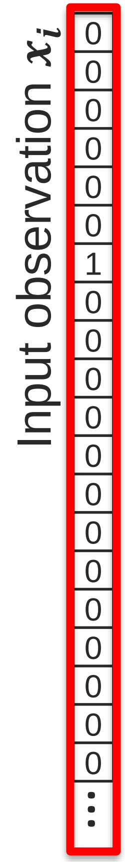
0 of 4 people found this review helpful

Spoken language

```
                MARTHA(CON'T)
Look around you. Look at all the
great things you've done and the
people you've helped.

                CLARK
But you've only put up the good
things they say about me.

                MARTHA
Clark, honey. If I were to use the
bad things they say I could cover
the barn, the house and the
outhouse.
```



"one-hot" vector

$|x_i|$ = number of words in dictionary

Word-level classification

Part-of-speech ?
(noun, verb,…)

Sentiment ?
(positive or negative)

Named entity ?
(names of person,…)

Unimodal Classification – Language Modality

Written language

Masterful!

By Antony Witheyman - January 12, 2006

Ideal for anyone with an interest in disguises who likes to see the subject tackled in a humourous manner.

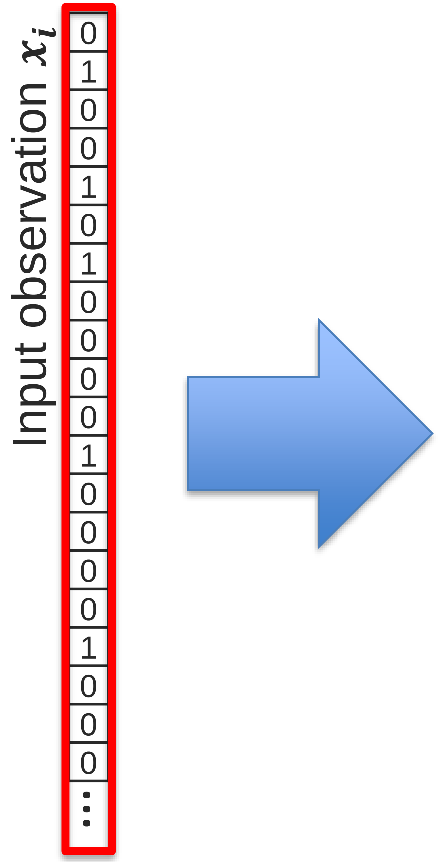
0 of 4 people found this review helpful

Spoken language

```
                MARTHA(CON'T)
Look around you. Look at all the
great things you've done and the
people you've helped.

                CLARK
But you've only put up the good
things they say about me.

                MARTHA
Clark, honey. If I were to use the
bad things they say I could cover
the barn, the house and the
outhouse.
```



Document-level classification

Sentiment ?
(positive or negative)

"bag-of-word" vector

$|x_i|$ = number of words in dictionary

How to Learn (Better) Language Representations?

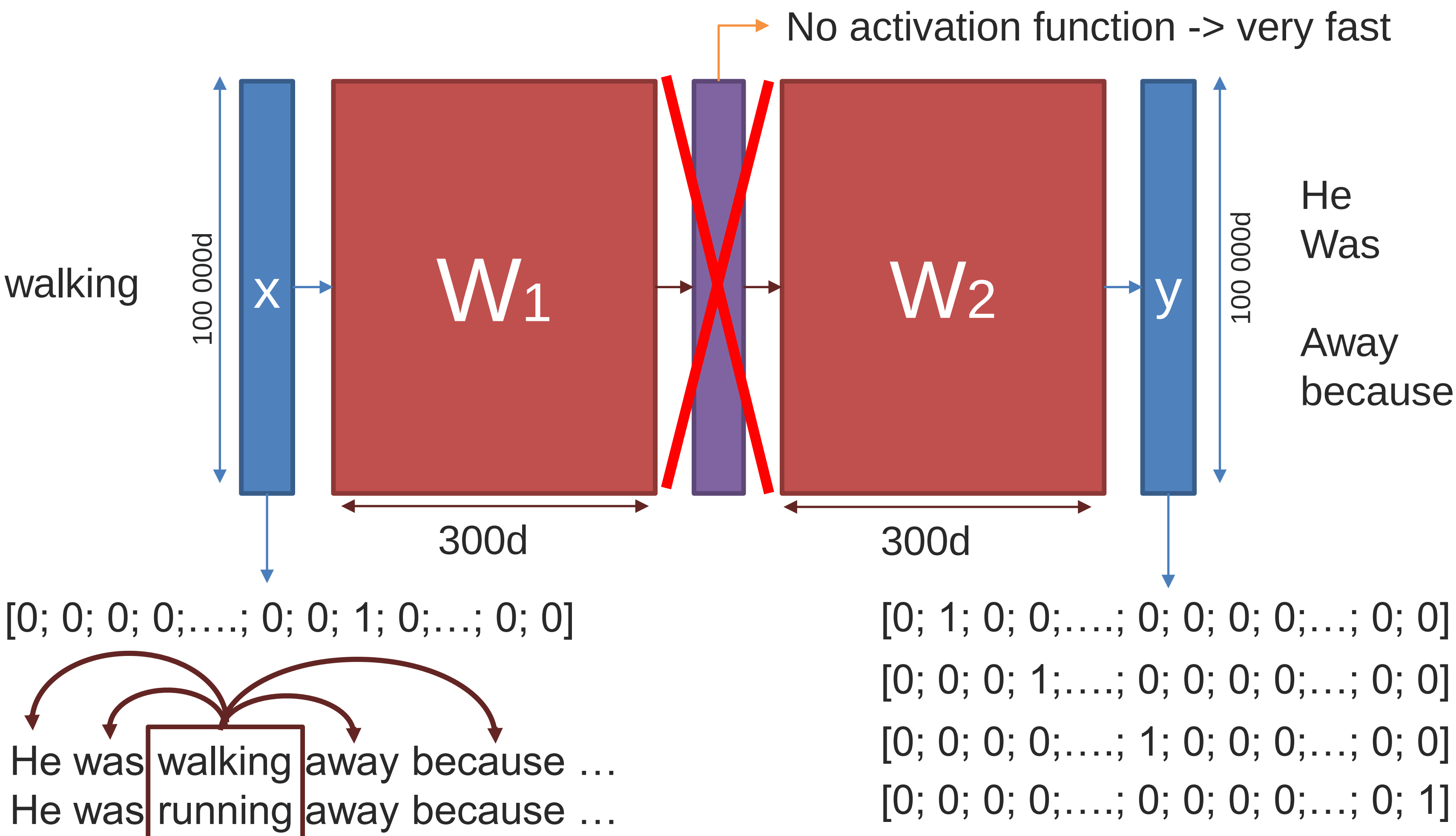
- **Distribution hypothesis:** Approximate the word meaning by its surrounding words
- Words used in a similar context will lie close together

He was walking away because …
He was running away because …

- **Instead of capturing co-occurrence counts directly, predict surrounding words of every word**

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c,j\neq 0}\log p(w_{t+j}|w_t)$$

How to Learn (Better) Language Representations?



Word2vec algorithm: https://code.google.com/p/word2vec/

How to use these word representations

If we would have a vocabulary of 100 000 words:

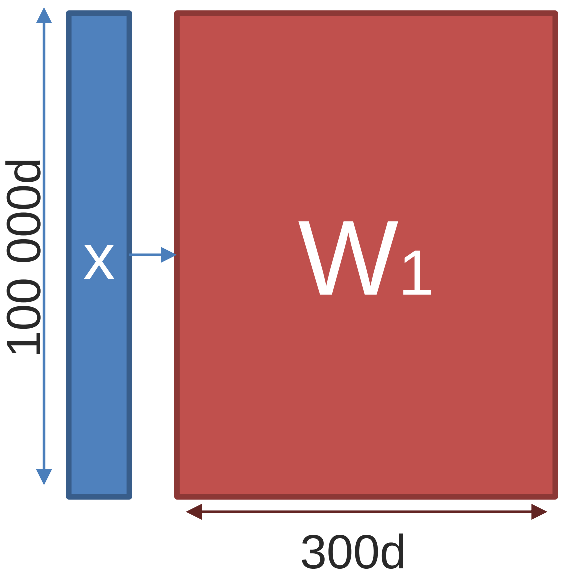
Classic NLP: 100 000 dimensional vector

Walking: [0; 0; 0; 0;….; 0; 0; 1; 0;…; 0; 0]

Running: [0; 0; 0; 0;….; 0; 0; 0; 0;…; 1; 0]

Similarity = 0.0

Transform: x'=x*W



Goal: 300 dimensional vector

Walking: [0,1; 0,0003; 0;….; 0,02; 0.08; 0,05]

Running: [0,1; 0,0004; 0;….; 0,01; 0.09; 0,05]

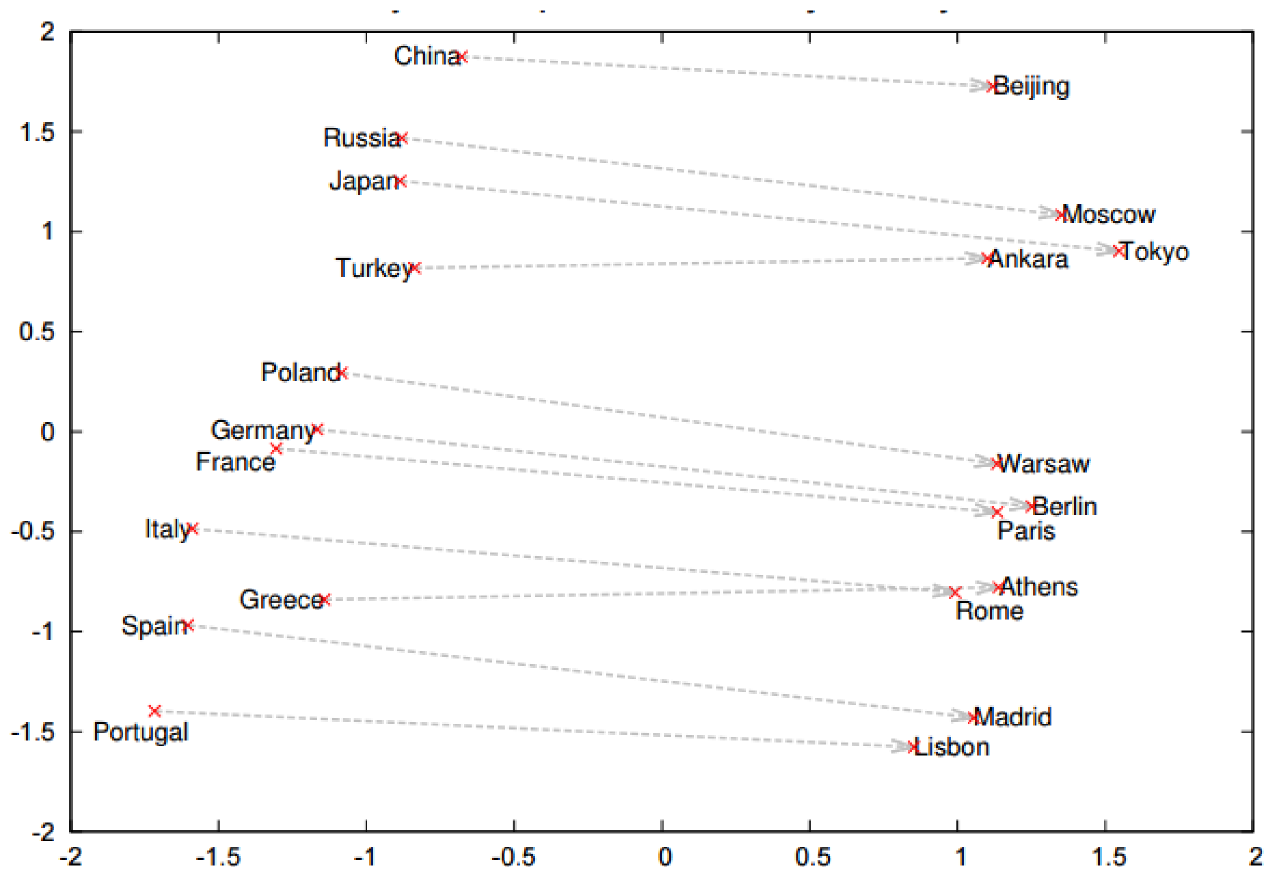
Similarity = 0.9

Vector space models of words

- While learning these word representations, we are actually building a vector space in which all words reside with certain relationships between them
- Encodes both syntactic and semantic relationships
- This vector space allows for algebraic operations:

$$\text{Vec(king)} - \text{vec(man)} + \text{vec(woman)} \approx \text{vec(queen)}$$

Vector space models of words: semantic relationships

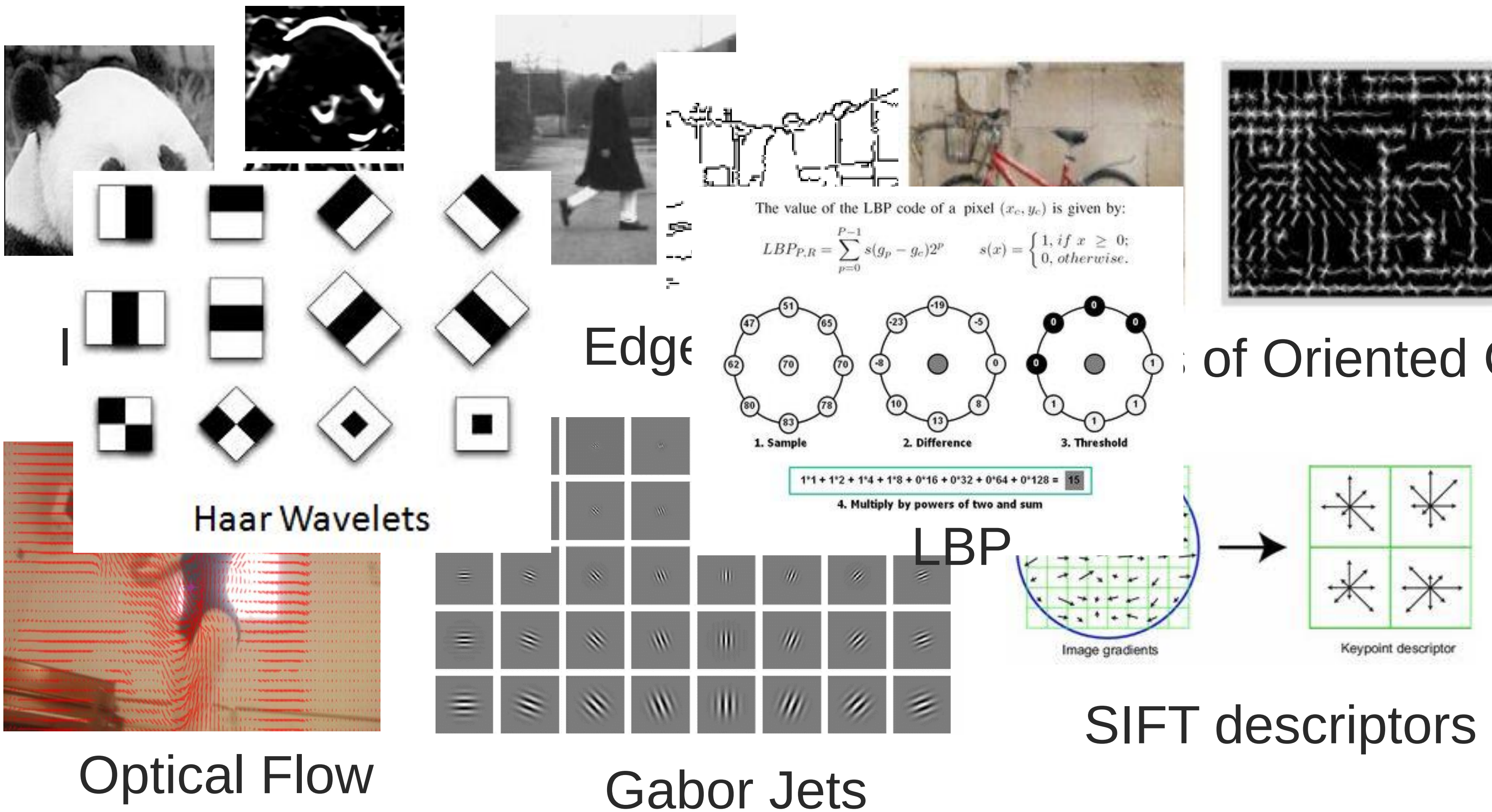


Trained on the Google news corpus with over 300 billion words

Mikolov et al., “Distributed Representations of Words and Phrases and their Compositionality”, NIPS 2013

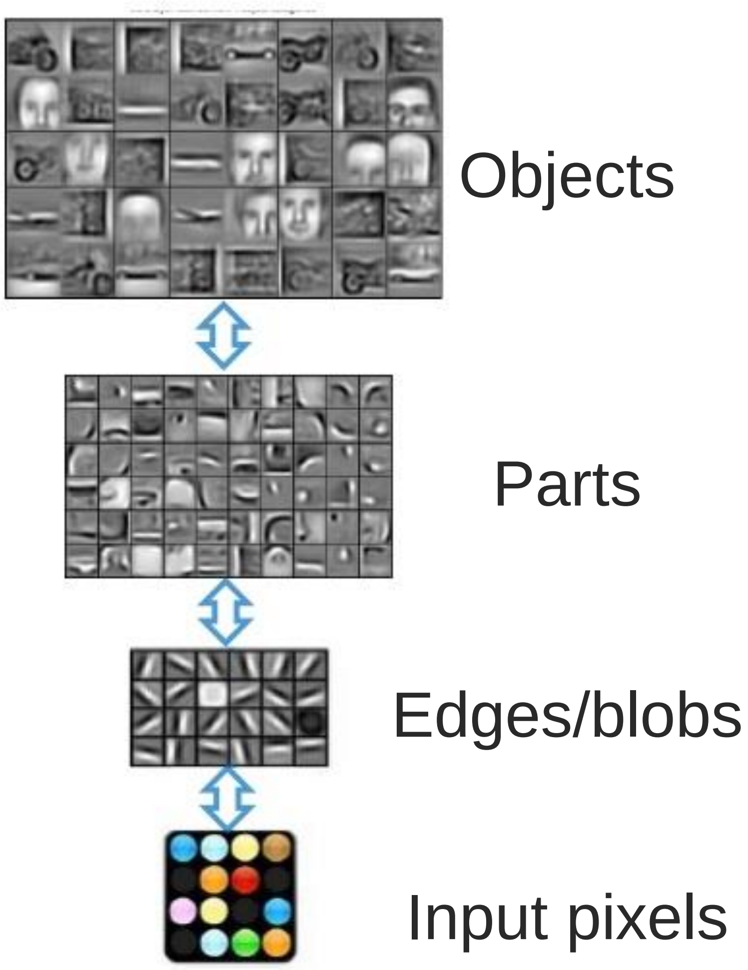
Unimodal representations: Visual Modality

Visual Descriptors



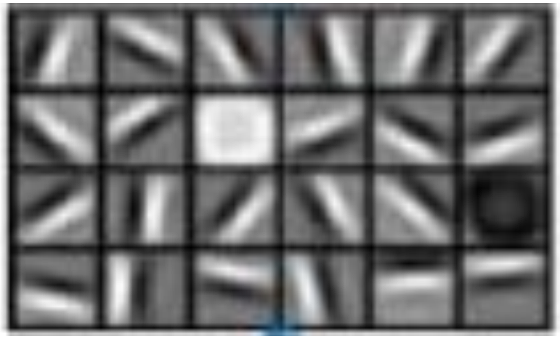
Why use Convolutional Neural Networks

- Using basic Multi Layer Perceptrons does not work well for images
- Intention to build more abstract representation as we go up every layer



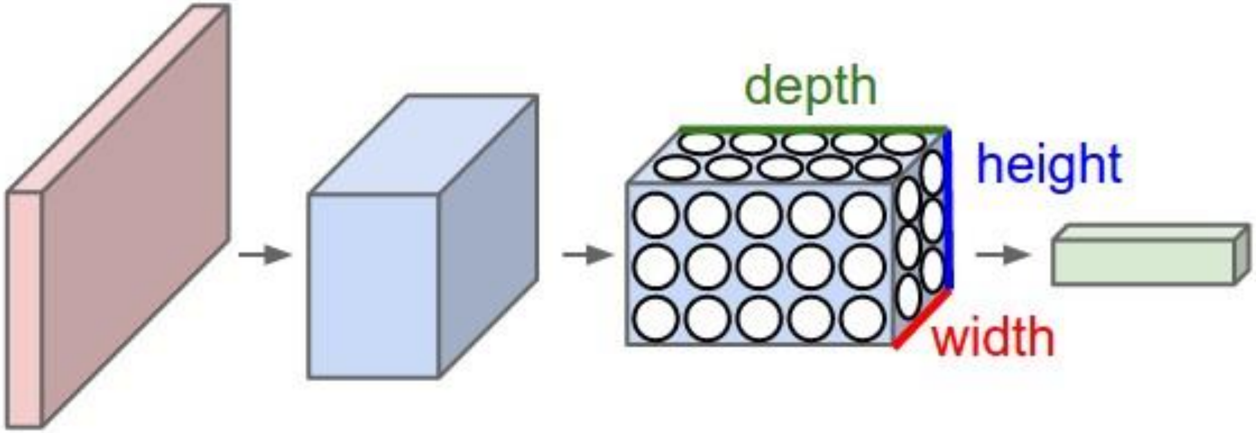
Why not just use an MLP for images (1)?

- MLP connects each pixel in an image to each neuron
- Does not exploit redundancy in image structure
  - Detecting edges, blobs
  - Don't need to treat the top left of image differently from the center
- Too many parameters
  - For a small 200×200 pixel RGB image the first matrix would have $120000 \times n$ parameters for the first layer alone
- MLP does not exploit translation invariance
- MLP does not necessarily encourage visual abstraction

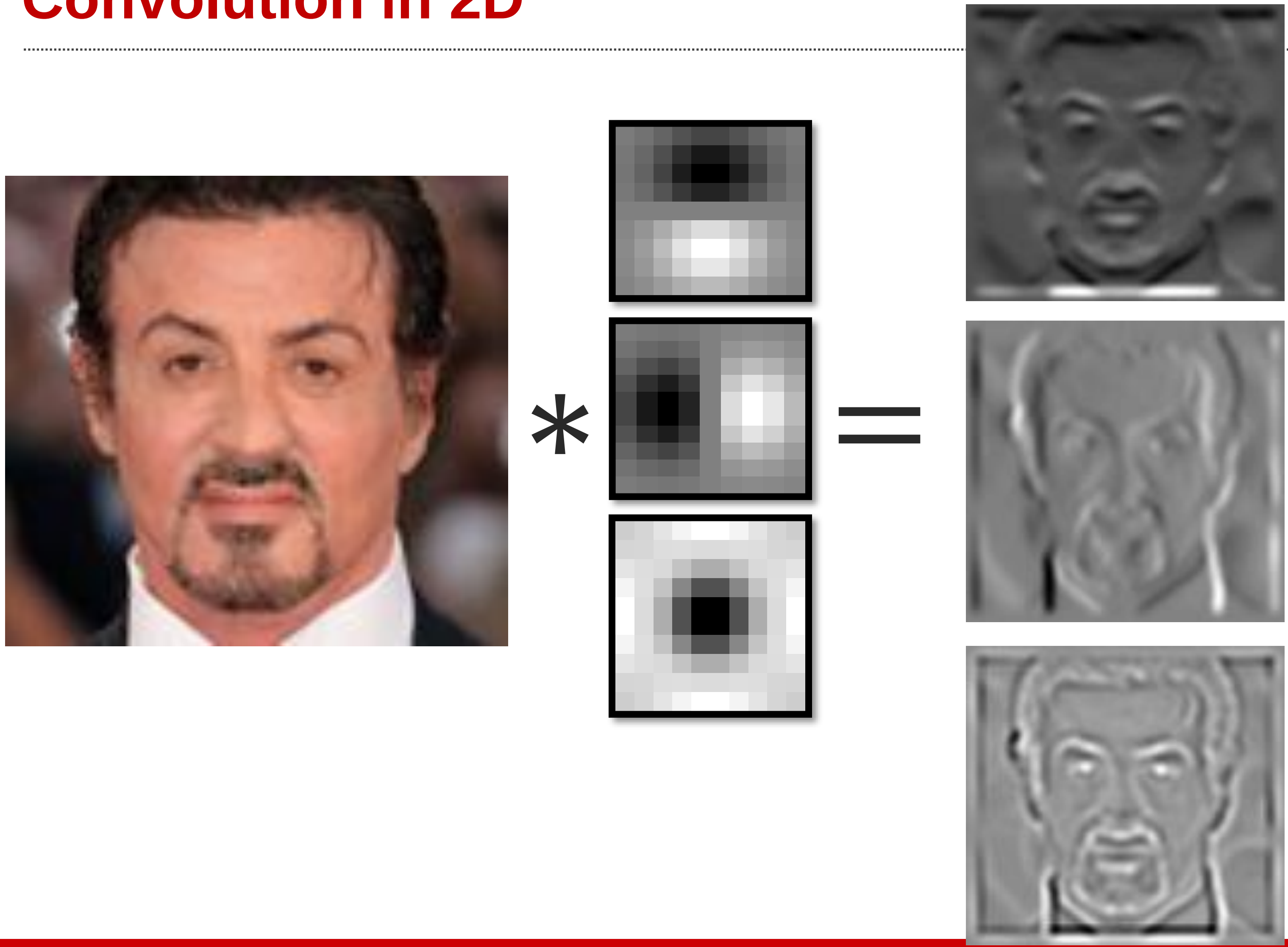


Main differences of CNN from MLP

- Addition of:
  - Convolution layer
  - Pooling layer
- Everything else is the same (loss, score and optimization)
- MLP layer is called Fully Connected layer

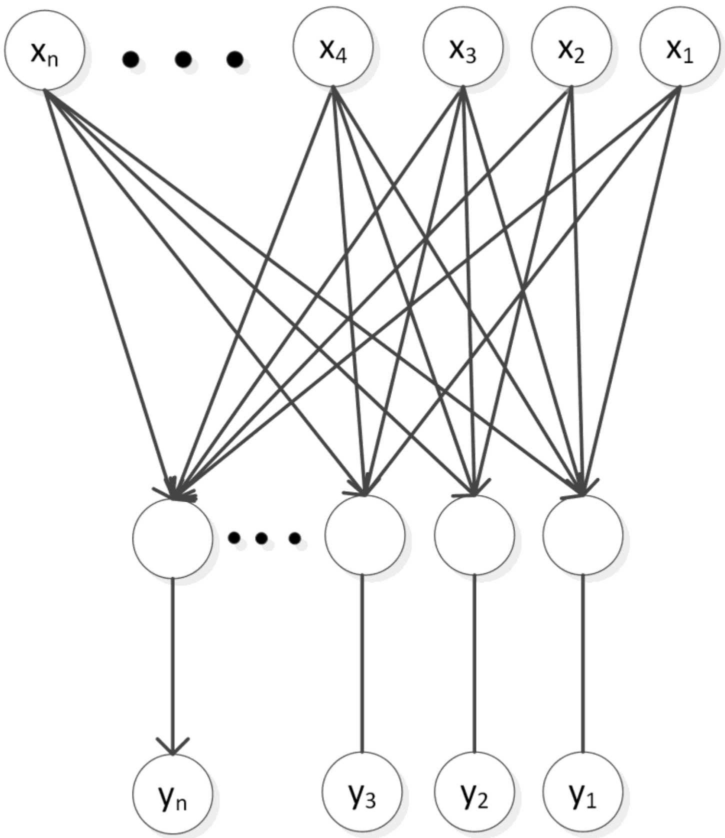


Convolution in 2D



Fully connected layer

- Weighted sum followed by an activation function



Input

Weighted sum
$Wx + b$

Activation function

Output

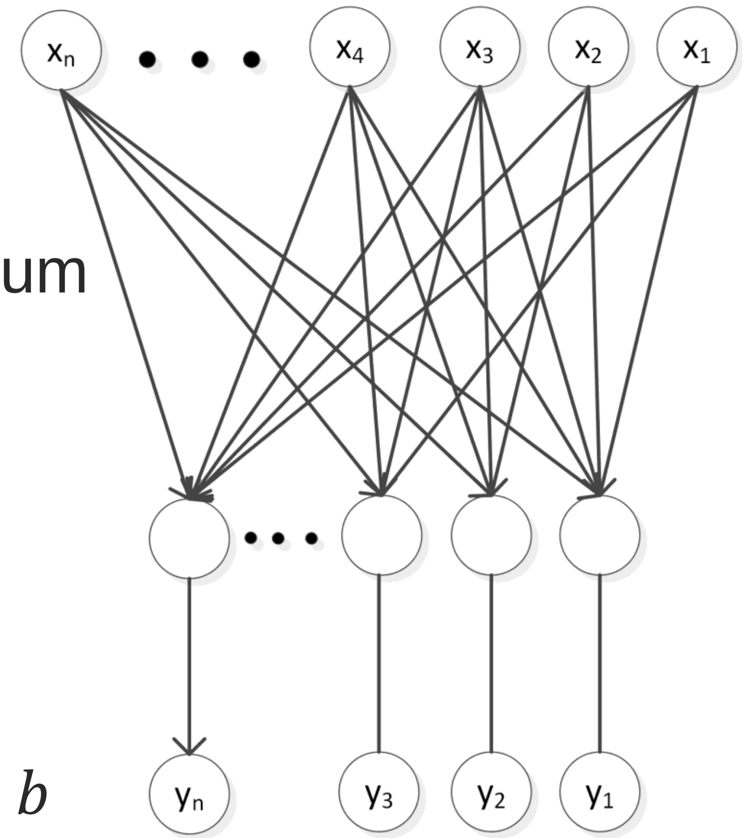
$y = f(Wx + b)$

Convolution as MLP (1)

- Remove activation

Input

Weighted sum
$Wx + b$



$y = Wx + b$

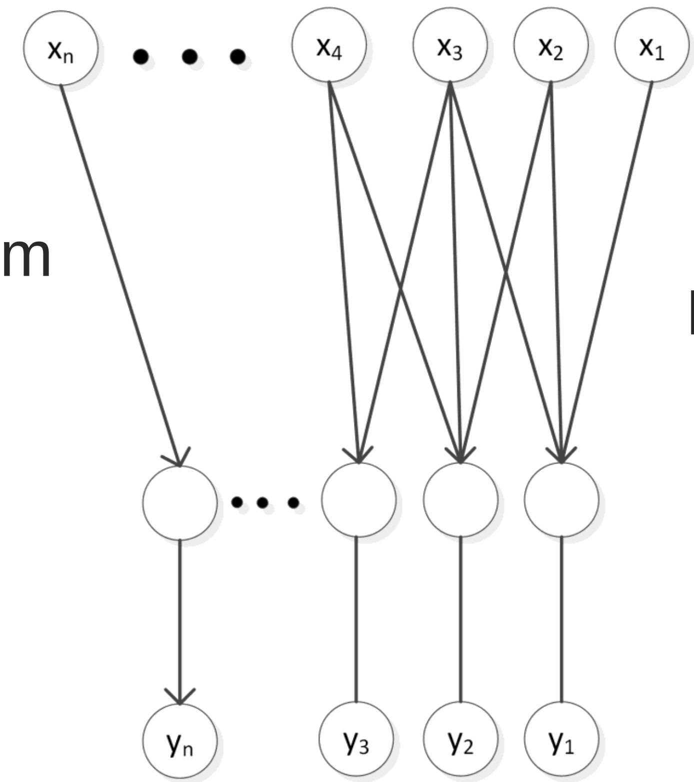
Kernel $\boxed{\boldsymbol{w_1}}\boxed{\boldsymbol{w_2}}\boxed{\boldsymbol{w_3}}$

Convolution as MLP (2)

- Remove redundant links making the matrix W sparse (optionally remove the bias term)

Input

Weighted sum
Wx

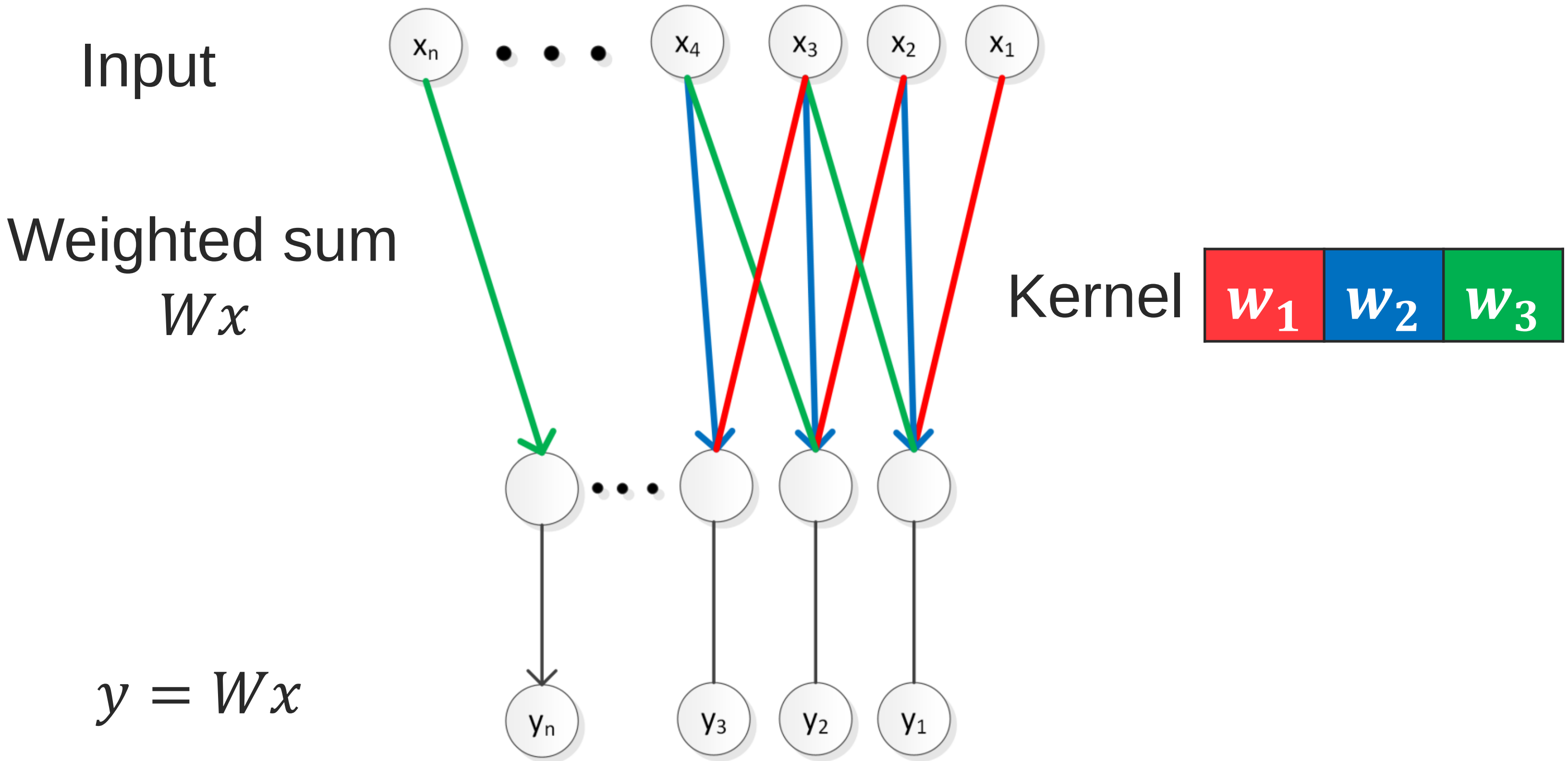


Kernel $\boxed{w_1}\boxed{w_2}\boxed{w_3}$

$y = Wx$

Convolution as MLP (3)

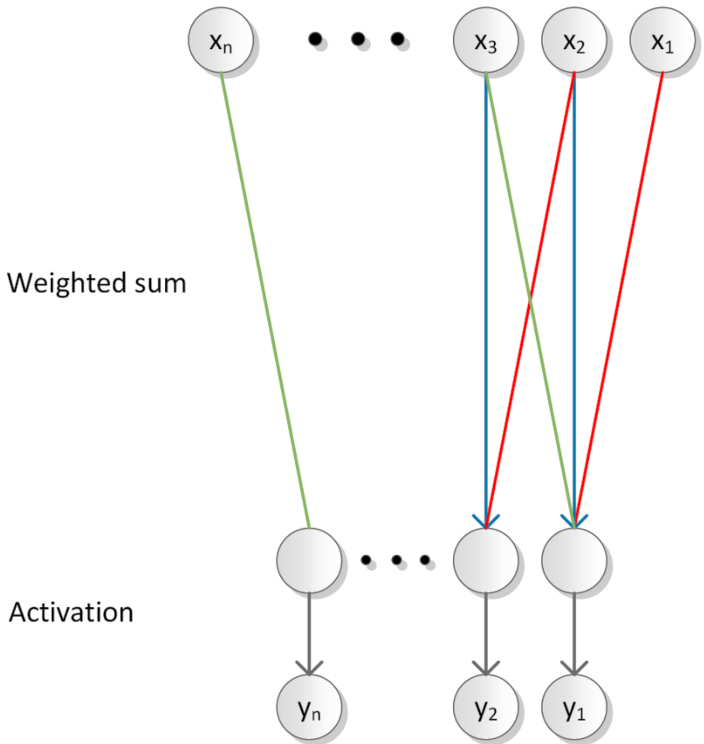
- We can also share the weights in matrix W not to do redundant computation



How do we do convolution in MLP recap

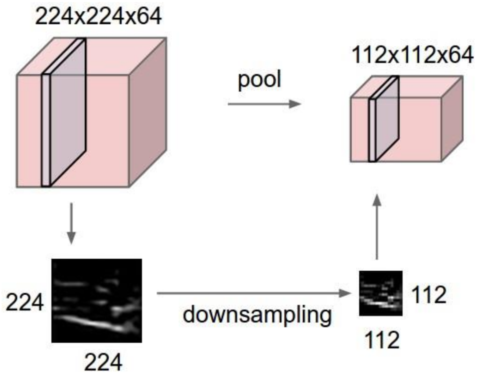
- Not a fully connected layer anymore
- Shared weights
  - Same colour indicates same (shared) weight

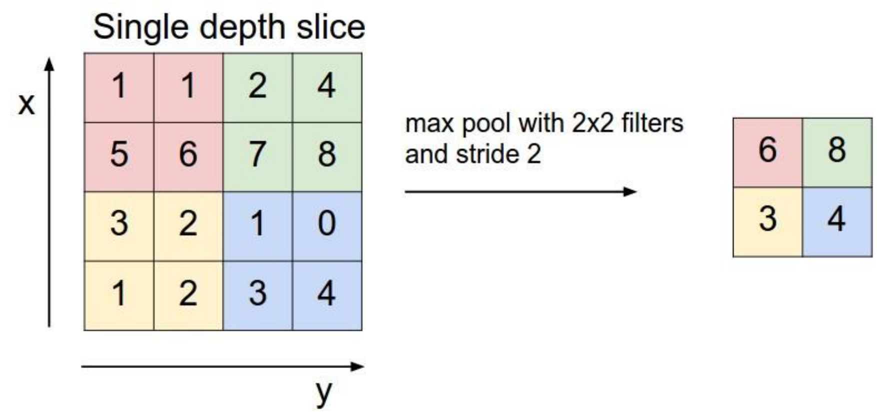
$$W = \begin{pmatrix} w_1 & w_2 & w_3 & & 0 & 0 & 0 \\ 0 & w_1 & w_2 & \cdots & 0 & 0 & 0 \\ 0 & 0 & w_1 & & 0 & 0 & 0 \\ & \vdots & & \ddots & & \vdots & \\ 0 & 0 & 0 & & w_3 & 0 & 0 \\ 0 & 0 & 0 & \cdots & w_2 & w_3 & 0 \\ 0 & 0 & 0 & & w_1 & w_2 & w_3 \end{pmatrix}$$



Pooling layer

- Used for sub-sampling



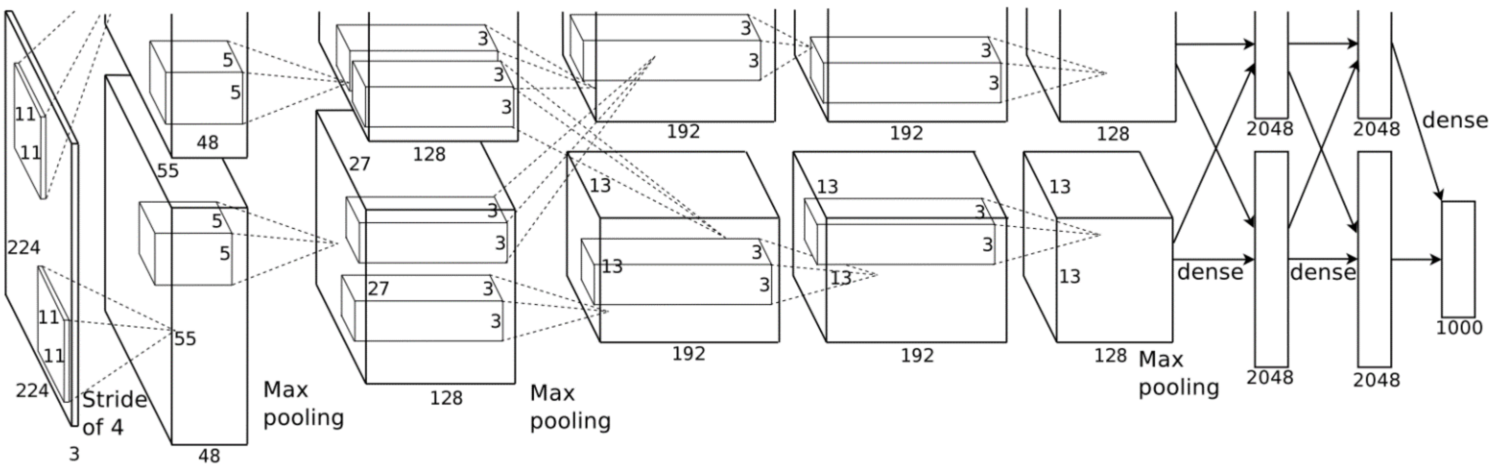


Pick the maximum value from input using a smooth and differentiable approximation

$$y = \frac{\sum_{i=1}^{n} x_i e^{\alpha x_i}}{\sum_{i=1}^{n} e^{\alpha x_i}}$$

Example: AlexNet Model

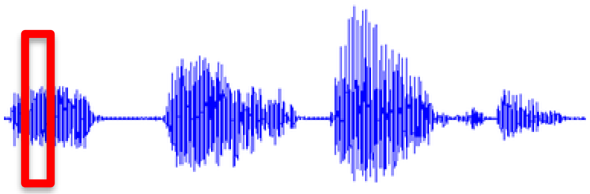
- Used for object classification task
  - 1000 way classification task – pick one



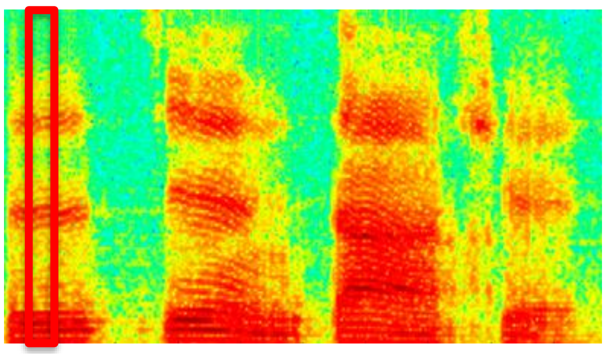
Unimodal representations: Acoustic Modality

Unimodal Classification – Acoustic Modality

Digitalized acoustic signal



- Sampling rates: 8~96kHz
- Bit depth: 8, 16 or 24 bits
- Time window size: 20ms
  - Offset: 10ms



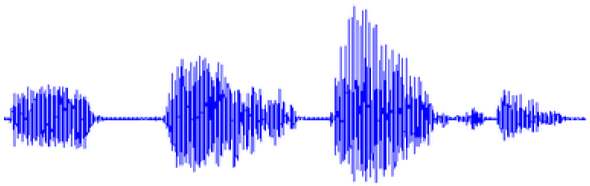
Spectogram

Input observation x_i

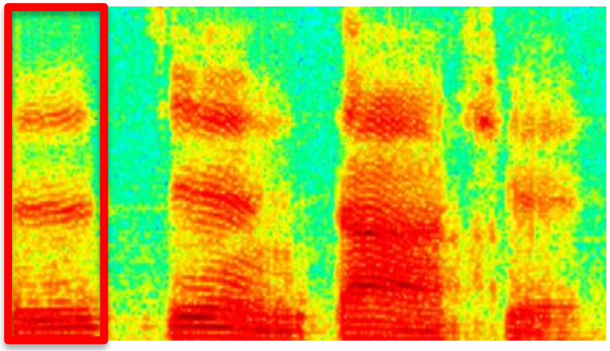
0.21
0.14
0.56
0.45
0.9
0.98
0.75
0.34
0.24
0.11
0.02

Unimodal Classification – Acoustic Modality

Digitalized acoustic signal



- Sampling rates: 8~96kHz
- Bit depth: 8, 16 or 24 bits
- Time window size: 20ms
  - Offset: 10ms



Spectogram

Input observation x_i

0.21
0.14
0.56
0.45
0.9
0.98
0.75
0.34
0.24
0.11
0.02
0.24
0.26
0.58
0.9
0.99
0.79
0.45
0.34
0.24

⋮

Emotion ?

Spoken word ?

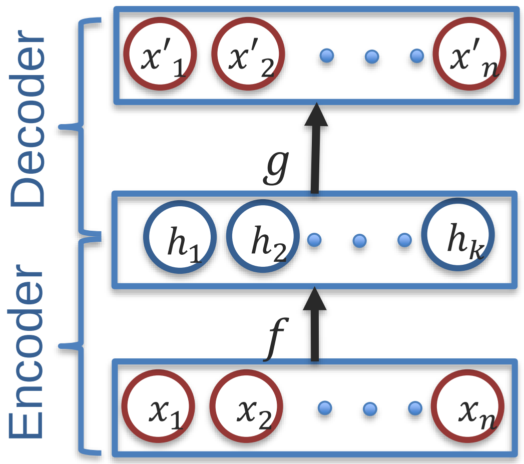
Voice quality ?

Audio representation for speech recognition

- Speech recognition systems historically much more complex than vision systems – language models, vocabularies etc.
- Large breakthrough of using representation learning instead of hand-crafted features
  - [Hinton et al., Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups, 2012]
- A huge boost in performance (up to 30% on some datasets)

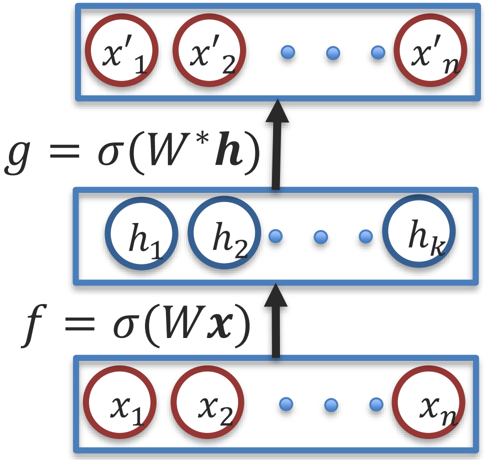
Autoencoders

- What does auto mean?
  - Greek for self – self encoding
- Feed forward network intended to reproduce the input
- Two parts encoder/decoder
  - $x' = f(g(x))$ – score function
  - g - encoder
  - f - decoder



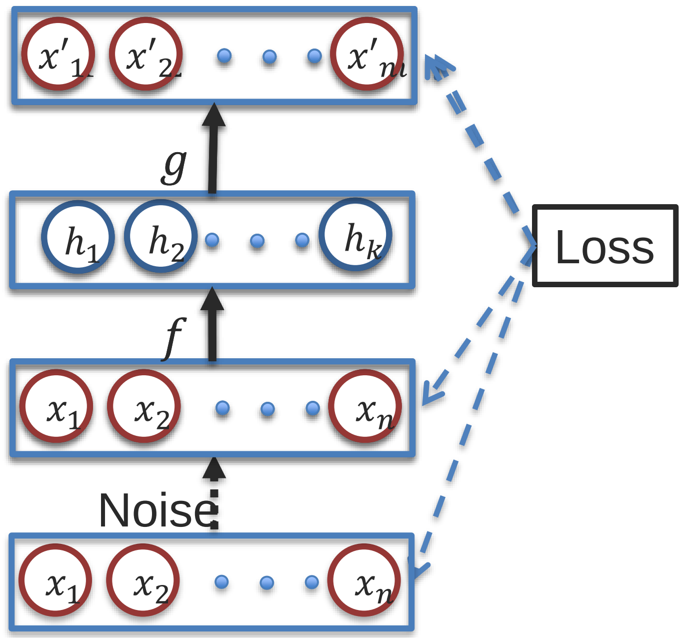
Autoencoders

- Mostly follows Neural Network structure
  - A matrix multiplication followed by a sigmoid
- Activation will depend on type of $\boldsymbol{x}$
  - Sigmoid for binary
  - Linear for real valued
- Often we use tied weights to force the sharing of weights in encoder/decoder
  - $W^* = W^T$
- word2vec is actually a bit similar to autoencoder (except for the auto part)



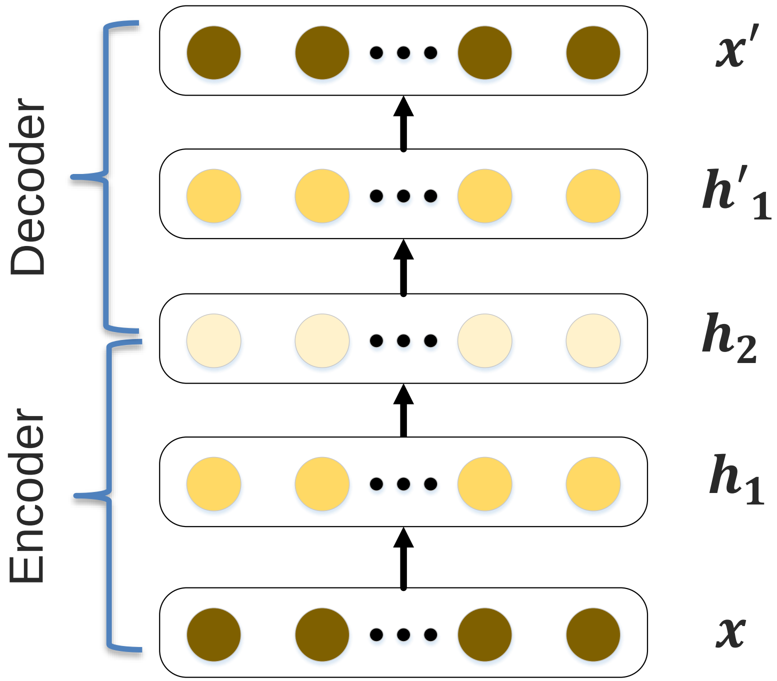
Denoising autoencoder

- Simple idea
  - Add noise to input $\boldsymbol{x}$ but learn to reconstruct original
- Leads to a more robust representation and prevents copying
- Learns what the relationship is to represent a certain $\boldsymbol{x}$
- Different noise added during each epoch



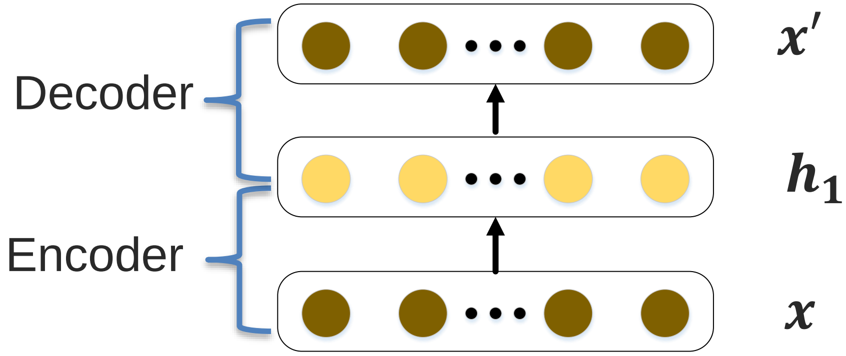
Stacked autoencoders

- Can stack autoencoders as well
- Each encoding unit has a corresponding decoder
- Inference as before is feed forward structure, but now with more hidden layers



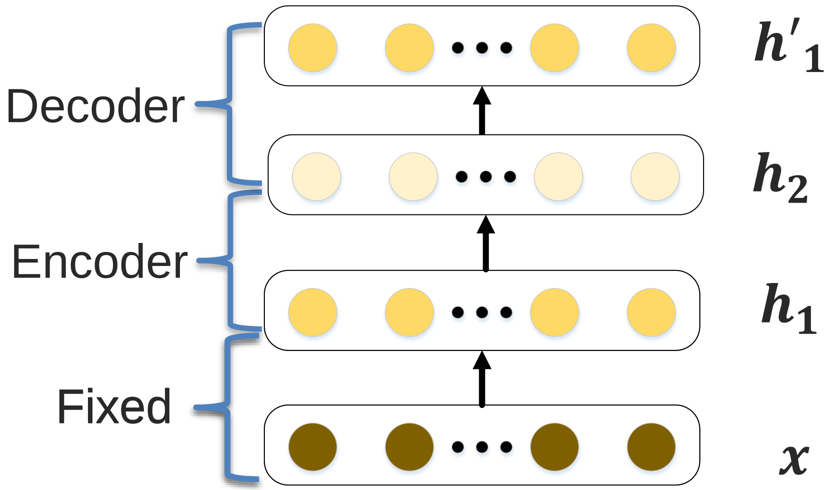
Stacked autoencoders

- Greedy layer-wise training
- Start with training first layer
  - Learn to encode $\boldsymbol{x}$ to $\boldsymbol{h_1}$ and to decode $\boldsymbol{x}$ from $\boldsymbol{h_1}$
  - Use backpropagation



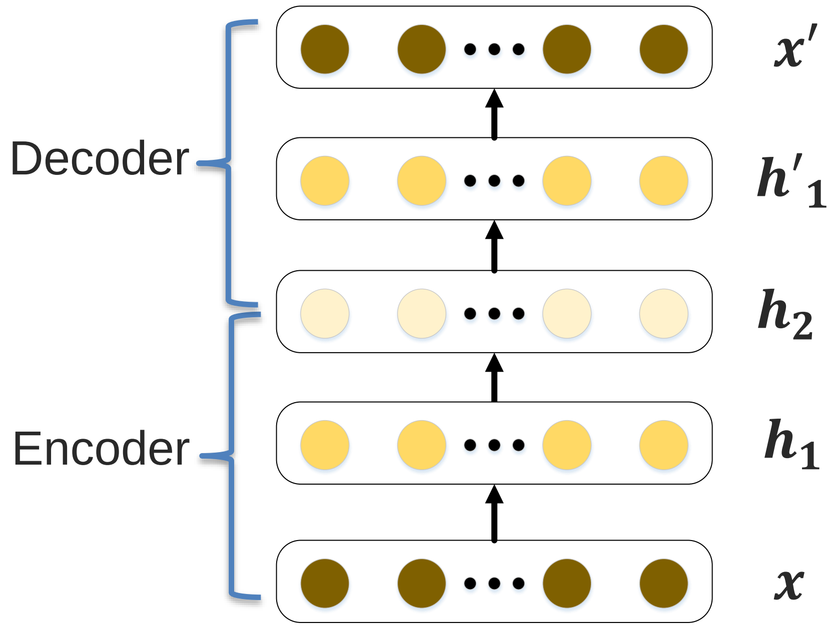
Stacked autoencoders

- Map from all $\boldsymbol{x}$'s to $\boldsymbol{h_1}$'s
  - Discard decoder for now
- Train the second layer
  - Learn to encode $\boldsymbol{h_1}$ to $\boldsymbol{h_2}$ and to decode $\boldsymbol{h_2}$ from $\boldsymbol{h_1}$
  - Repeat for as many layers



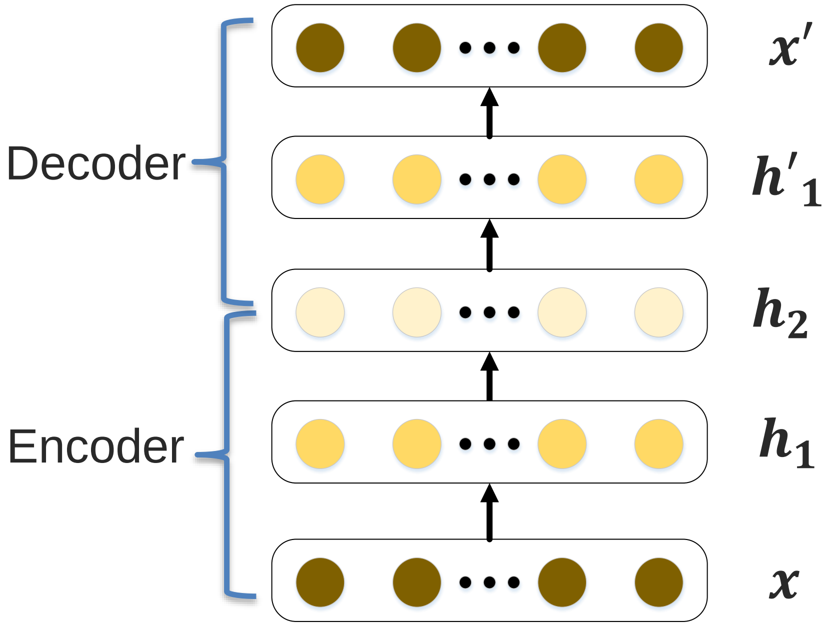
Stacked autoencoders

- Reconstruct using previously learned decoders mappings
- Fine-tune the full network end-to-end



Stacked denoising autoencoders

- Can extend this to a denoising model
- Add noise when training each of the layers
    - Often with increasing amount of noise per layer
    - 0.1 for first, 0.2 for second, 0.3 for third

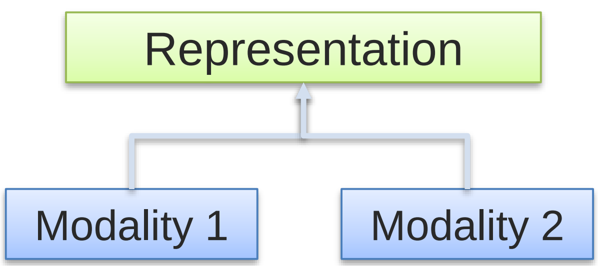


Multimodal Representations

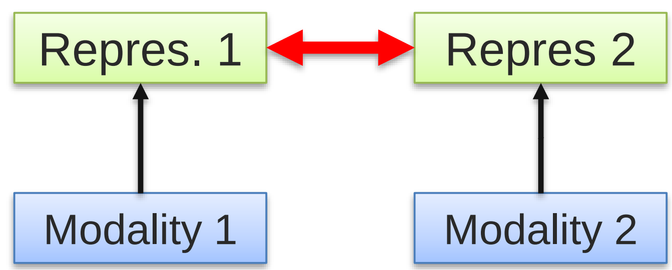
Core Challenge: Representation

Definition: Learning how to represent and summarize multimodal data in away that exploits the complementarity and redundancy.

A Joint representations:

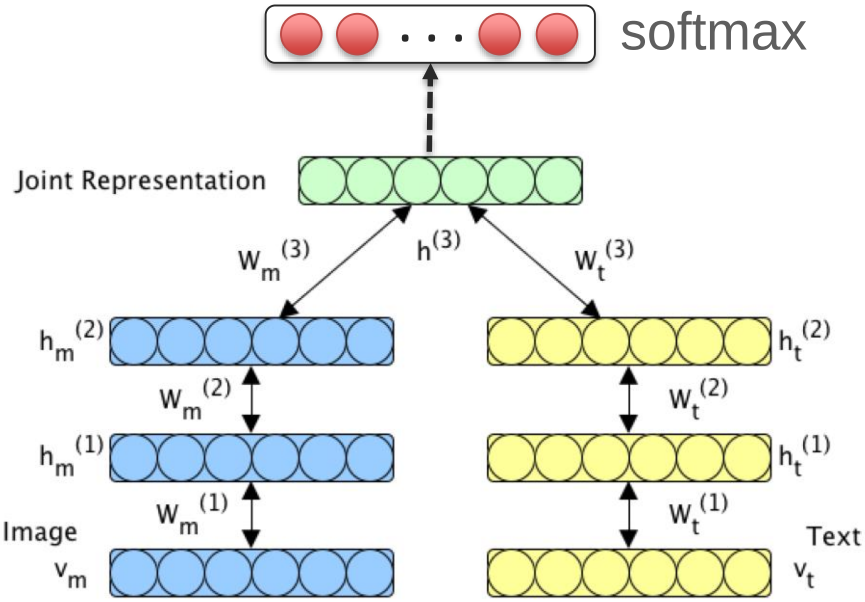


B Coordinated representations:



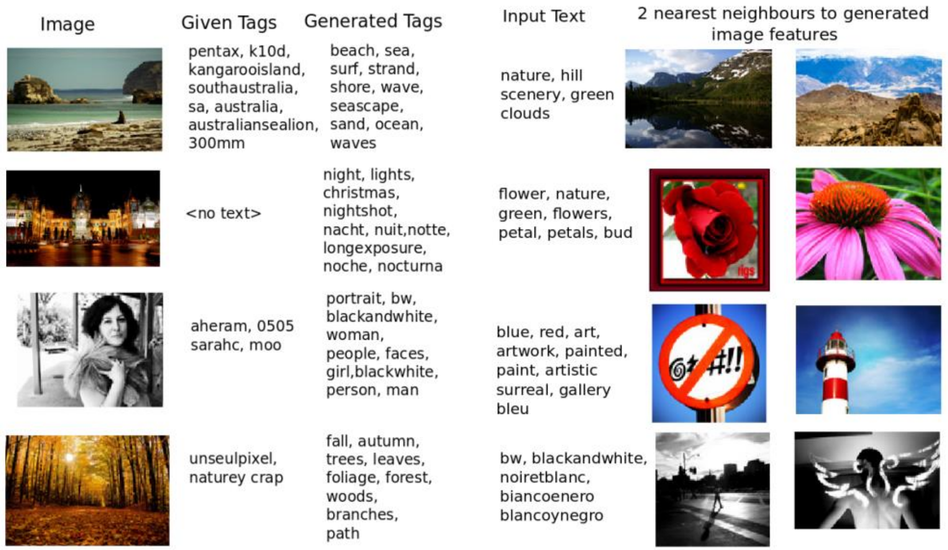
Deep Multimodal Boltzmann machines

- 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]

Deep Multimodal Boltzmann machines

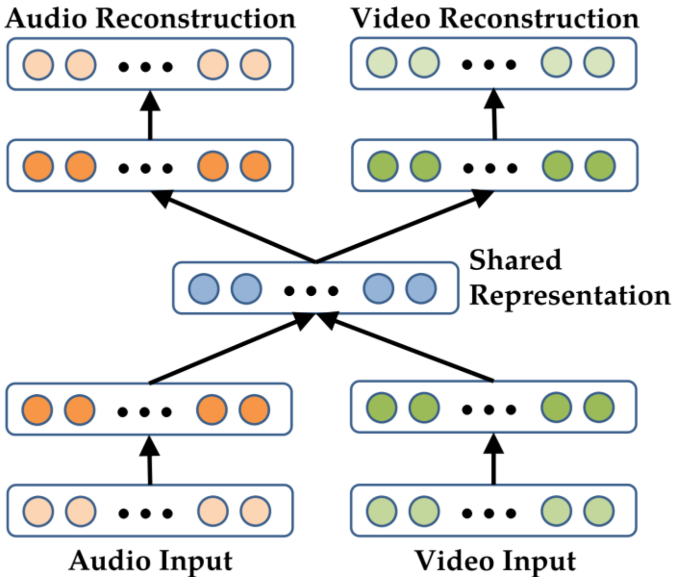


Model	MAP	Prec@50
Random	0.124	0.124
SVM (Huiskes et al., 2010)	0.475	0.758
LDA (Huiskes et al., 2010)	0.492	0.754
DBM	0.526 ± 0.007	0.791 ± 0.008
DBM (using unlabelled data)	**0.585** ± 0.004	**0.836** ± 0.004

Srivastava and Salakhutdinov, “Multimodal Learning with Deep Boltzmann Machines”, NIPS 2012

Deep Multimodal autoencoders

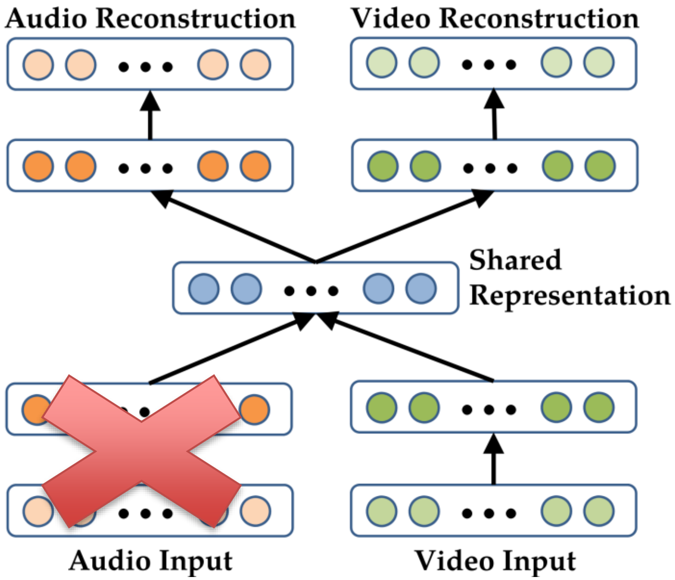
- A deep representation learning approach
- A bimodal auto-encoder
  - Used for Audio-visual speech recognition



[Ngiam et al., Multimodal Deep Learning, 2011]

Deep Multimodal autoencoders - training

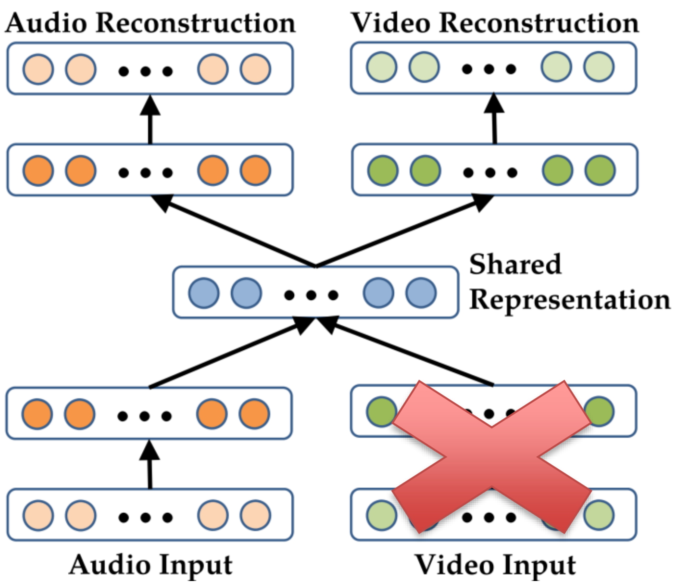
- Individual modalities can be pretrained
  - RBMs
  - Denoising Autoencoders
- To train the model to reconstruct the other modality
  - Use both
  - Remove audio



[Ngiam et al., Multimodal Deep Learning, 2011]

Deep Multimodal autoencoders - training

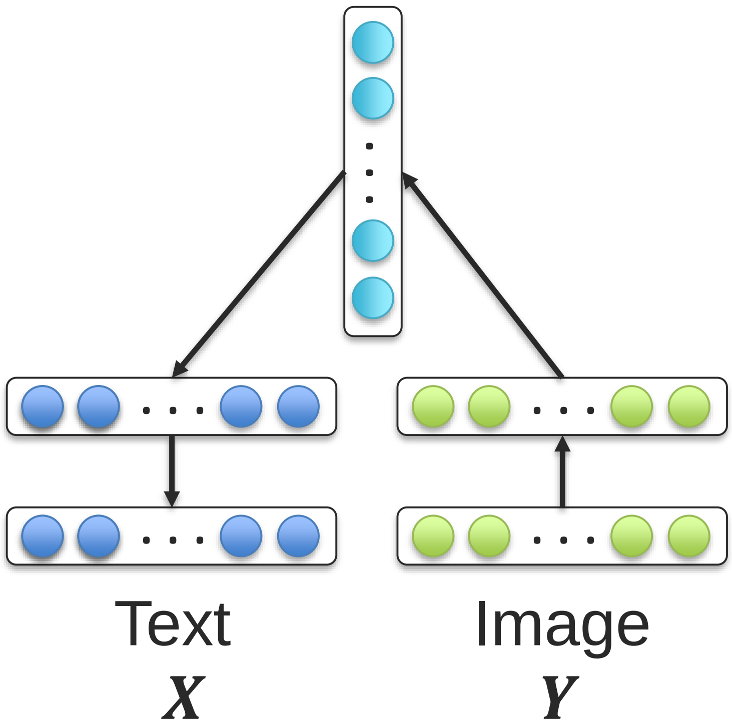
- Individual modalities can be pretrained
  - RBMs
  - Denoising Autoencoders
- To train the model to reconstruct the other modality
  - Use both
  - Remove audio
  - Remove video



[Ngiam et al., Multimodal Deep Learning, 2011]

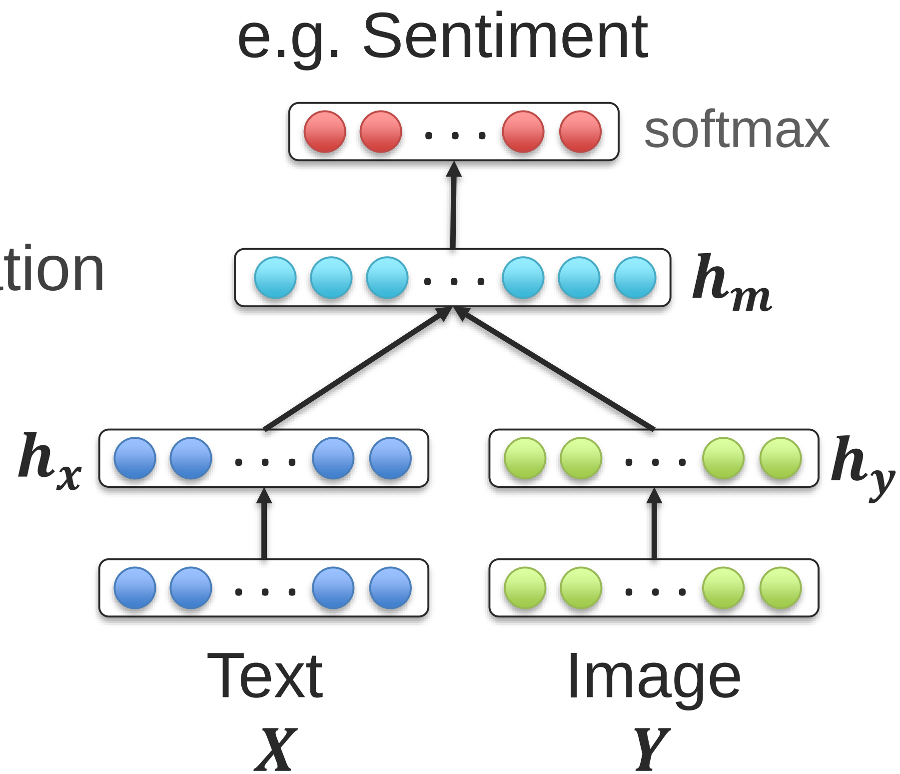
Multimodal Encoder-Decoder

- Visual modality often encoded using CNN
- Language modality will be decoded using LSTM
  - A simple multilayer perceptron will be used to translate from visual (CNN) to language (LSTM)



Multimodal Joint Representation

- For supervised learning tasks
- Joining the unimodal representations:
  - Simple concatenation
  - Element-wise multiplication or summation
  - Multilayer perceptron
- How to explicitly model both unimodal and bimodal interactions?



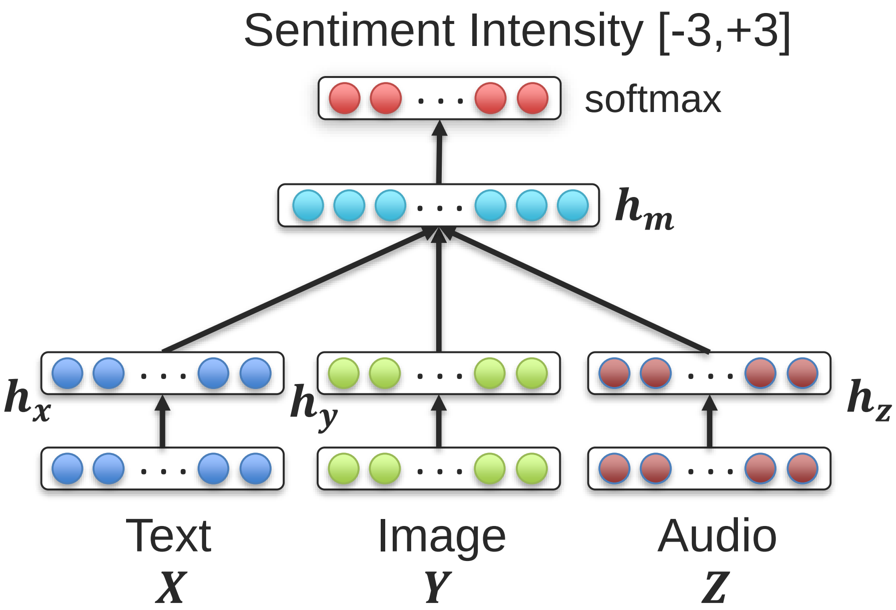
Multimodal Sentiment Analysis

MOSI dataset (Zadeh et al, 2016)

- 2199 subjective video segments
- Sentiment intensity annotations
- 3 modalities: text, video, audio

Multimodal joint representation:

$$h_m = f(W \cdot [h_x, h_y, h_z])$$



Unimodal, Bimodal and Trimodal Interactions

Speaker's behaviors → **Sentiment Intensity**

Unimodal

- "This movie is sick" → ? → *Ambiguous !*
- "This movie is fair" → +
- Smile → +
  - → *Unimodal cues*
- Loud voice → ? → *Ambiguous !*

Bimodal

- "This movie is sick" + Smile → ++
- "This movie is sick" + Frown → – –
  - → *Resolves ambiguity (bimodal interaction)*
- "This movie is sick" + Loud voice → ? → *Still Ambiguous !*

Trimodal

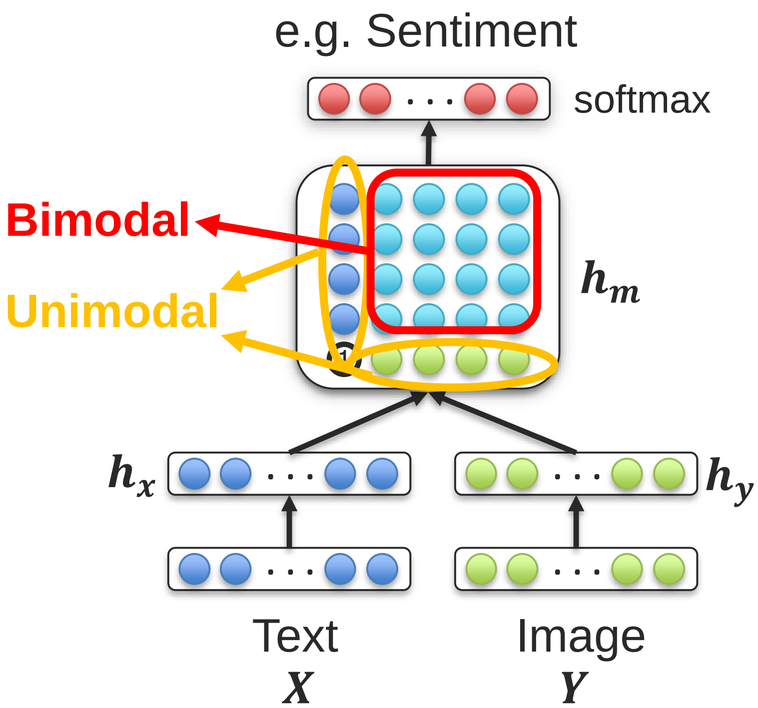
- "This movie is sick" + Smile + Loud voice → +++
- "This movie is fair" + Smile + Loud voice → +
  - → *Different trimodal interactions !*

Multimodal Tensor Fusion Network (TFN)

Models both unimodal and bimodal interactions:

$$\boldsymbol{h}_m = \begin{bmatrix} \boldsymbol{h}_x \\ 1 \end{bmatrix} \otimes \begin{bmatrix} \boldsymbol{h}_y \\ 1 \end{bmatrix} = \begin{bmatrix} \boxed{\boldsymbol{h}_x} & \boxed{\boldsymbol{h}_x \otimes \boldsymbol{h}_y} \\ 1 & \boxed{\boldsymbol{h}_y} \end{bmatrix}$$

Important !

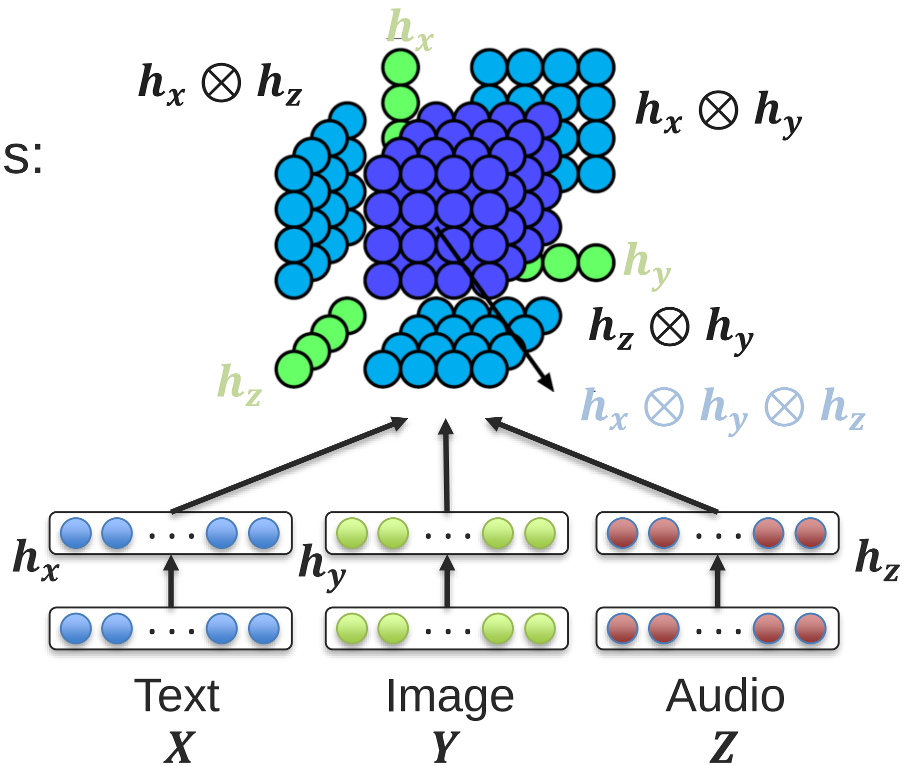


[Zadeh, Jones and Morency, **EMNLP 2017**]

Multimodal Tensor Fusion Network (TFN)

Can be extended to three modalities:

$$h_m = \begin{bmatrix} h_x \\ 1 \end{bmatrix} \otimes \begin{bmatrix} h_y \\ 1 \end{bmatrix} \otimes \begin{bmatrix} h_z \\ 1 \end{bmatrix}$$



Explicitly models unimodal, bimodal and trimodal interactions !

[Zadeh, Jones and Morency, **EMNLP 2017**]

Experimental Results – MOSI Dataset

Multimodal Baseline	Binary		5-class	Regression	
	Acc(%)	F1	Acc(%)	MAE	r
Random	50.2	48.7	23.9	1.88	-
C-MKL	73.1	75.2	35.3	-	-
SAL-CNN	73.0	-	-	-	-
SVM-MD	71.6	72.3	32.0	1.10	0.53
RF	71.4	72.1	31.9	1.11	0.51
TFN	**77.1**	**77.9**	**42.0**	**0.87**	**0.70**
Human	85.7	87.5	53.9	0.71	0.82
Δ^{SOTA}	$\uparrow$ 4.0	$\uparrow$ 2.7	$\uparrow$ 6.7	$\downarrow$ 0.23	$\uparrow$ 0.17

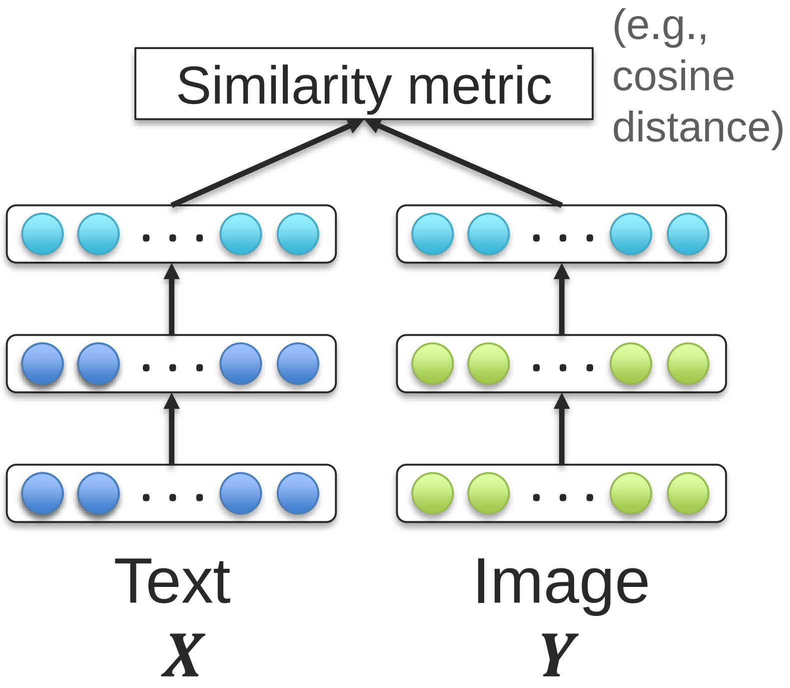
Improvement over State-Of-The-Art

Baseline	Binary		5-class	Regression	
	Acc(%)	F1	Acc(%)	MAE	r
$\text{TFN}_{language}$	74.8	75.6	38.5	0.99	0.61
TFN_{visual}	66.8	70.4	30.4	1.13	0.48
$\text{TFN}_{acoustic}$	65.1	67.3	27.5	1.23	0.36
$\text{TFN}_{bimodal}$	75.2	76.0	39.6	0.92	0.65
$\text{TFN}_{trimodal}$	74.5	75.0	38.9	0.93	0.65
$\text{TFN}_{notrimodal}$	75.3	76.2	39.7	0.919	0.66
TFN	**77.1**	**77.9**	**42.0**	**0.87**	**0.70**
TFN_{early}	75.2	76.2	39.0	0.96	0.63

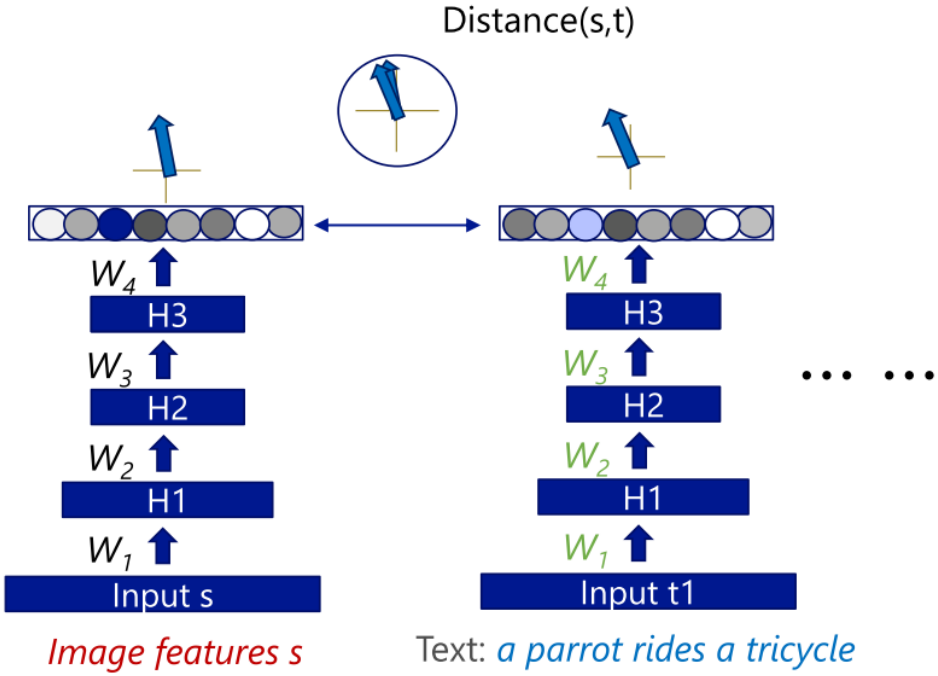
Coordinated Multimodal Representations

Coordinated Multimodal Representations

Learn (unsupervised) two or more coordinated representations from multiple modalities. A loss function is defined to bring closer these multiple representations.

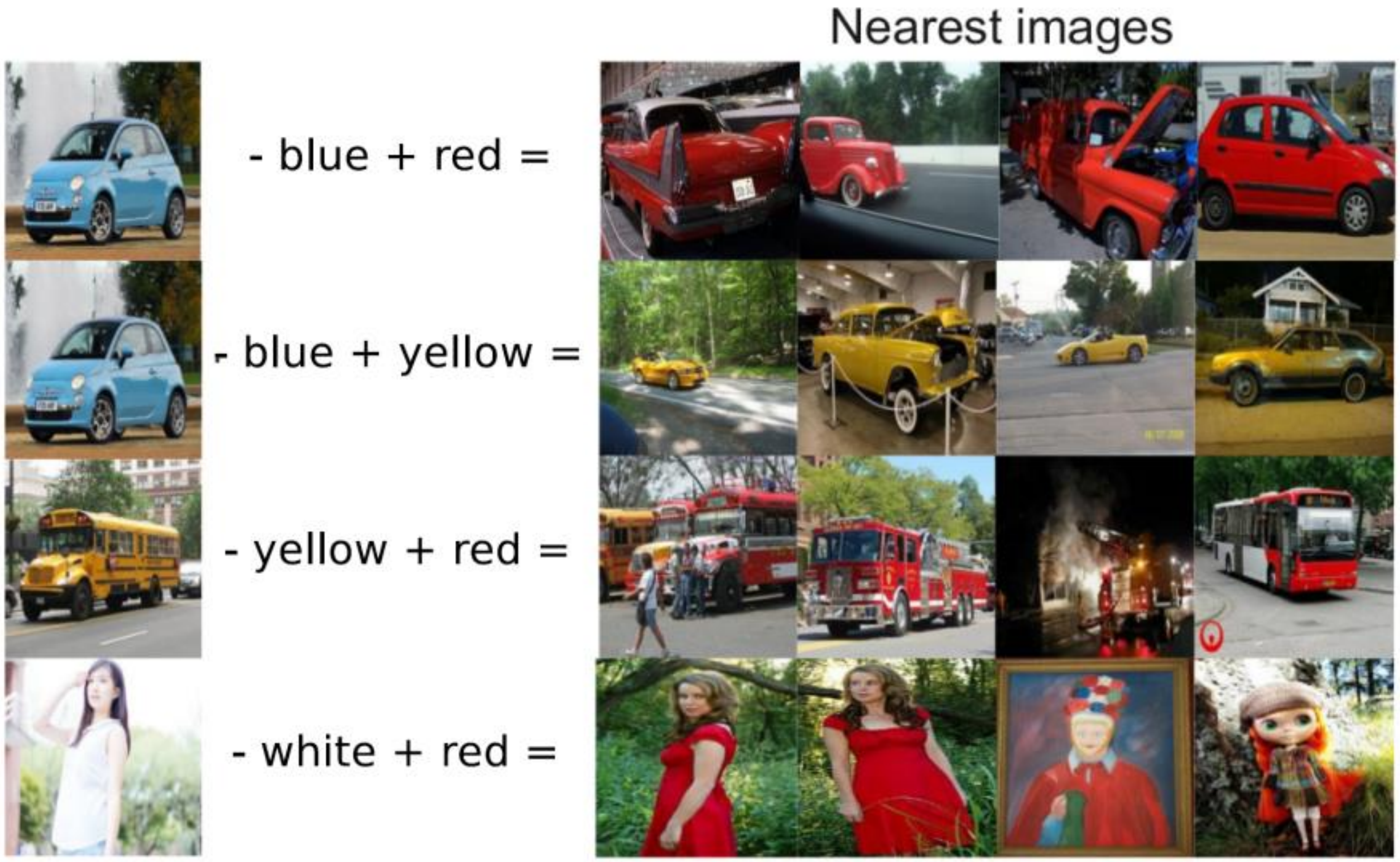


Coordinated Multimodal Embeddings



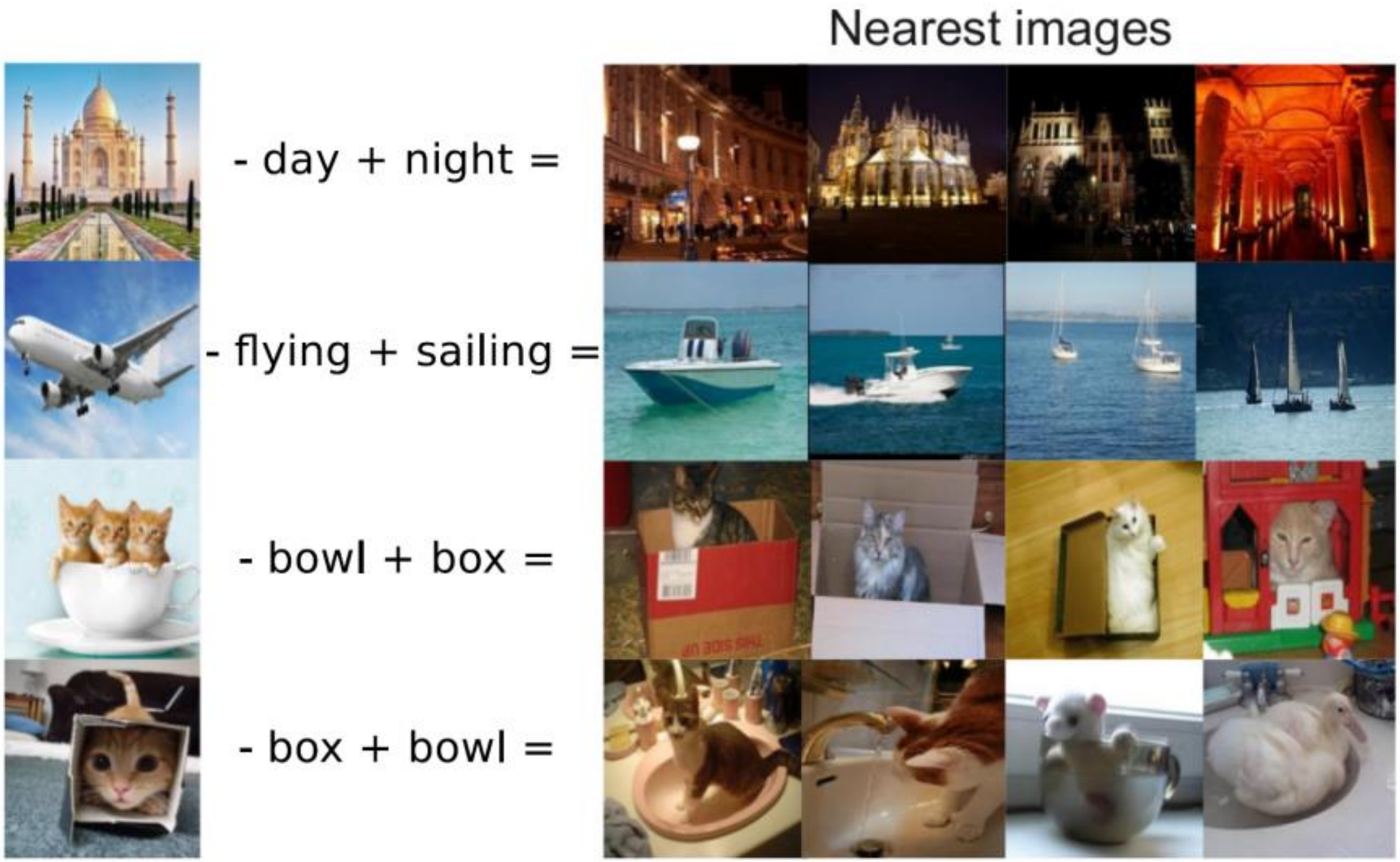
[Huang et al., Learning Deep Structured Semantic Models for Web Search using Clickthrough Data, 2013]

Multimodal Vector Space Arithmetic



[Kiros et al., Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models, 2014]

Multimodal Vector Space Arithmetic



[Kiros et al., Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models, 2014]

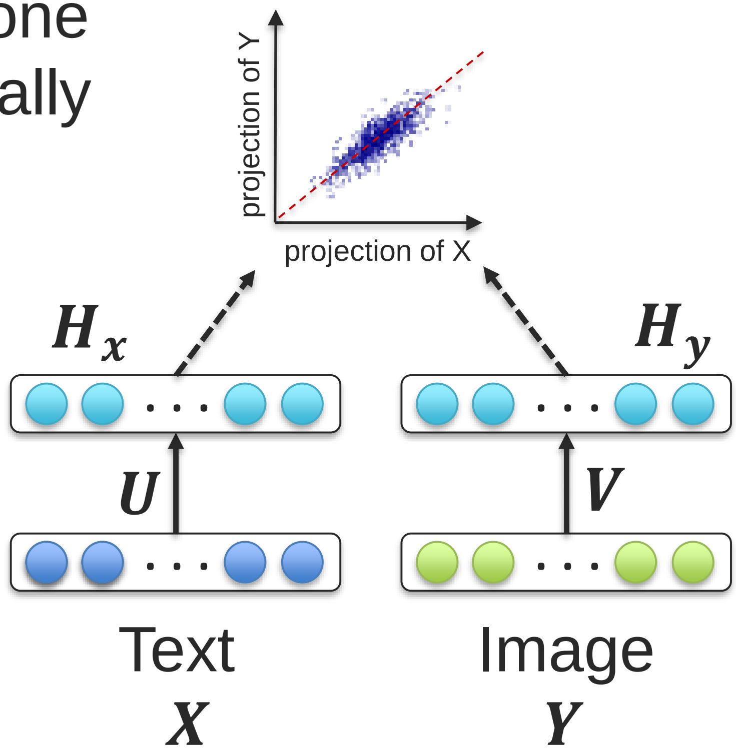
Canonical Correlation Analysis

"canonical": reduced to the simplest or clearest schema possible

1 Learn two linear projections, one for each view, that are maximally correlated:

$$(\boldsymbol{u}^*, \boldsymbol{v}^*) = \underset{\boldsymbol{u},\boldsymbol{v}}{\operatorname{argmax}}\, corr(\boldsymbol{H_x}, \boldsymbol{H_y})$$

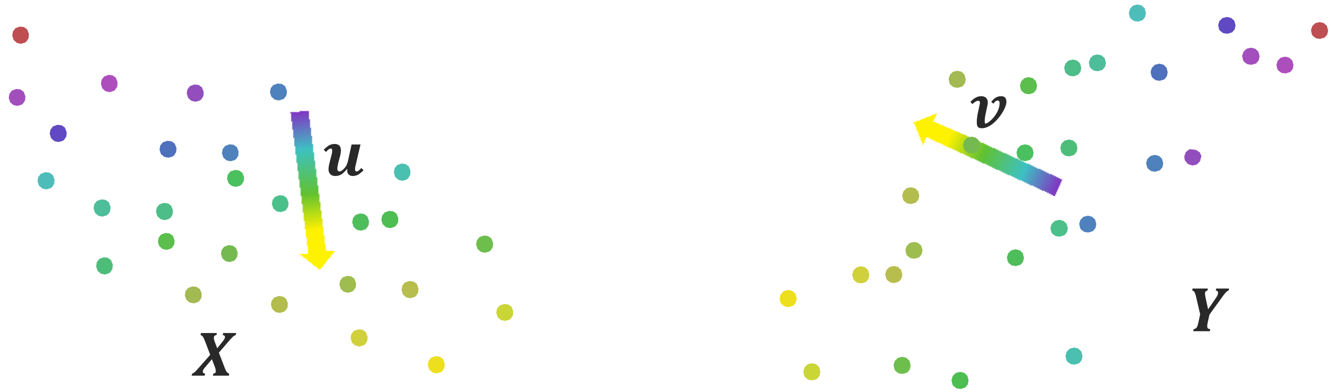
$$= \underset{\boldsymbol{u},\boldsymbol{v}}{\operatorname{argmax}}\, corr(\boldsymbol{u}^T \boldsymbol{X}, \boldsymbol{v}^T \boldsymbol{Y})$$



Correlated Projection

1 Learn two linear projections, one for each view, that are maximally correlated:

$$(\boldsymbol{u}^*, \boldsymbol{v}^*) = \underset{\boldsymbol{u},\boldsymbol{v}}{\operatorname{argmax}}\, corr\left(\boldsymbol{u}^T\boldsymbol{X}, \boldsymbol{v}^T\boldsymbol{Y}\right)$$



Two views $\boldsymbol{X}, \boldsymbol{Y}$ where same instances have the same color

Canonical Correlation Analysis

We want to learn multiple projection pairs $(\boldsymbol{u}_{(i)}\boldsymbol{X}, \boldsymbol{v}_{(i)}\boldsymbol{Y})$:

$$(\boldsymbol{u}_{(i)}^*, \boldsymbol{v}_{(i)}^*) = \underset{\boldsymbol{u}_{(i)}, \boldsymbol{v}_{(i)}}{\operatorname{argmax}}\, corr(\boldsymbol{u}_{(i)}^T\boldsymbol{X}, \boldsymbol{v}_{(i)}^T\boldsymbol{Y}) \qquad \approx \boldsymbol{u}_{(i)}^T\boldsymbol{\Sigma}_{XY}\boldsymbol{v}_{(i)}$$

2 We want these multiple projection pairs to be orthogonal ("canonical") to each other:

$$\boldsymbol{u}_{(i)}^T\boldsymbol{\Sigma}_{XY}\boldsymbol{v}_{(j)} = \boldsymbol{u}_{(j)}^T\boldsymbol{\Sigma}_{XY}\boldsymbol{v}_{(i)} = \boldsymbol{0} \qquad \text{for } i \neq j$$

$$\boldsymbol{U}\boldsymbol{\Sigma}_{XY}\boldsymbol{V} = tr(\boldsymbol{U}\boldsymbol{\Sigma}_{XY}\boldsymbol{V}) \qquad \text{where } \boldsymbol{U} = [\boldsymbol{u}_{(1)}, \boldsymbol{u}_{(2)}, \dots, \boldsymbol{u}_{(k)}]$$

$$\text{and } \boldsymbol{V} = [\boldsymbol{v}_{(1)}, \boldsymbol{v}_{(2)}, \dots, \boldsymbol{v}_{(k)}]$$

Canonical Correlation Analysis

3 Since this objective function is invariant to scaling, we can constraint the projections to have unit variance:

$$\boldsymbol{U^T \Sigma_{XX} U = I} \qquad \boldsymbol{V^T \Sigma_{YY} V = I}$$

Canonical Correlation Analysis:

maximize: $tr(\boldsymbol{U^T \Sigma_{XY} V})$

subject to: $\boldsymbol{U^T \Sigma_{YY} U = V^T \Sigma_{YY} V = I}$

Canonical Correlation Analysis

maximize: $tr(\boldsymbol{U}^T \boldsymbol{\Sigma}_{XY} \boldsymbol{V})$

subject to: $\boldsymbol{U}^T \boldsymbol{\Sigma}_{YY} \boldsymbol{U} = \boldsymbol{V}^T \boldsymbol{\Sigma}_{YY} \boldsymbol{V} = \boldsymbol{I}$

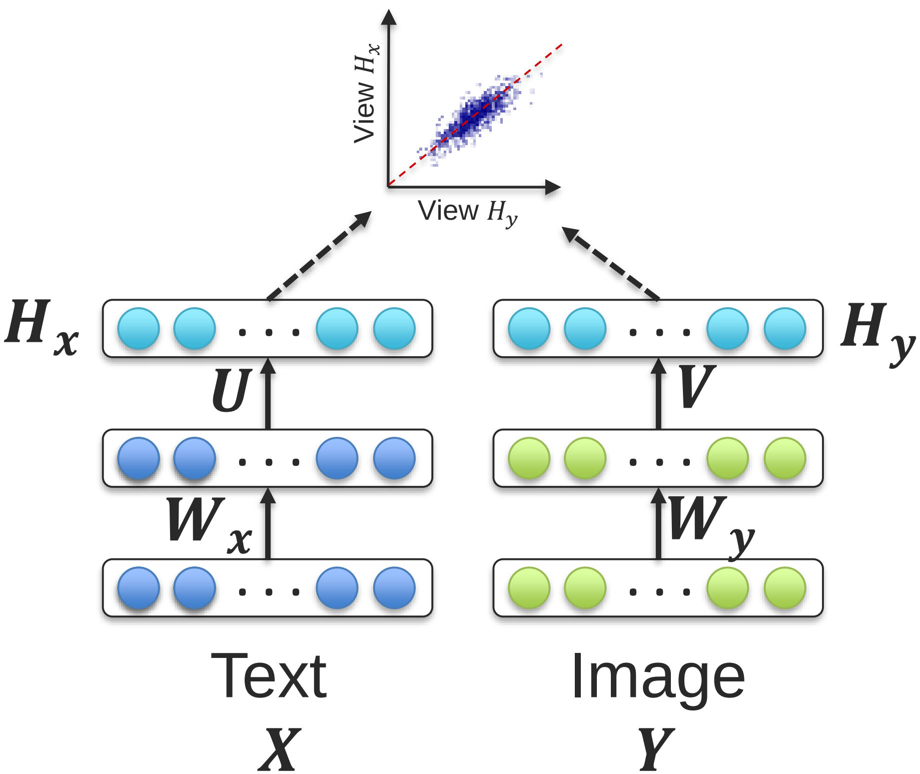
$$\Sigma = \left[\begin{array}{c|c} \boldsymbol{\Sigma}_{XX} & \boldsymbol{\Sigma}_{YX} \\ \hline \boldsymbol{\Sigma}_{XY} & \boldsymbol{\Sigma}_{YY} \end{array}\right] \overset{\boldsymbol{U},\boldsymbol{V}}{\Longrightarrow} \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & \lambda_1 & 0 & 0 \\ 0 & 1 & 0 & 0 & \lambda_2 & 0 \\ 0 & 0 & 1 & 0 & 0 & \lambda_3 \\ \hline \lambda_1 & 0 & 0 & 1 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 & 1 & 0 \\ 0 & 0 & \lambda_3 & 0 & 0 & 1 \end{array}\right]$$

Deep Canonical Correlation Analysis

Same objective function as CCA:

$$\underset{\boldsymbol{V},\boldsymbol{U},\boldsymbol{W}_x,\boldsymbol{W}_y}{\operatorname{argmax}} \; corr(\boldsymbol{H}_x, \boldsymbol{H}_y)$$

1. Linear projections maximizing correlation
2. Orthogonal projections
3. Unit variance of the projection vectors

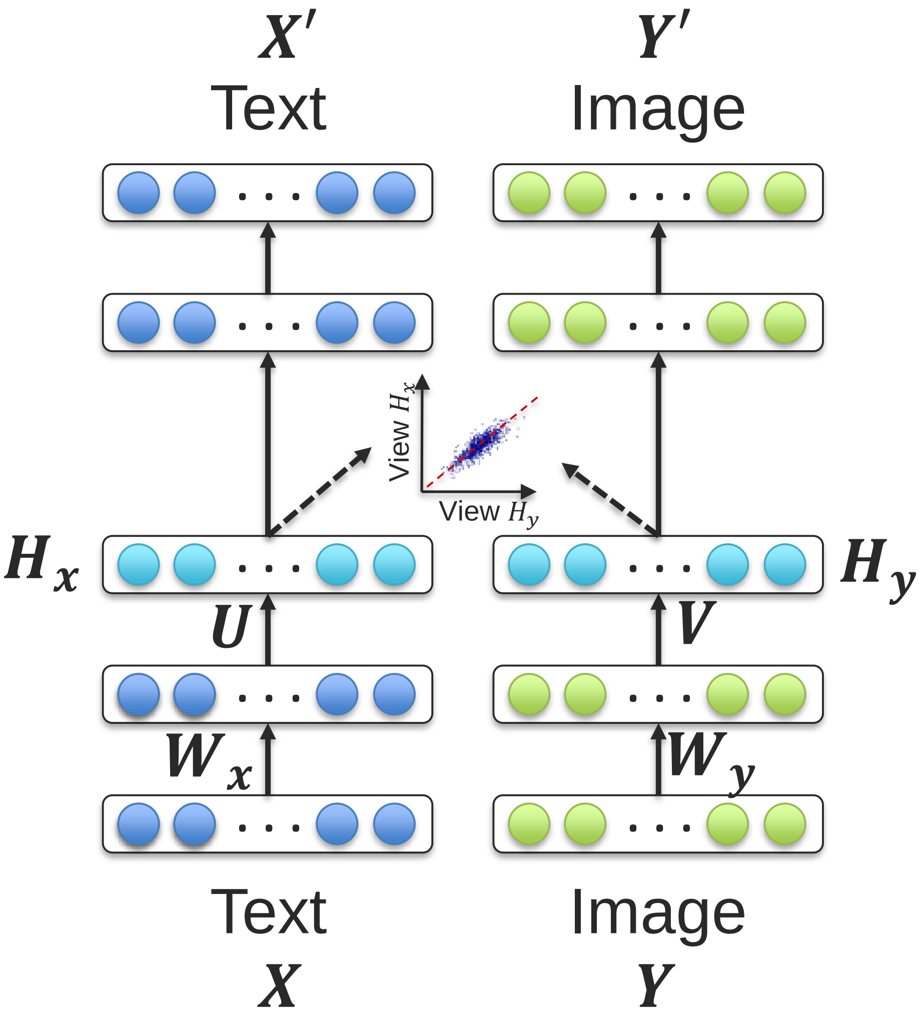


Andrew et al., ICML 2013

Deep Canonically Correlated Autoencoders (DCCAE)

Jointly optimize for DCCA and autoencoders loss functions

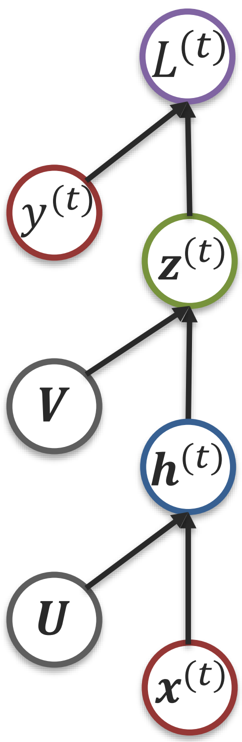
- A trade-off between multi-view correlation and reconstruction error from individual views



Wang et al., ICML 2015

Basic Concepts: Recurrent Neural Networks

Feedforward Neural Network



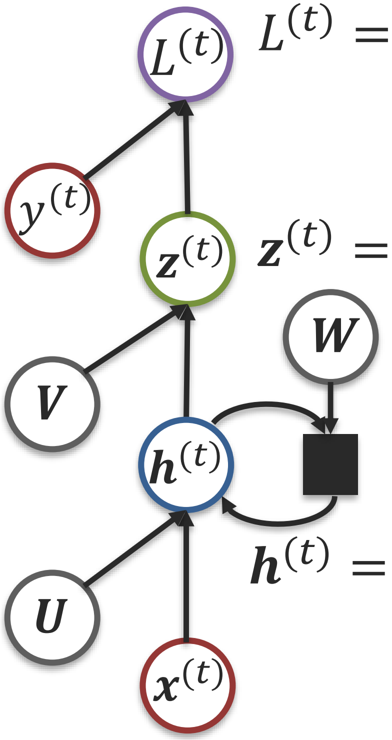
$$L^{(t)} = -logP(Y = y^{(t)} | \boldsymbol{z}^{(t)})$$

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

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

Recurrent Neural Networks

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



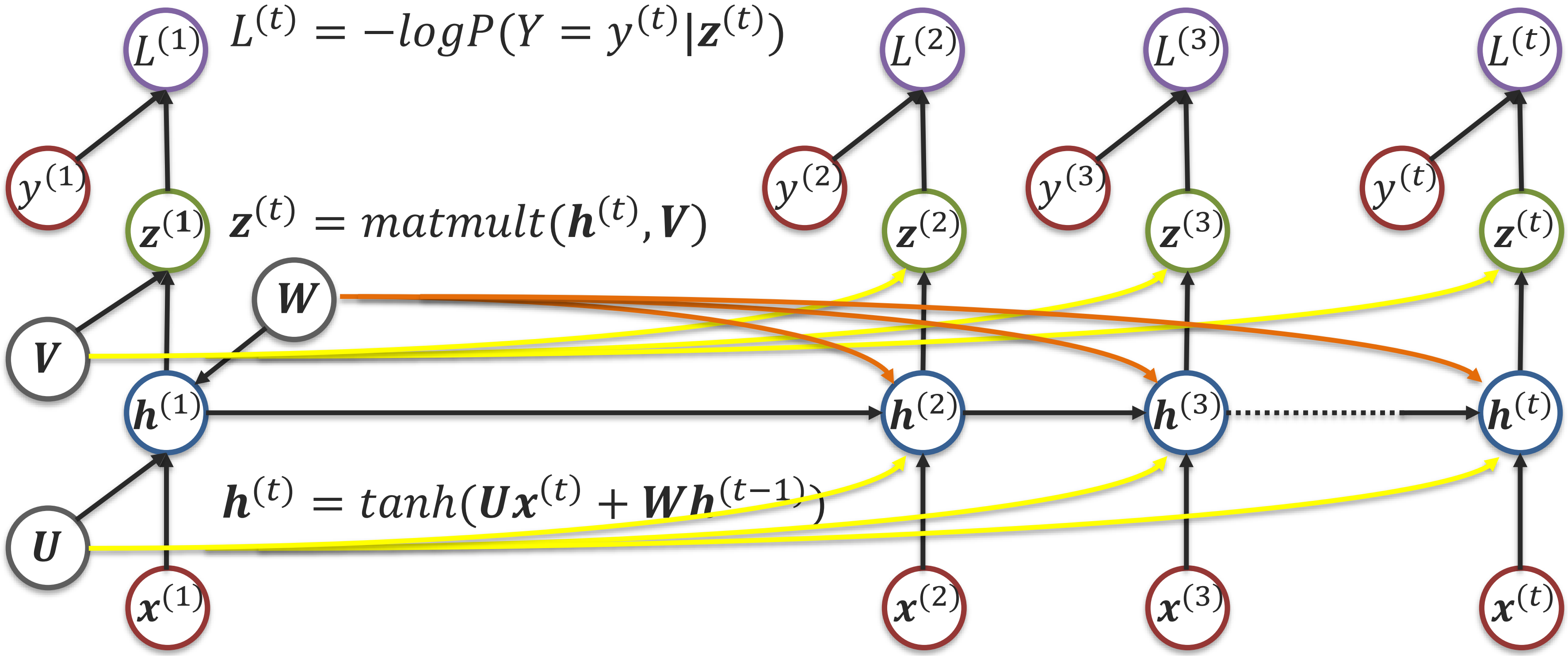
$$L^{(t)} = -logP(Y = y^{(t)} | \boldsymbol{z}^{(t)})$$

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

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

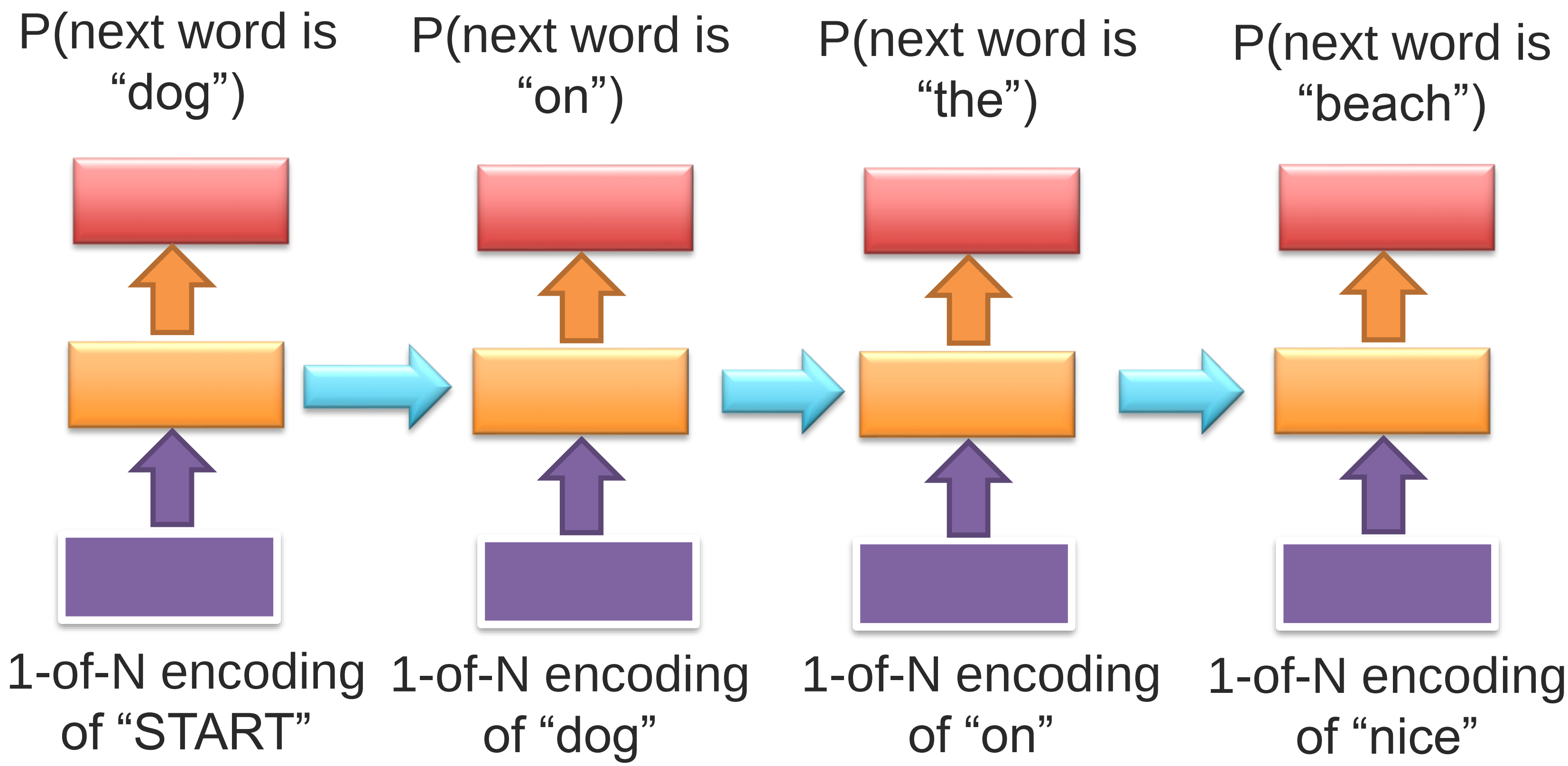
Recurrent Neural Networks - Unrolling

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



Same model parameters are used for all time parts.

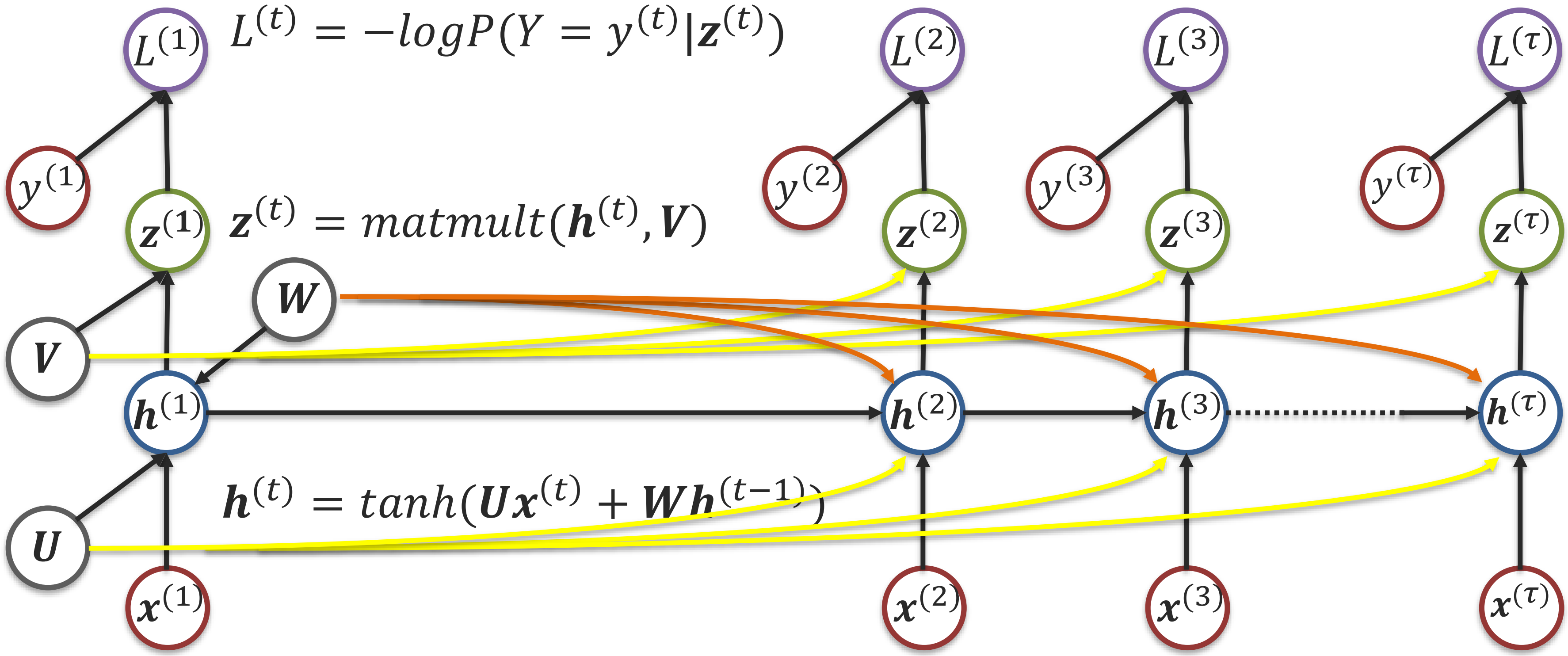
Recurrent Neural Networks – Language models



- Model long-term information

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 $L^{(t)}$

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

$\boldsymbol{z}^{(\tau)}$ or $\boldsymbol{z}^{(t)}$

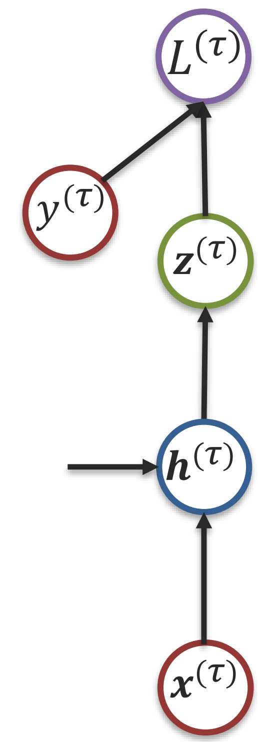
$$\left(\nabla_{\boldsymbol{z}^{(t)}} L\right)_i = \frac{\partial L}{\partial z_i^{(t)}} = \frac{\partial L}{\partial L^{(t)}} \frac{\partial L^{(t)}}{\partial z_i^{(t)}} = sigmoid\left(z_i^t\right) - \mathbf{1}_{i,y^{(t)}}$$

$\boldsymbol{h}^{(\tau)}$

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

$\boldsymbol{h}^{(t)} \rightarrow \boldsymbol{h}^{(t+1)}$

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



Backpropagation Through Time

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

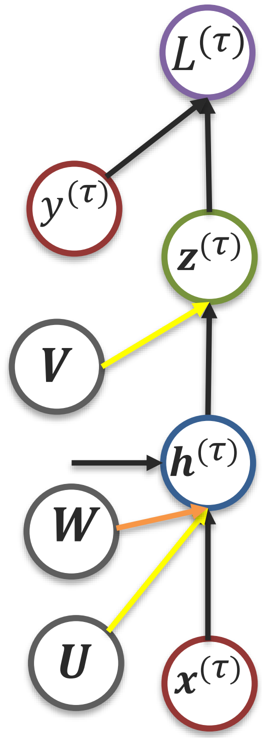
Gradient = "backprop" gradient

x "local" Jacobian

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

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

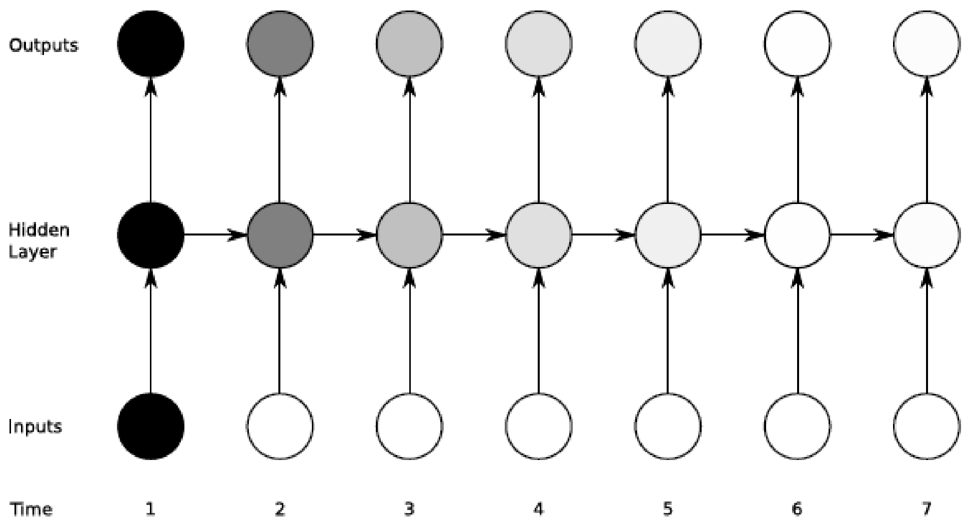
$\boldsymbol{U}$ $$\nabla_{\boldsymbol{U}} L = \sum_t \left(\nabla_{\boldsymbol{h}^{(t)}} L\right) \frac{\partial \boldsymbol{h}^{(t)}}{\partial \boldsymbol{U}}$$



Long-term Dependencies

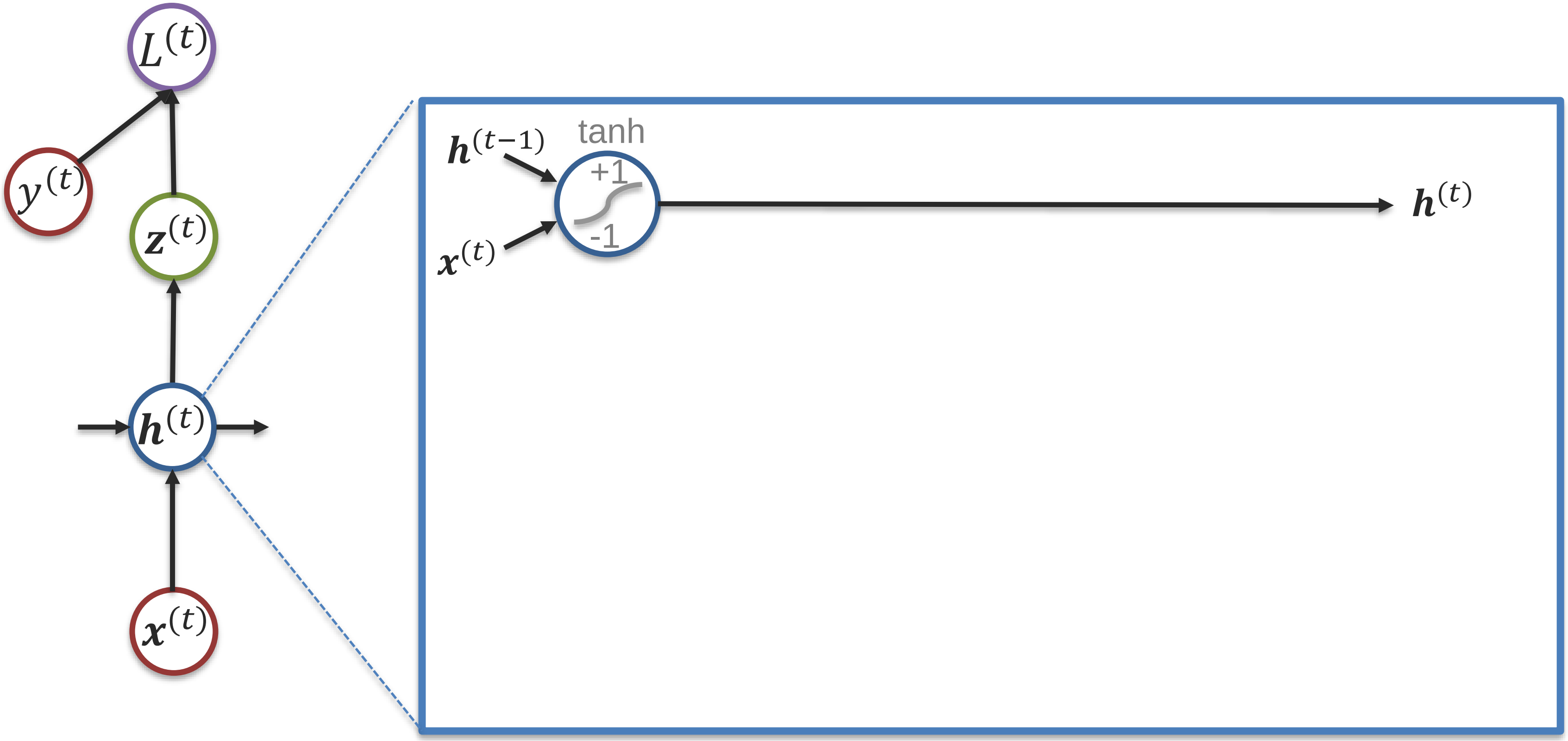
Vanishing gradient problem for RNNs:

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



- 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.

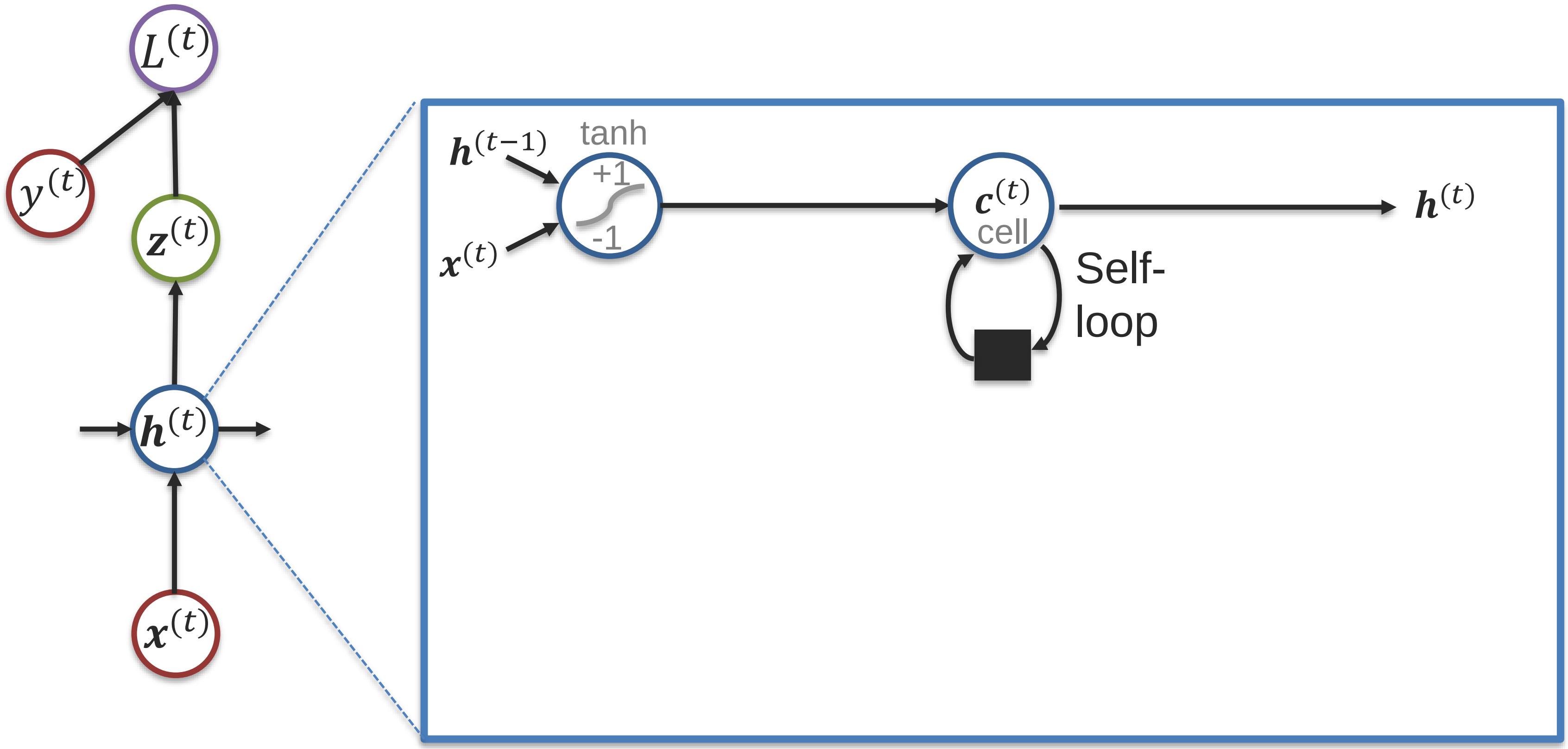
Recurrent Neural Networks



LSTM ideas: (1) “Memory” Cell and Self Loop

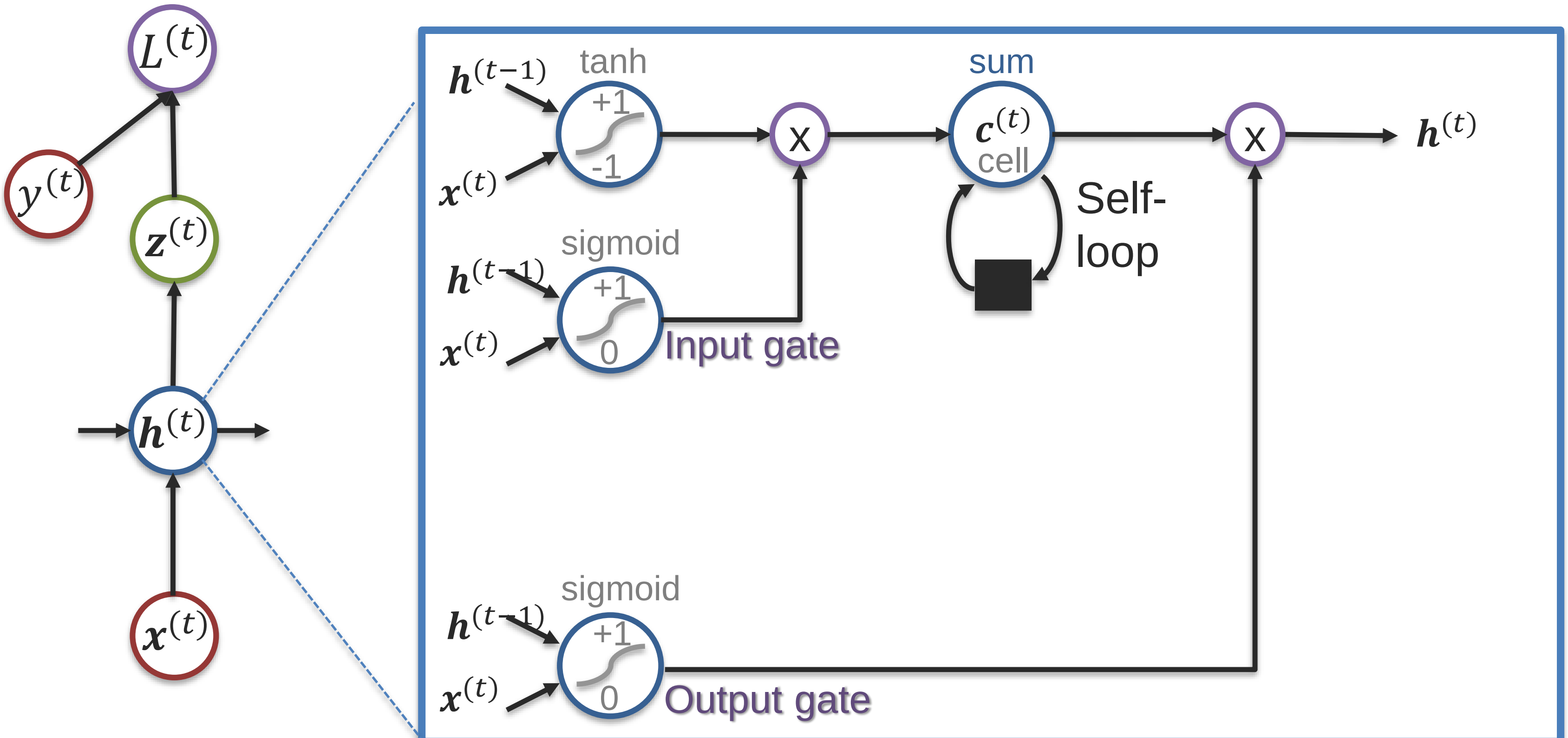
[Hochreiter and Schmidhuber, 1997]

Long Short-Term Memory (LSTM)



LSTM Ideas: (2) Input and Output Gates

[Hochreiter and Schmidhuber, 1997]

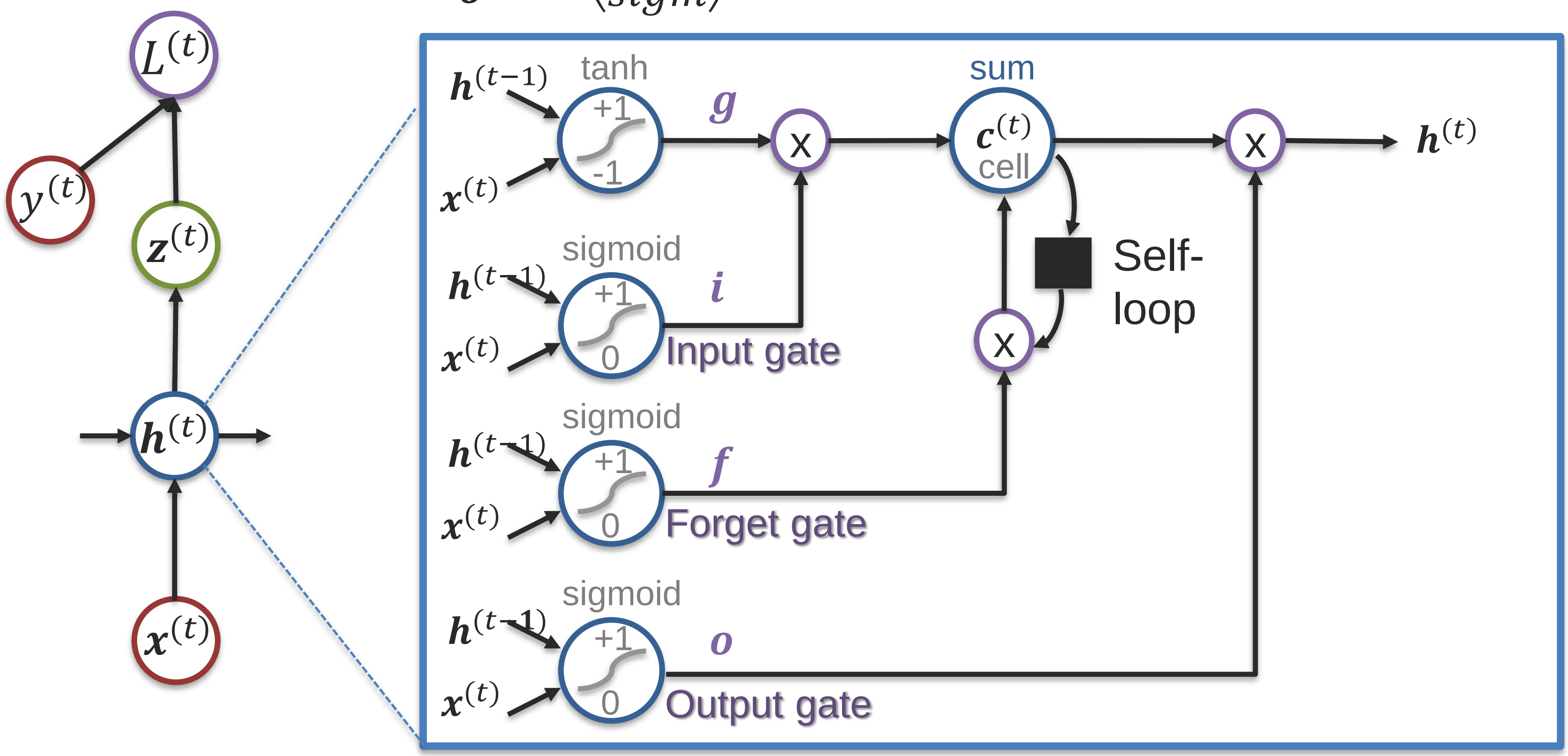


LSTM Ideas: (3) Forget Gate [Gers et al., 2000]

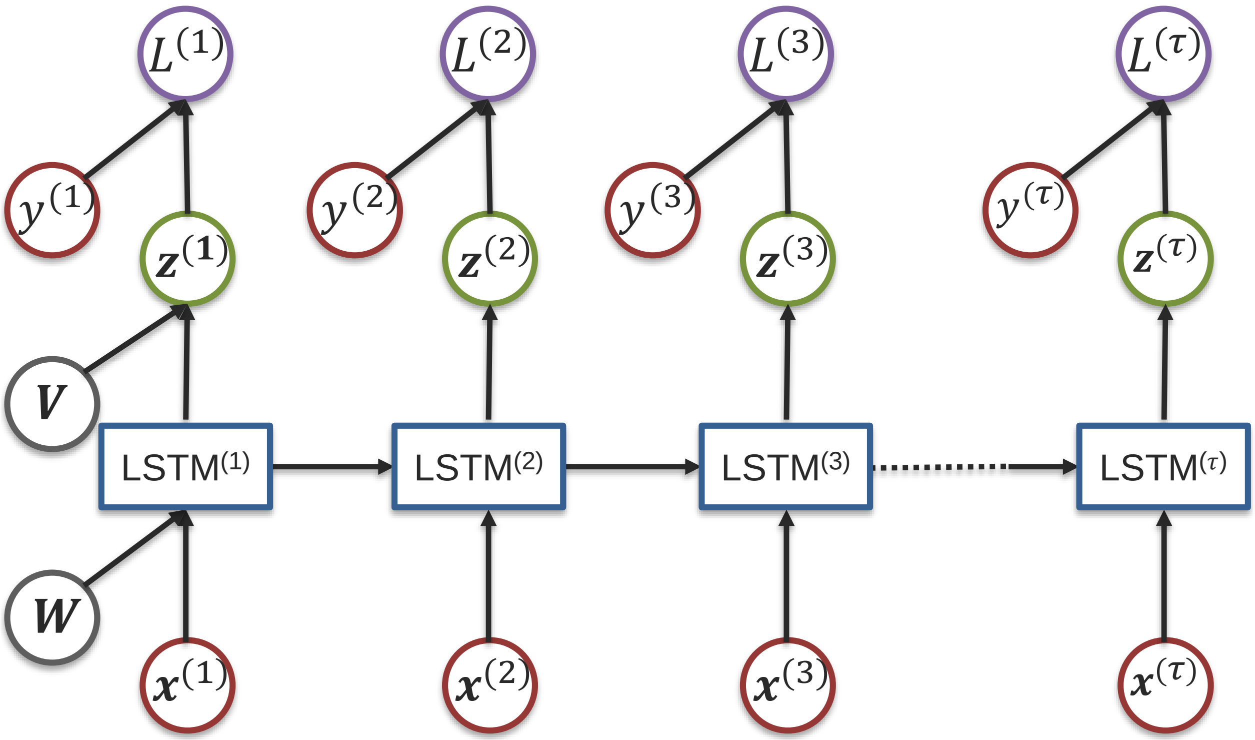
$$\begin{pmatrix} \boldsymbol{g} \\ \boldsymbol{i} \\ \boldsymbol{f} \\ \boldsymbol{o} \end{pmatrix} = \begin{pmatrix} tanh \\ sigm \\ sigm \\ sigm \end{pmatrix} \boldsymbol{W} \begin{pmatrix} \boldsymbol{h}^{(t-1)} \\ \boldsymbol{x}^{(t)} \end{pmatrix}$$

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

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

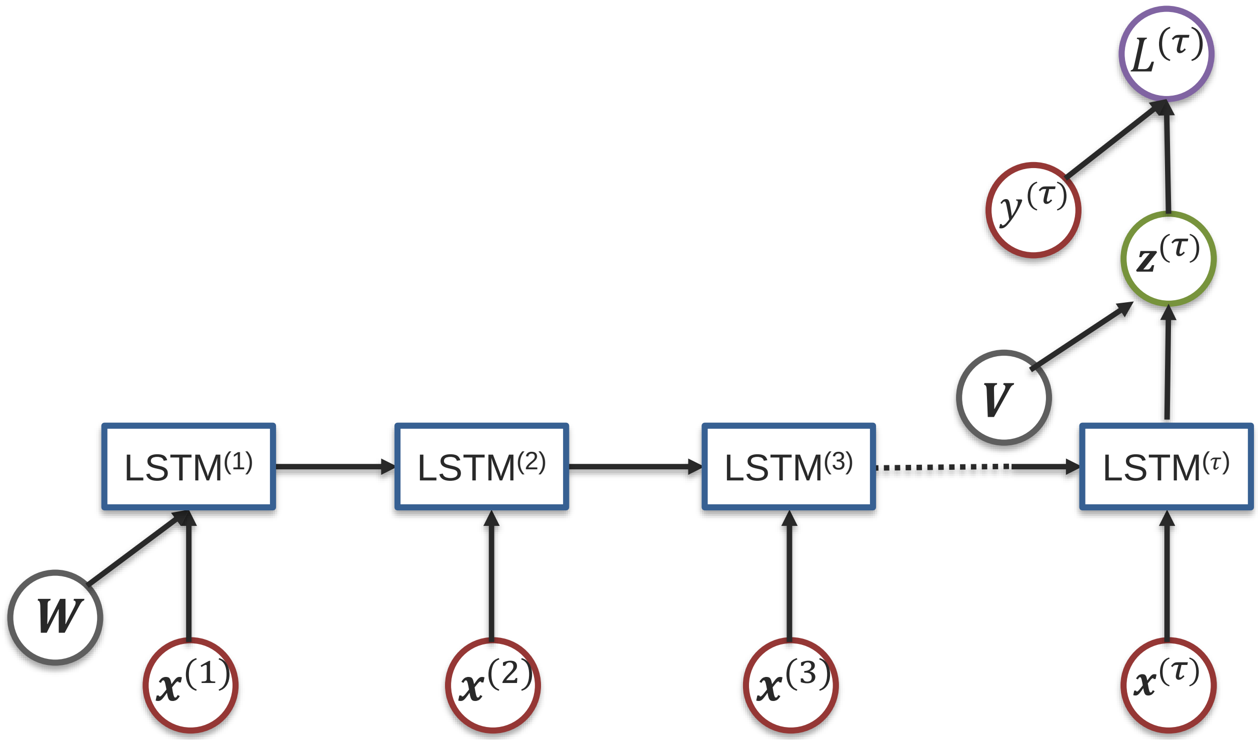


Recurrent Neural Network using LSTM Units



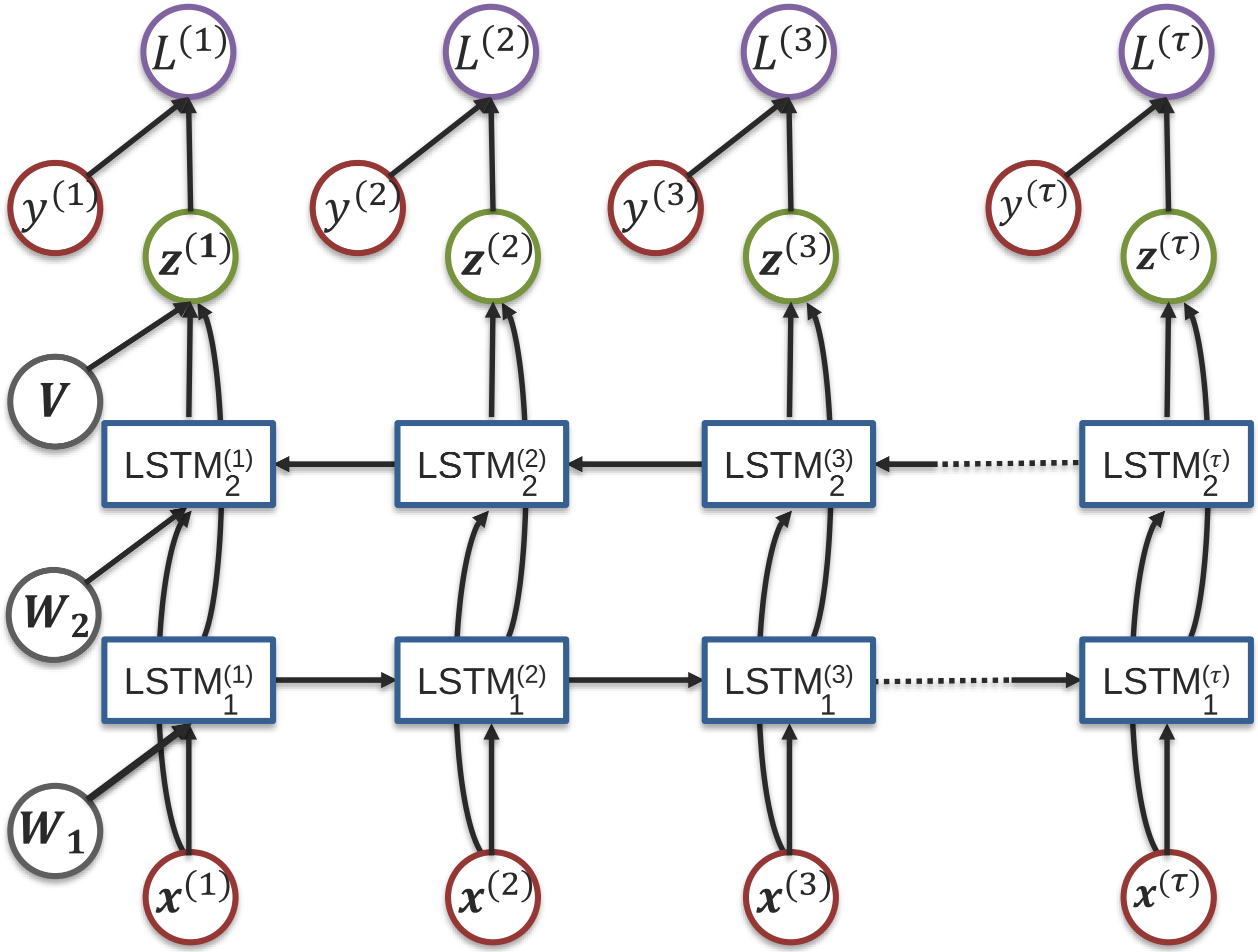
Gradient can still be computer using backpropagation!

Recurrent Neural Network using LSTM Units

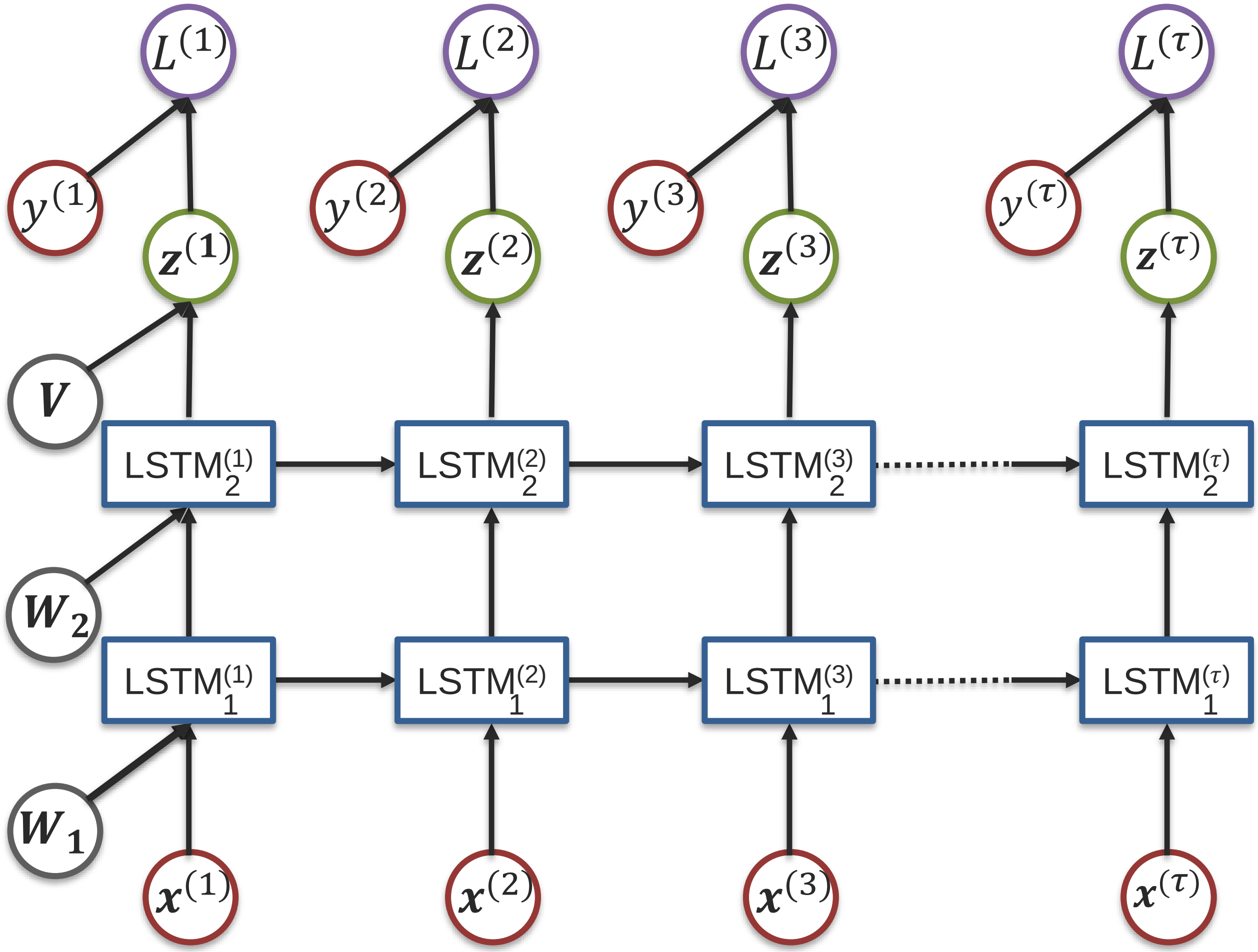


Gradient can still be computer using backpropagation!

Bi-directional LSTM Network



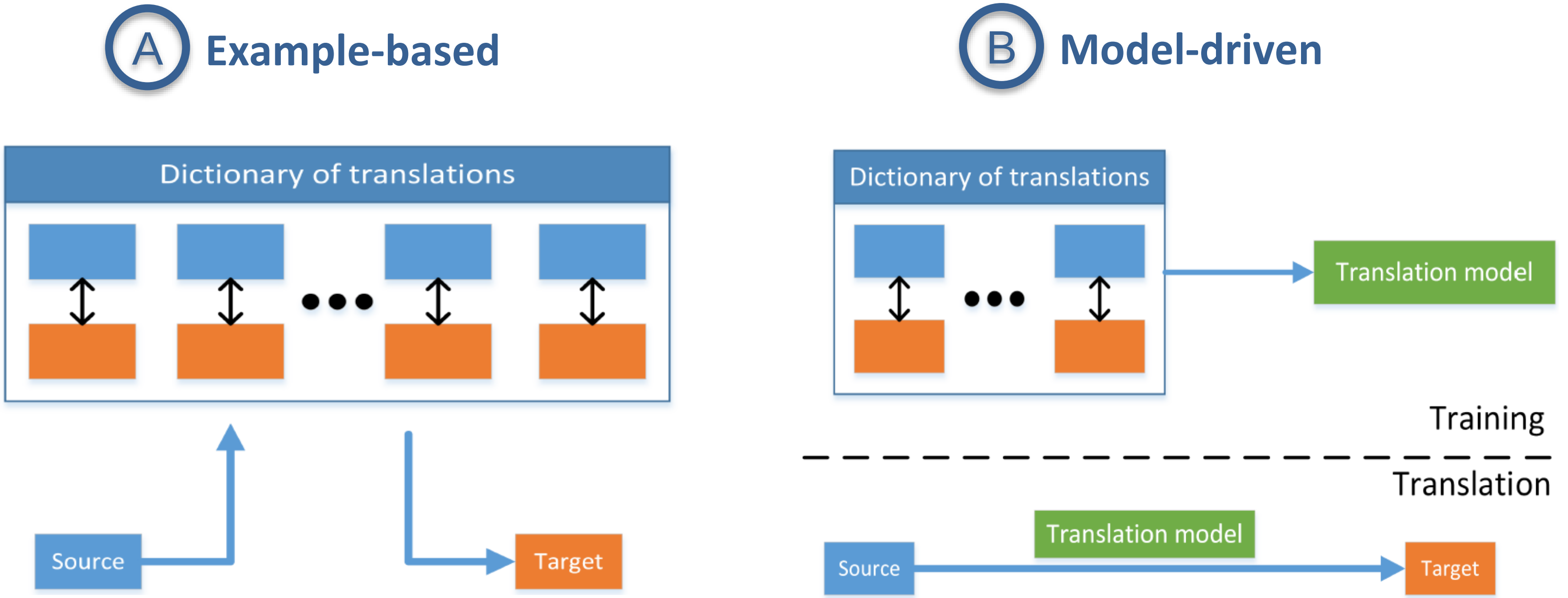
Deep LSTM Network



Translation and Alignment

Core Challenge 4: Translation

Definition: Process of changing data from one modality to another, where the translation relationship can often be open-ended or subjective.

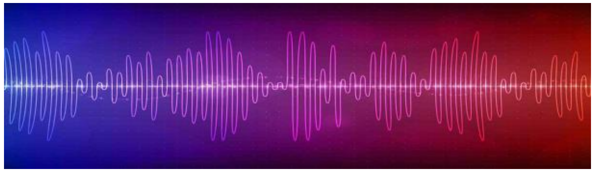


Translation

- **Visual animations**

- **Image captioning**

- **Speech synthesis**



Challenges:

I. Different representations
II. Multiple source modalities
III. Open ended translations
IV. Subjective evaluation
V. Repetitive processes

Example-based translation

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

.'Iniesta is really impressing me.' said Zinedine

Nods of approval could be seen across the continent: Andres Iniesta was named the best player of Euro 2012. In six Spain games in Poland and Ukraine, Iniesta did not score once but appreciation for the 28-year-old extends well beyond goals, it is now as broad as Europe. Iniesta has not quite gained the inevitability of gravity but the reliability of his talent is unquestionable

. . .

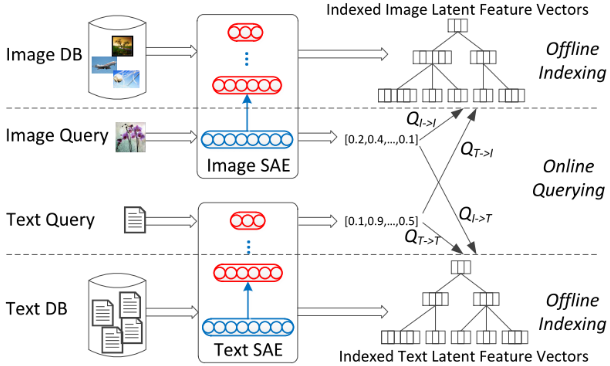
Kobe Bryant said, "To be really frank with you, I really do not look at it as that, for the simple fact that Michael Jordan has really taught me a lot. Really taught me a lot. The trainer of his, Tim Grover, he's passed on to me and I work with him a great deal, and he's shown me a lot. So I can't sit there and say, well, I'm trying to catch Michael Jordan at six, I want to pass him after six.

. . .

[Wei et al. 2015]

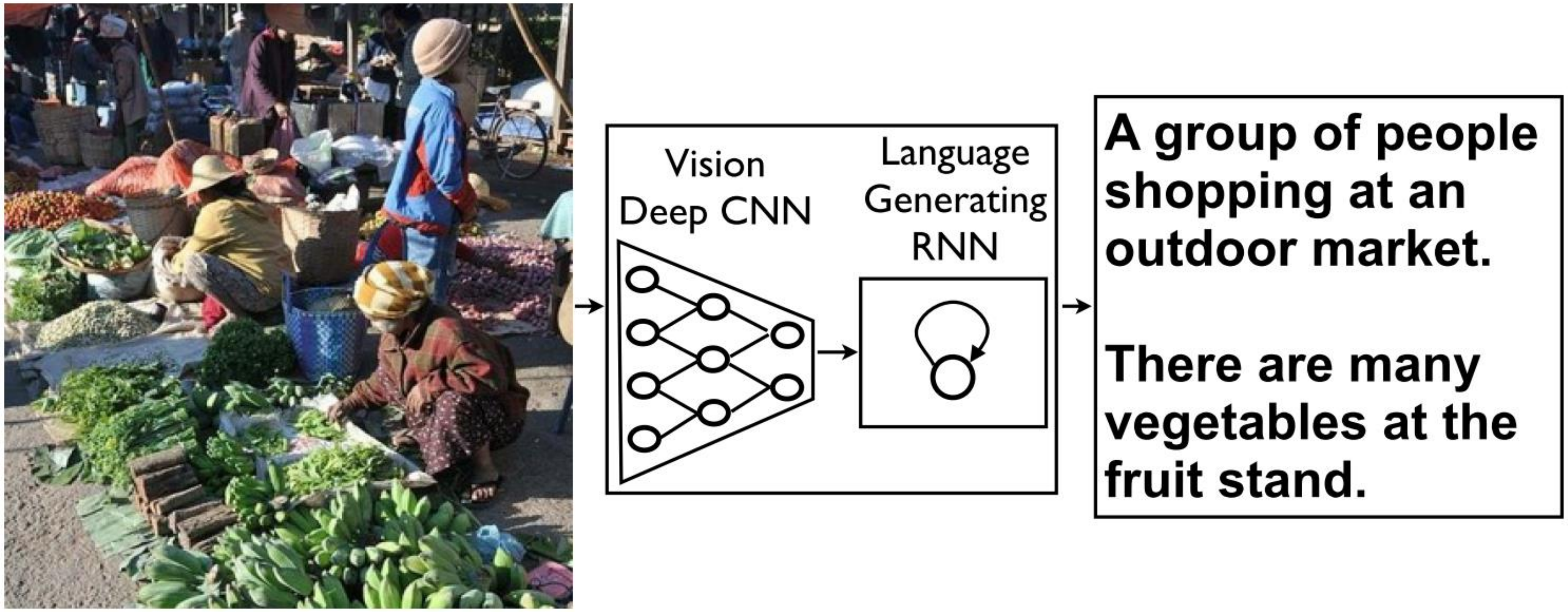
Example-based translation

- 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]

Model-based Image captioning with Encoder-Decoder



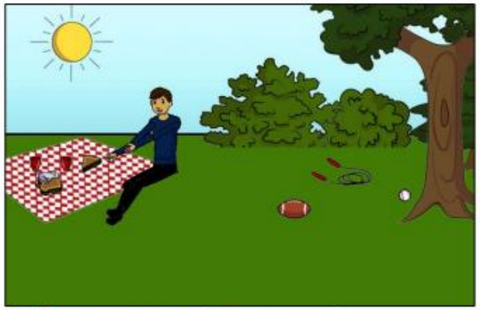
[Vinyals et al., “Show and Tell: A Neural Image Caption Generator”, CVPR 2015]

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?

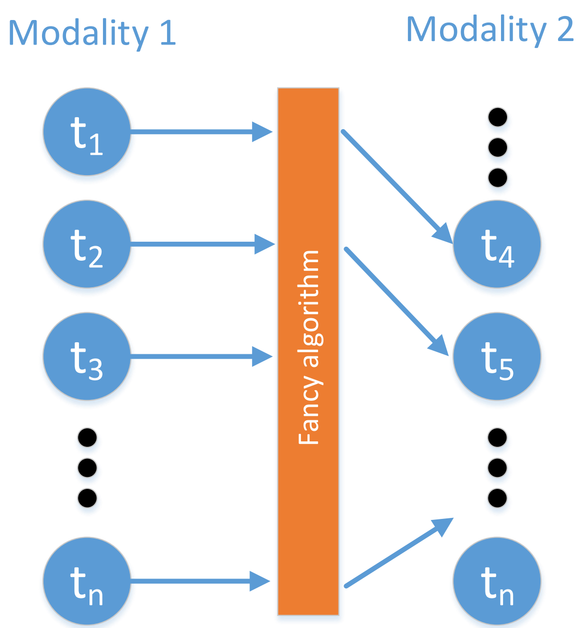
Does it appear to be rainy?
Does this person have 20/20 vision?

Evaluation on “Unbounded” Translations

- 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

Core Challenge: Alignment

Definition: Identify the direct relations between (sub)elements from two or more different modalities.



A Explicit Alignment

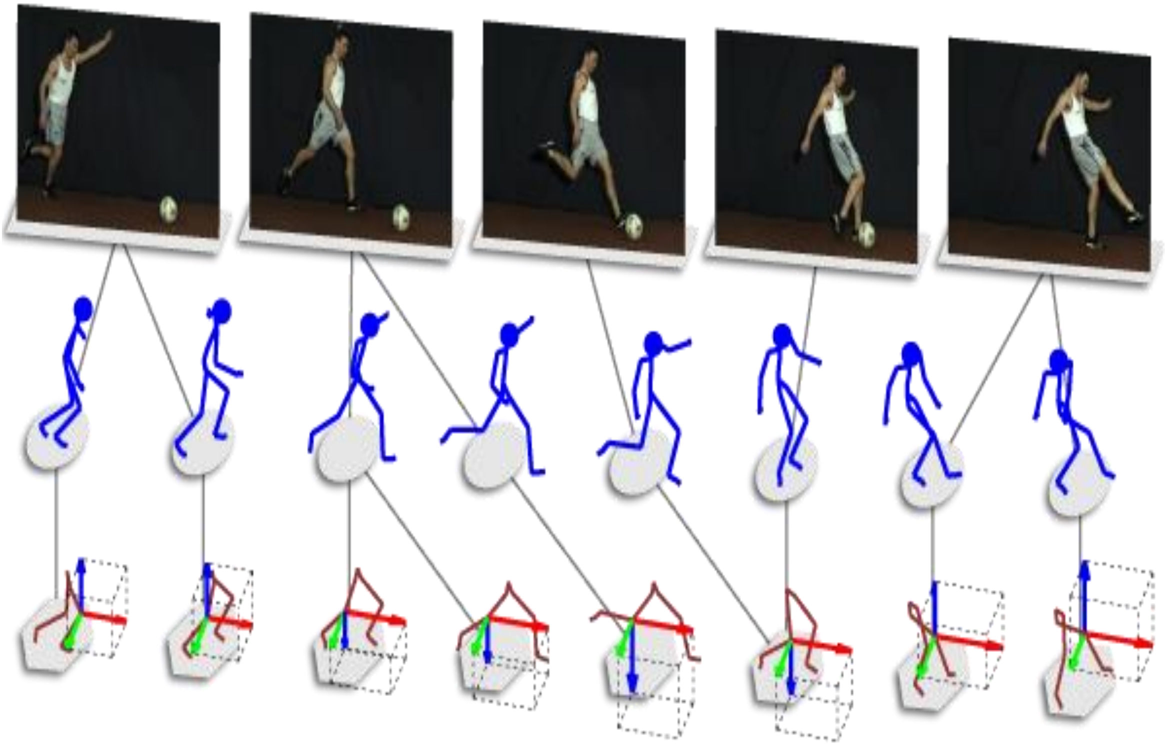
The goal is to directly find correspondences between elements of different modalities

B Implicit Alignment

Uses internally latent alignment of modalities in order to better solve a different problem

Explicit alignment

Temporal sequence alignment



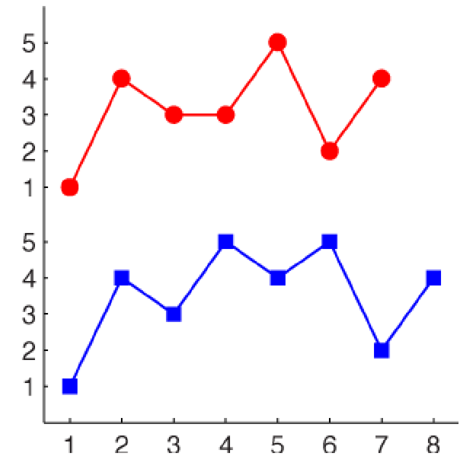
Applications:
- Re-aligning asynchronous data
- Finding similar data across modalities (we can estimate the aligned cost)
- Event reconstruction from multiple sources

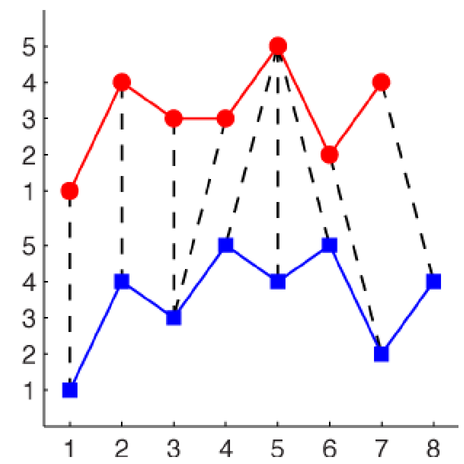
Let's start unimodal – Dynamic Time Warping

- We have two unaligned temporal unimodal signals
  - $\mathbf{X} = \left[\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{n_x}\right] \in \mathbb{R}^{d \times n_x}$
  - $\mathbf{Y} = \left[\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_{n_y}\right] \in \mathbb{R}^{d \times n_y}$
- Find set of indices to minimize the alignment difference:

$$L(\boldsymbol{p}_t^x, \boldsymbol{p}_t^y) = \sum_{t=1}^{l} \left\| \boldsymbol{x}_{\boldsymbol{p}_t^x} - \boldsymbol{y}_{\boldsymbol{p}_t^y} \right\|_2^2$$

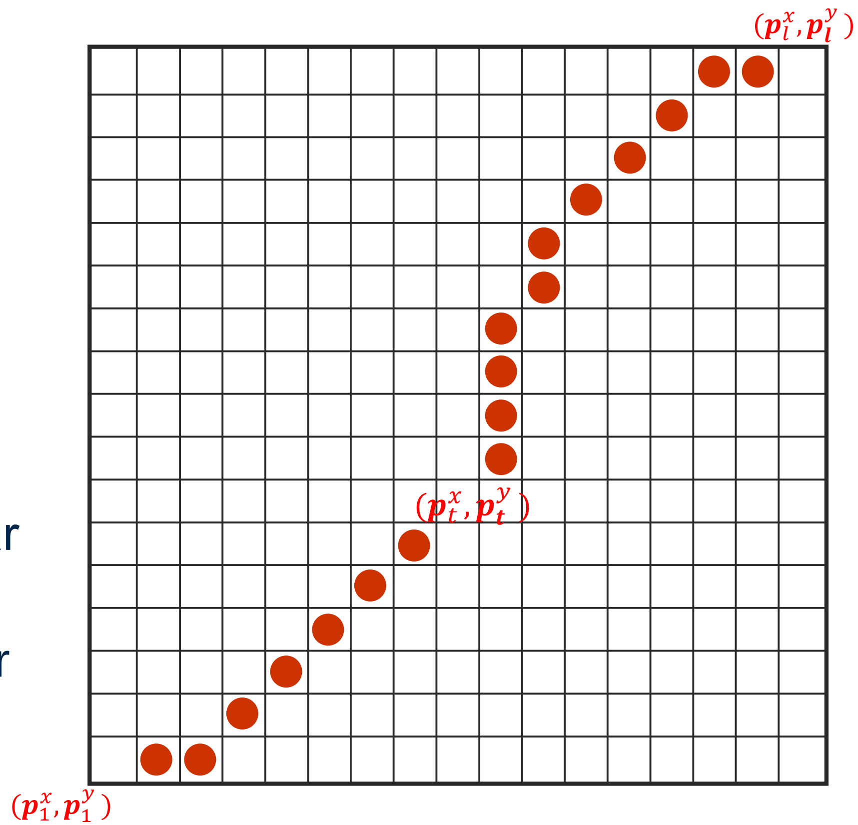
- Where $\boldsymbol{p}_t^x$ and $\boldsymbol{p}_t^y$ are index vectors of same length
- Finding these indices is called Dynamic Time Warping





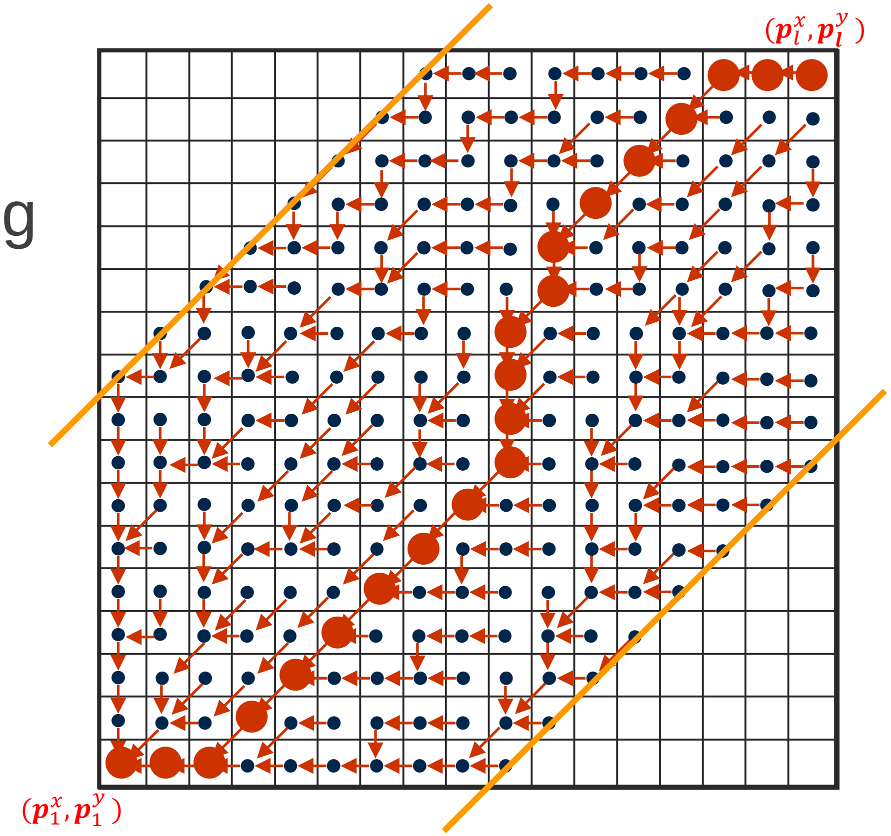
Dynamic Time Warping continued

- Lowest cost path in a cost matrix
- Restrictions
  - Monotonicity – no going back in time
  - Continuity - no gaps
  - Boundary conditions - start and end at the same points
  - Warping window - don't get too far from diagonal
  - Slope constraint – do not insert or skip too much



Dynamic Time Warping continued

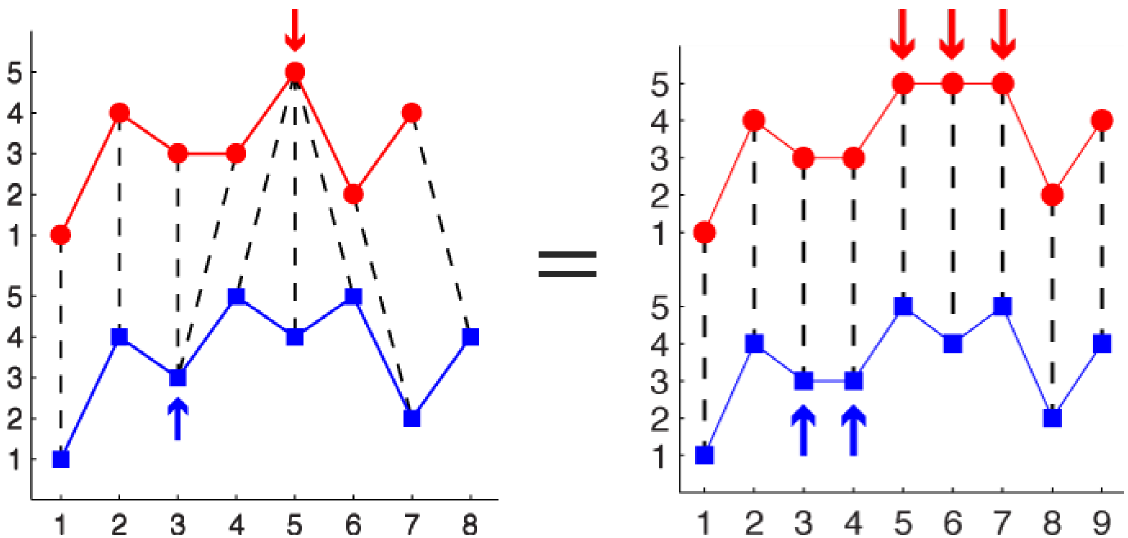
- Lowest cost path in a cost matrix
- Solved using dynamic programming whilst respecting the restrictions



DTW alternative formulation

$$L(\boldsymbol{p}_t^x, \boldsymbol{p}_t^y) = \sum_{t=1}^{l} \left\| \boldsymbol{x}_{p_t^x} - \boldsymbol{y}_{p_t^y} \right\|_2^2$$

Replication doesn't change the objective!



$$= \mathbf{X}\;\; \begin{bmatrix} 1&0&0&0&0&0&0&0&0\\ 0&1&0&0&0&0&0&0&0\\ 0&0&1&0&0&0&0&0&0\\ 0&0&0&1&0&0&0&0&0\\ 0&0&0&0&1&1&1&0&0\\ 0&0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&0&1 \end{bmatrix}$$

$$= \mathbf{Y}\;\; \begin{bmatrix} 1&0&0&0&0&0&0&0&0\\ 0&1&0&0&0&0&0&0&0\\ 0&0&1&1&0&0&0&0&0\\ 0&0&0&0&1&0&0&0&0\\ 0&0&0&0&0&1&0&0&0\\ 0&0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&0&1 \end{bmatrix}$$

Alternative objective:

$$L(\boldsymbol{W}_x, \boldsymbol{W}_y) = \left\| \boldsymbol{X}\boldsymbol{W}_x - \boldsymbol{Y}\boldsymbol{W}_y \right\|_F^2$$

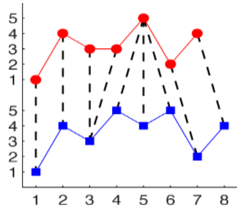
Frobenius norm $\|\boldsymbol{A}\|_F^2 = \sum_i \sum_j \left|a_{i,j}\right|^2$

$\boldsymbol{X}, \boldsymbol{Y}$ – original signals (same #rows, possibly different #columns)

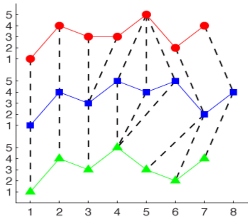
$\boldsymbol{W}_x, \boldsymbol{W}_y$ - alignment matrices

DTW - limitations

- Computationally complex



$O(n_x n_y)$



$O(n_x n_y n_z)$

m sequences

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

- Sensitive to outliers

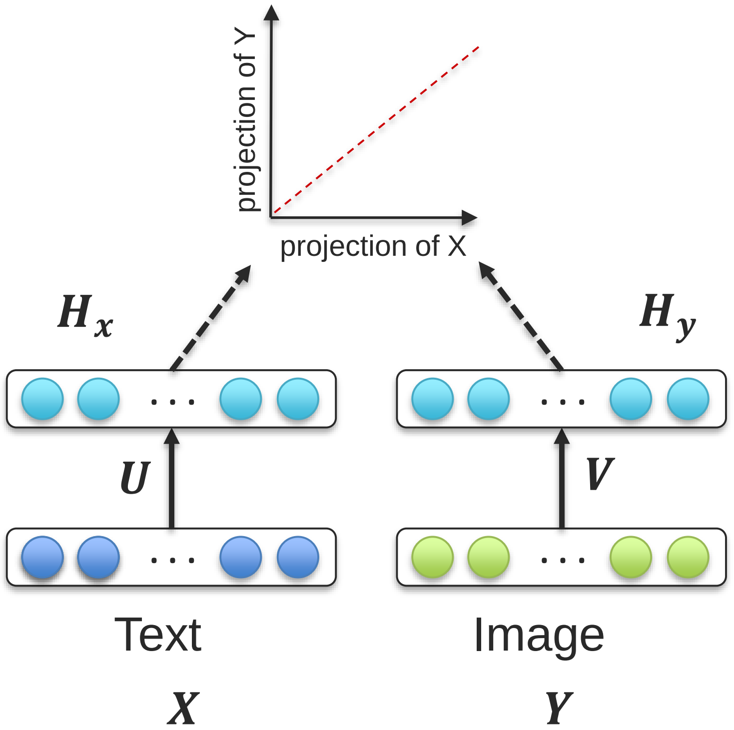
- Unimodal!

Canonical Correlation Analysis reminder

maximize: $tr(\boldsymbol{U}^T \boldsymbol{\Sigma}_{XY} \boldsymbol{V})$

subject to: $\boldsymbol{U}^T \boldsymbol{\Sigma}_{YY} \boldsymbol{U} = \boldsymbol{V}^T \boldsymbol{\Sigma}_{YY} \boldsymbol{V} = \boldsymbol{I}$

1. Linear projections maximizing correlation
2. Orthogonal projections
3. Unit variance of the projection vectors

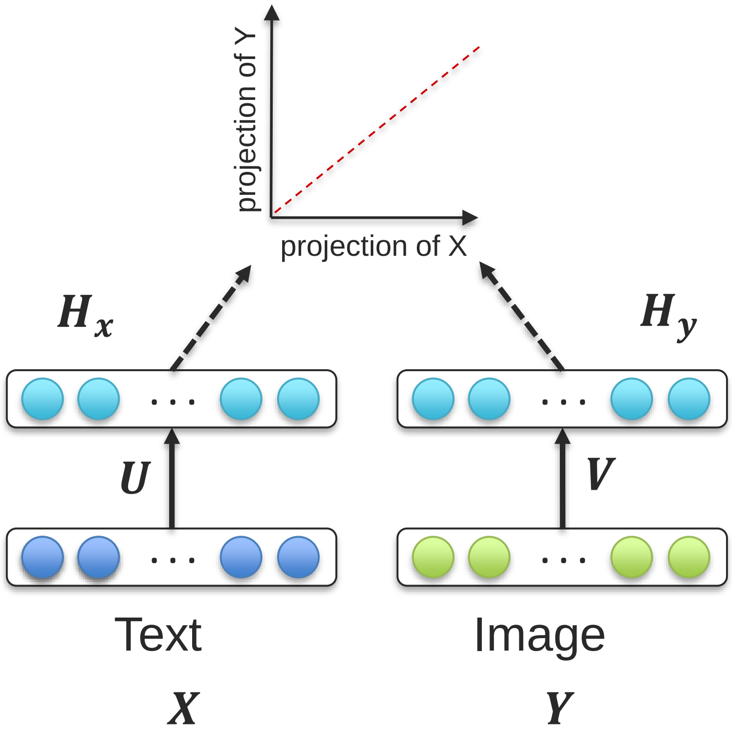


Canonical Correlation Analysis reminder

- When data is normalized it is actually equivalent to smallest RMSE reconstruction
- CCA loss can also be re-written as:

$$L(\boldsymbol{U},\boldsymbol{V}) = \|\mathbf{U}^T\mathbf{X} - \mathbf{V}^T\mathbf{Y}\|_F^2$$

subject to: $\boldsymbol{U}^T\boldsymbol{\Sigma}_{YY}\boldsymbol{U} = \boldsymbol{V}^T\boldsymbol{\Sigma}_{YY}\boldsymbol{V} = \boldsymbol{I}$



Canonical Time Warping

- Dynamic Time Warping + Canonical Correlation Analysis = Canonical Time Warping

$$L(\boldsymbol{U},\boldsymbol{V},\boldsymbol{W}_x,\boldsymbol{W}_y) = \left\|\mathbf{U}^T\mathbf{X}\mathbf{W}_\mathbf{x} - \mathbf{V}^T\mathbf{Y}\mathbf{W}_\mathbf{y}\right\|_F^2$$

- Allows to align multi-modal or multi-view (same modality but from a different point of view)
- $\boldsymbol{W}_x, \boldsymbol{W}_y$ – temporal alignment
- $\boldsymbol{U}, \boldsymbol{V}$ – cross-modal (spatial) alignment

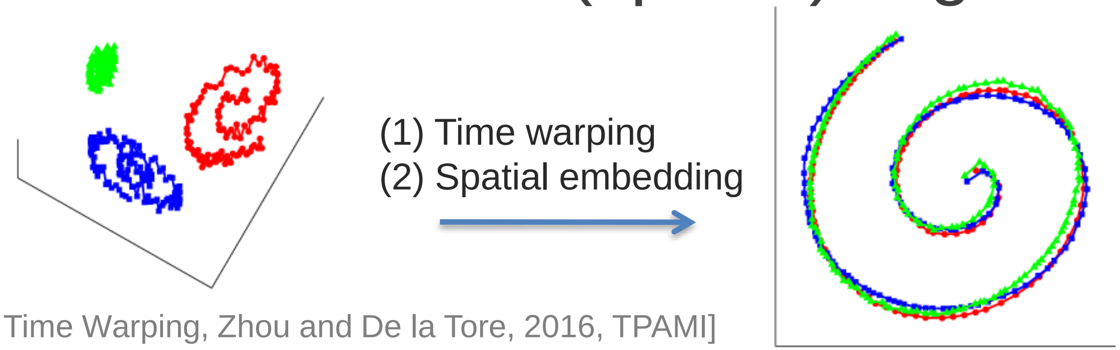
[Canonical Time Warping for Alignment of Human Behavior, Zhou and De la Tore, 2009]

Generalized Time warping

- Generalize to multiple sequences all of different modality

$$L(\boldsymbol{U}_i, \boldsymbol{W}_i) = \sum_{i=1} \sum_{j=1} \left\| \mathbf{U}_i^T \mathbf{X}_\mathrm{i} \mathbf{W}_\mathrm{i} - \mathbf{U}_j^T \mathbf{X}_\mathrm{j} \mathbf{W}_j \right\|_F^2$$

- $\boldsymbol{W}_i$ – set of temporal alignments
- $\boldsymbol{U}_i$ – set of cross-modal (spatial) alignments



[Generalized Canonical Time Warping, Zhou and De la Tore, 2016, TPAMI]

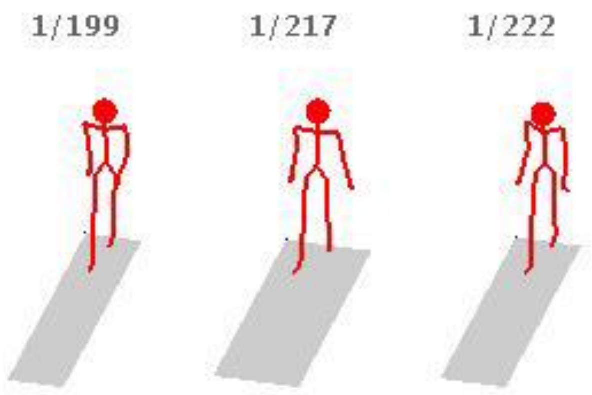
Alignment examples (unimodal)

CMU Motion Capture

Subject 1: 199 frames

Subject 2: 217 frames

Subject 3: 222 frames

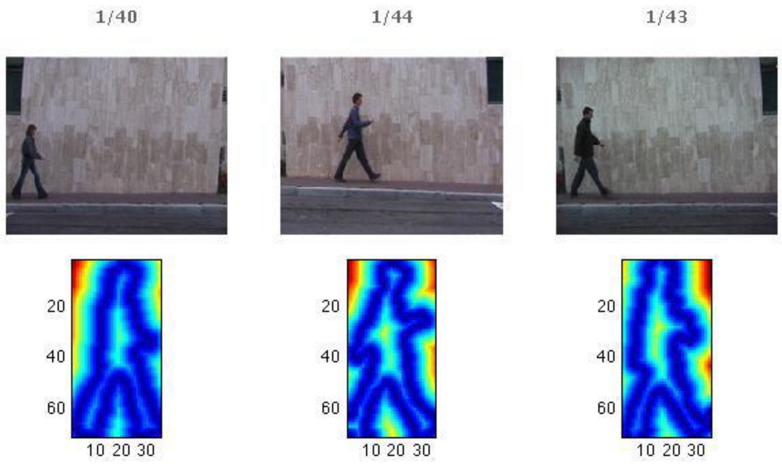


Weizmann

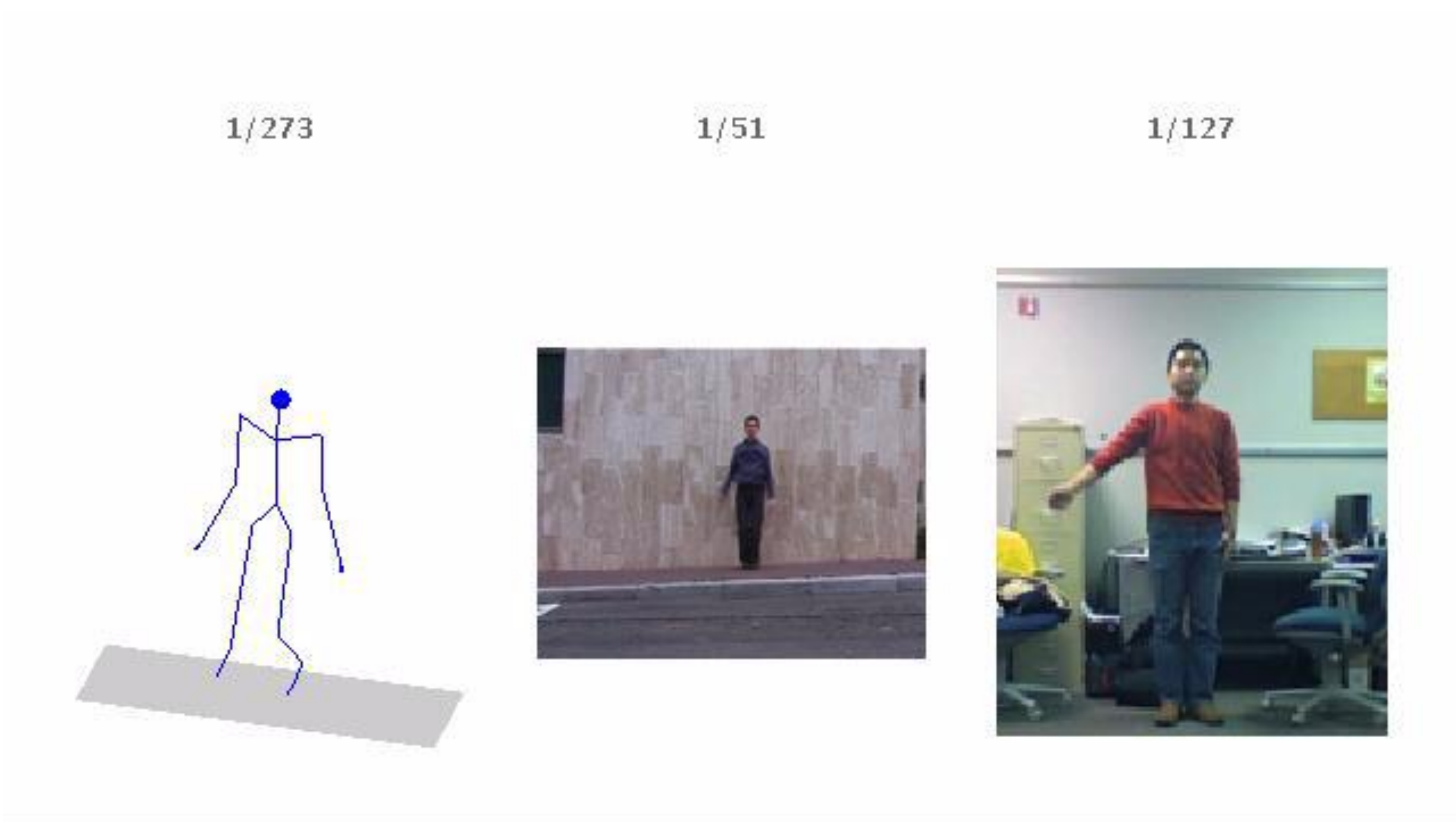
Subject 1: 40 frames

Subject 2: 44 frames

Subject 3: 43 frames



Alignment examples (multimodal)

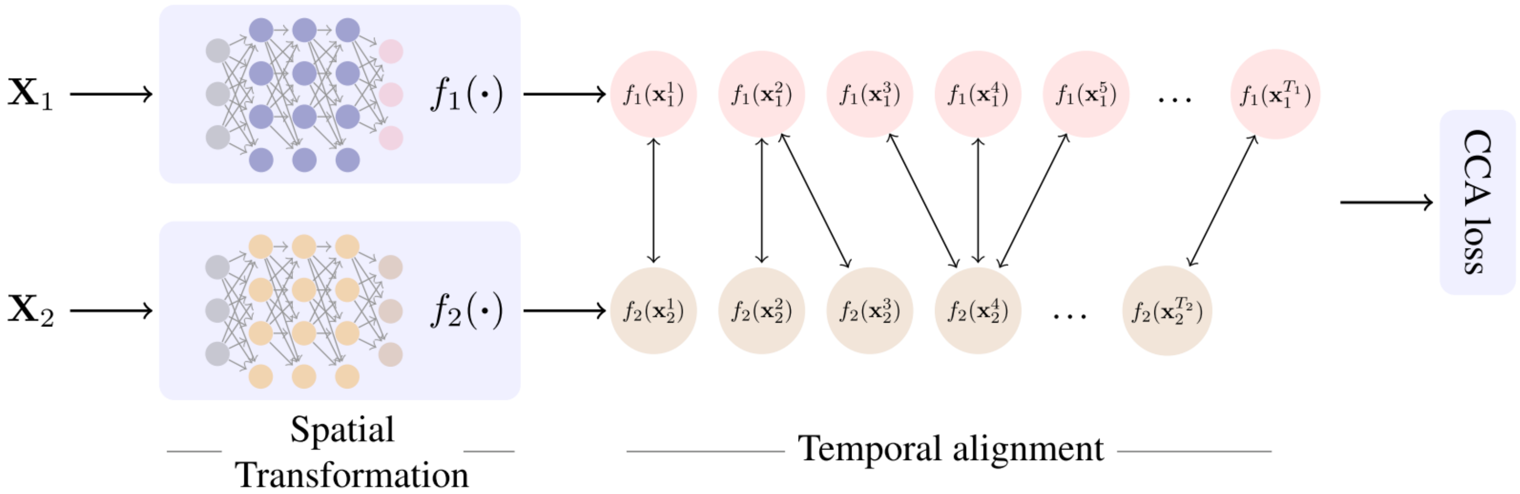


But how to model non-linear alignment functions?

Deep Canonical Time Warping

$$L(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{W}_x, \boldsymbol{W}_y) = \left\| f_{\boldsymbol{\theta}_1}(\mathbf{X})\mathbf{W_x} - f_{\boldsymbol{\theta}_1}(\mathbf{Y})\mathbf{W_y} \right\|_F^2$$

- Could be seen as generalization of DCCA and GTW



[Deep Canonical Time Warping, Trigeorgis et al., 2016, CVPR]

Deep Canonical Time Warping

$$L(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{W}_x, \boldsymbol{W}_y) = \left\| f_{\boldsymbol{\theta}_1}(\mathbf{X})\mathbf{W}_\mathbf{x} - f_{\boldsymbol{\theta}_1}(\mathbf{Y})\mathbf{W}_\mathbf{y} \right\|_F^2$$

- The projections are orthogonal (like in DCCA)
- Optimization is again iterative:
  - Solve for alignment ($\boldsymbol{W}_x, \boldsymbol{W}_y$) with fixed projections ($\boldsymbol{\theta}_1, \boldsymbol{\theta}_2$)
    - Eigen decomposition
  - Solve for projections ($\boldsymbol{\theta}_1, \boldsymbol{\theta}_2$) with fixed alignment ($\boldsymbol{W}_x, \boldsymbol{W}_y$)
    - Gradient descent
  - Repeat till convergence

[Deep Canonical Time Warping, Trigeorgis et al., 2016, CVPR]

Implicit alignment

Machine Translation

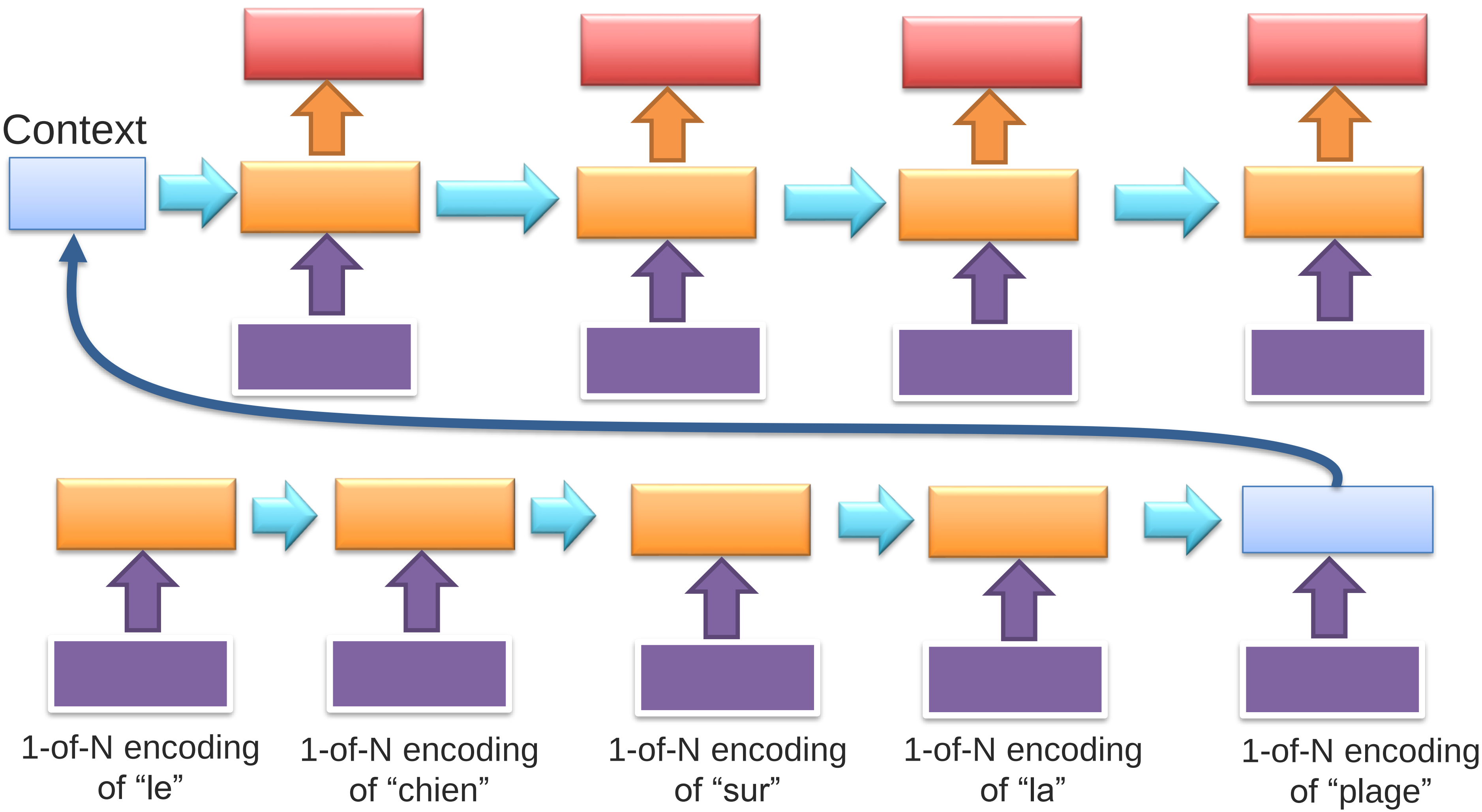
- Given a sentence in one language translate it to another

Dog on the beach → le chien sur la plage

- Not exactly multimodal task – but a good start! Each language can be seen almost as a modality.

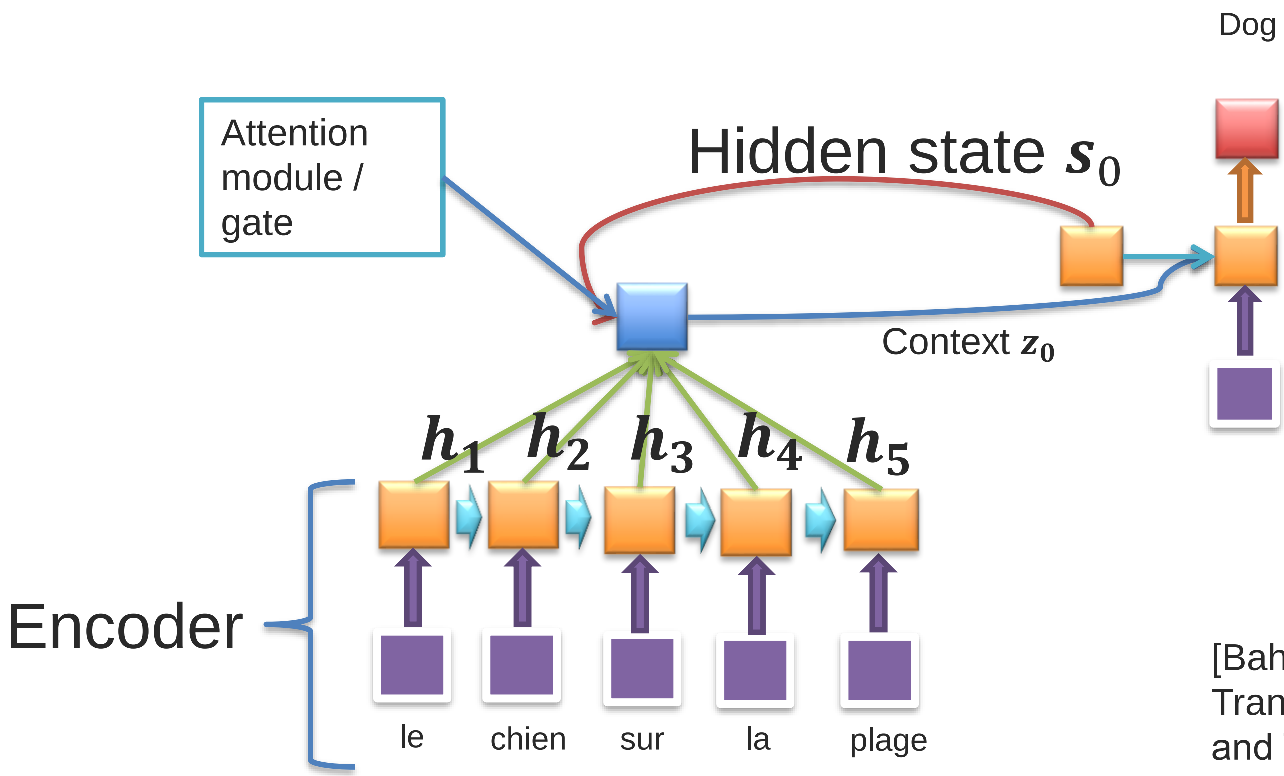
Encoder-Decoder Architecture for Machine Translation

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



Attention Model for Machine Translation

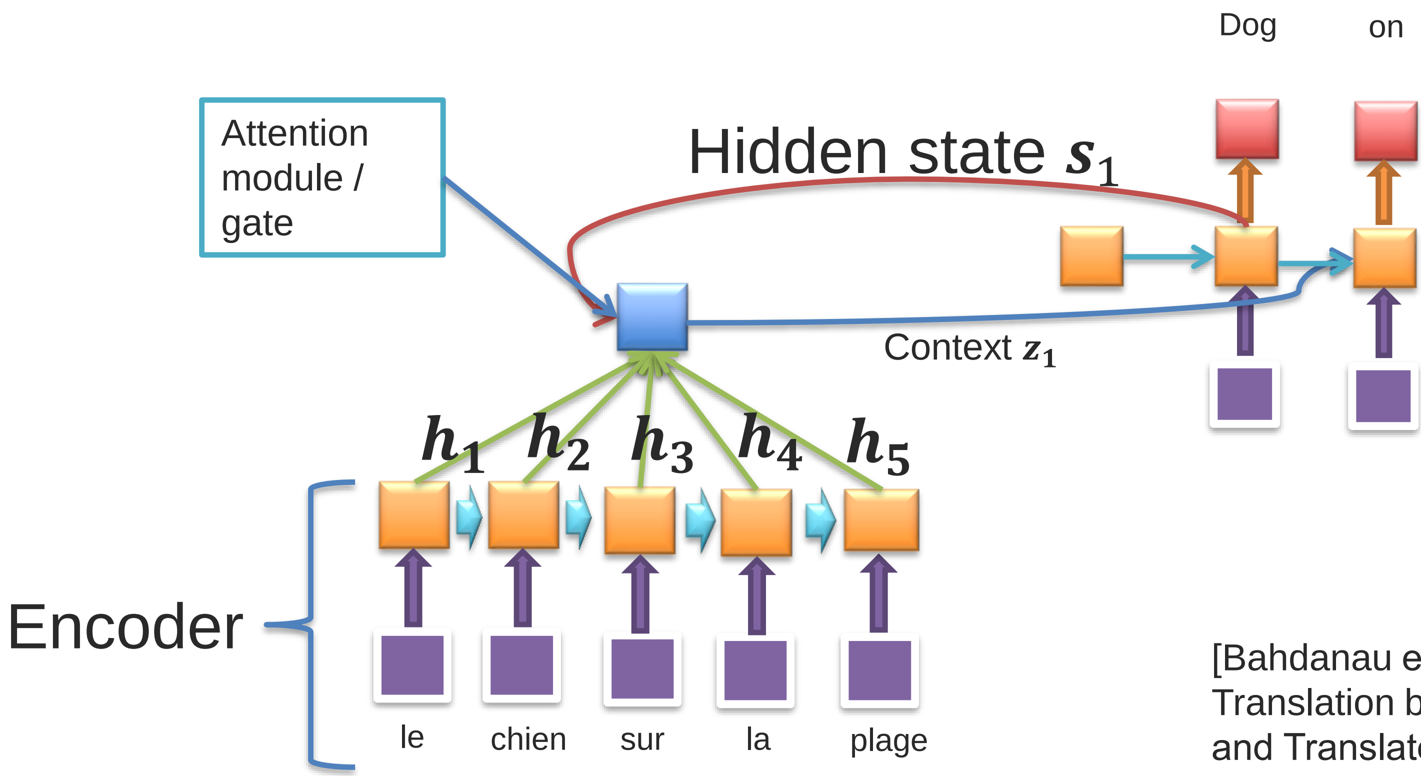
- Before encoder would just take the final hidden state, now we actually care about the intermediate hidden states



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

Attention Model for Machine Translation

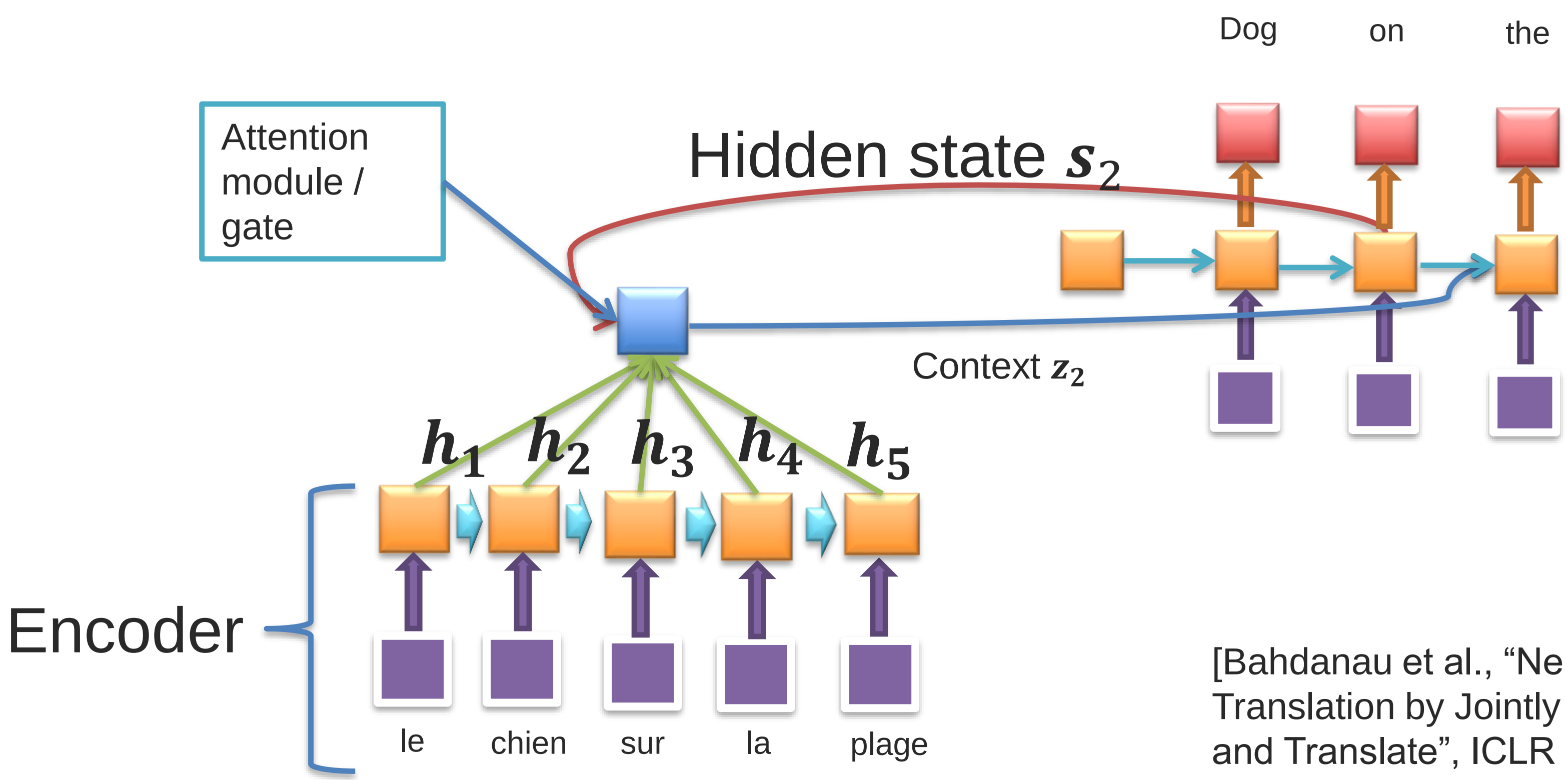
- Before encoder would just take the final hidden state, now we actually care about the intermediate hidden states



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

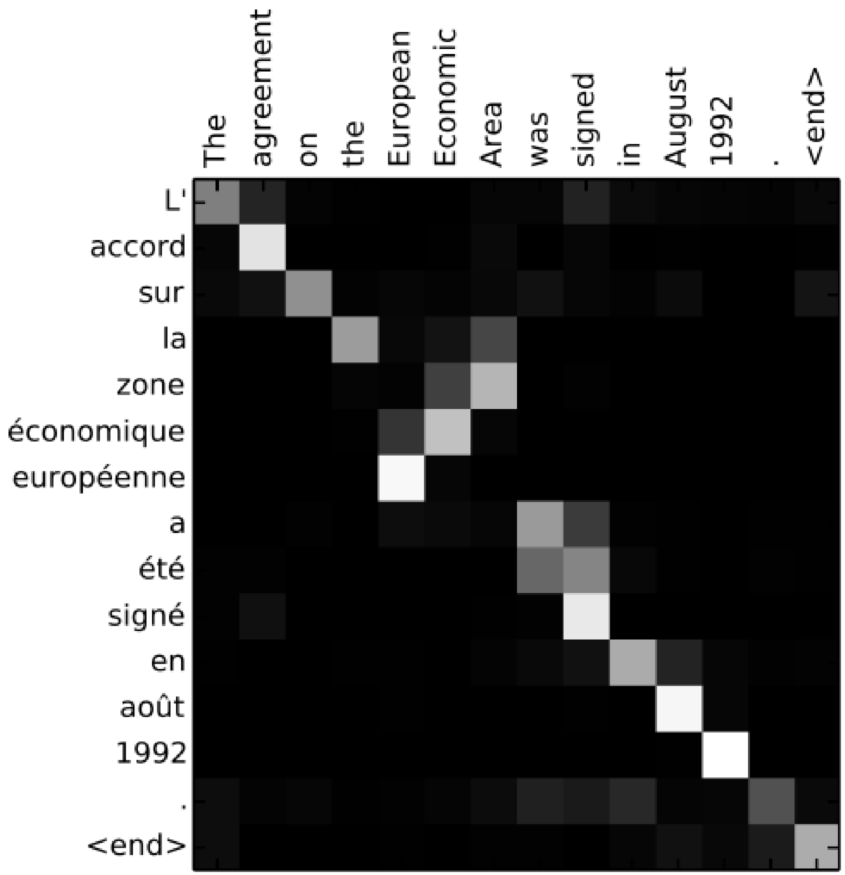
Attention Model for Machine Translation

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

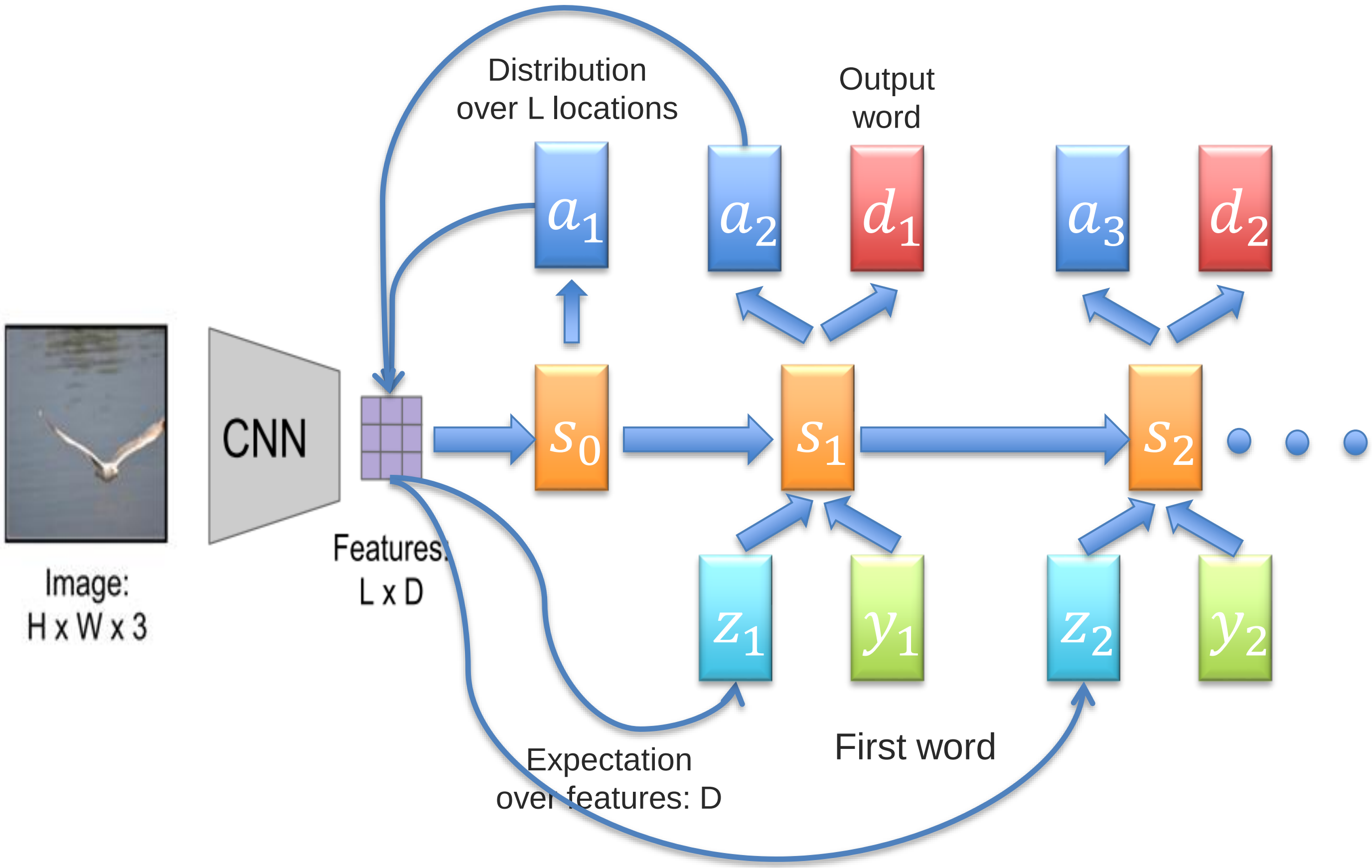


[Bahdanau et al., "Neural Machine Translation by Jointly Learning to Align and Translate", ICLR 2015]

Attention Model for Machine Translation



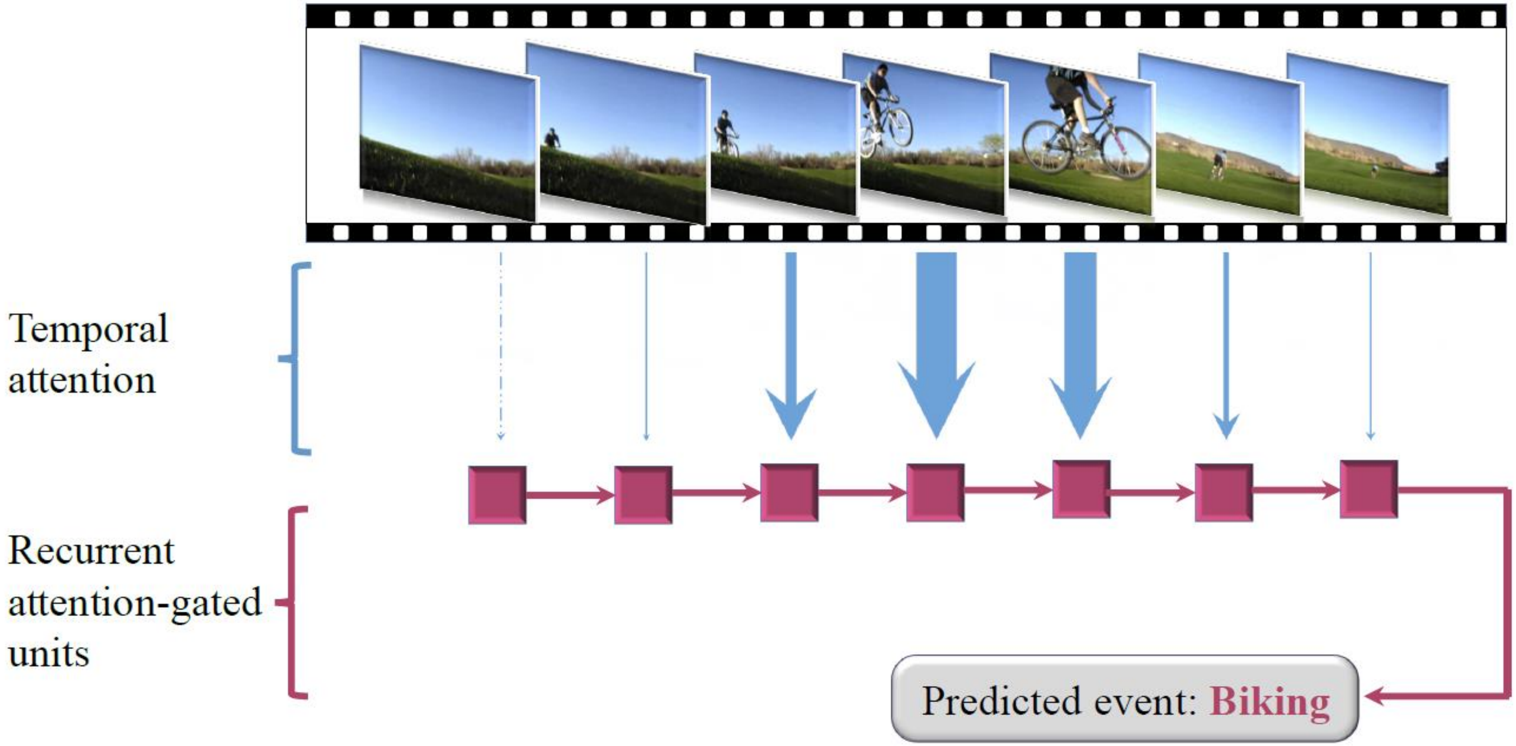
Attention Model for Image Captioning



Attention Model for Image Captioning

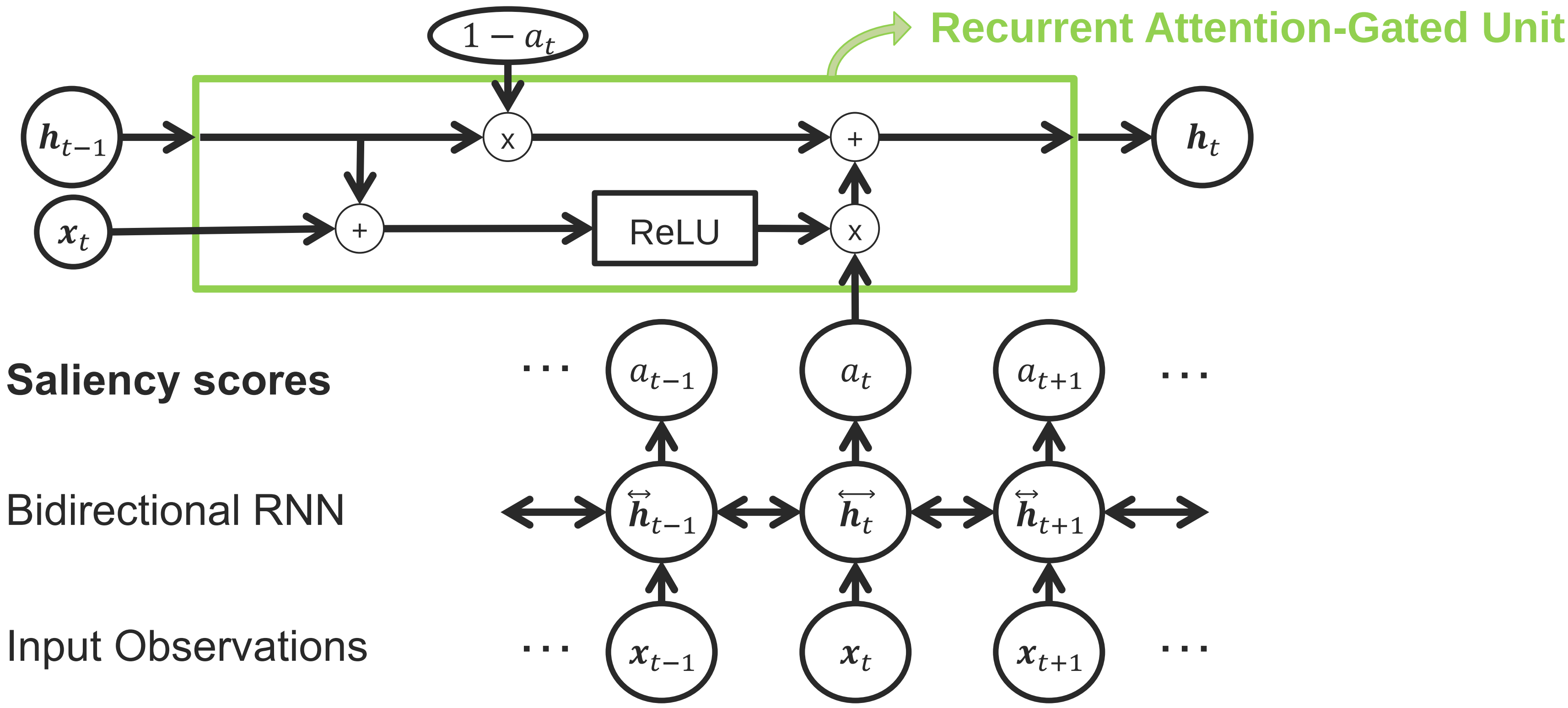
Xu et.al., ICML 2015

Attention Model for Video Sequences



[Pei, Baltrušaitis, Tax and Morency. Temporal Attention-Gated Model for Robust Sequence Classification, **CVPR**, *2017]*

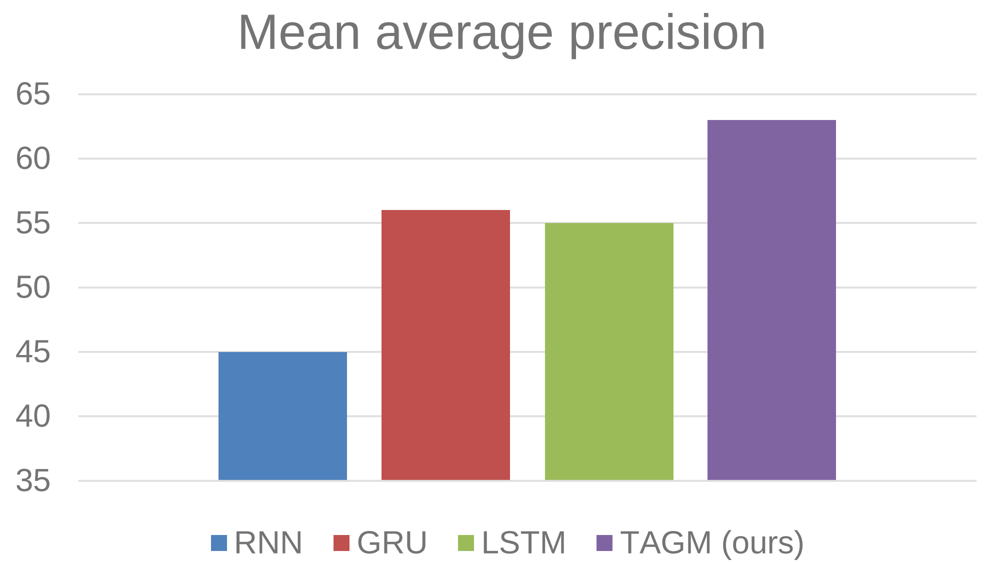
Temporal Attention-Gated Model (TAGM)



Temporal Attention Gated Model (TAGM)

CCV dataset

- 20 video categories
- Biking, birthday, wedding etc.



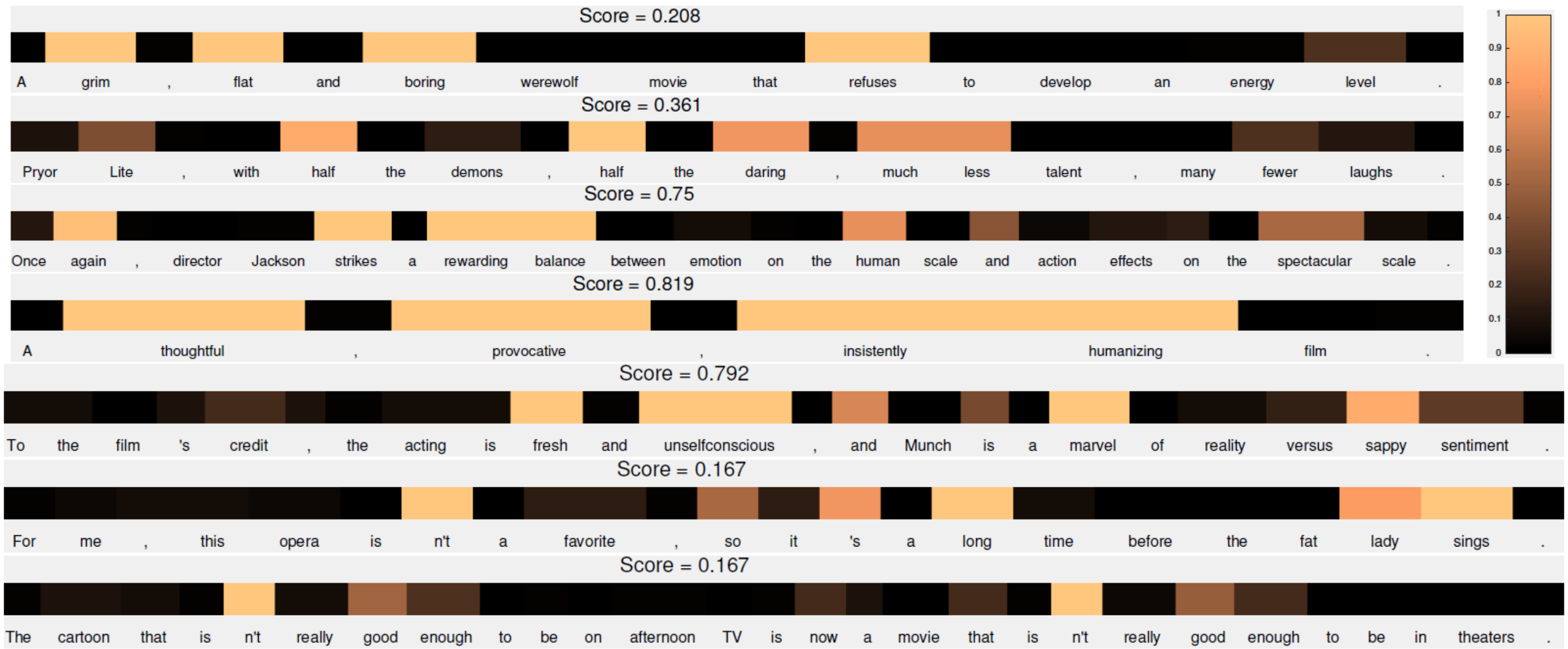
[Pei, Baltrušaitis, Tax and Morency. Temporal Attention-Gated Model for Robust Sequence Classification, **CVPR**, *2017]*

Temporal Attention Gated Model (TAGM)

[Pei, Baltrušaitis, Tax and Morency. Temporal Attention-Gated Model for Robust Sequence Classification, **CVPR**, *2017]*

Temporal Attention Gated Model (TAGM)

Text-based Sentiment Analysis

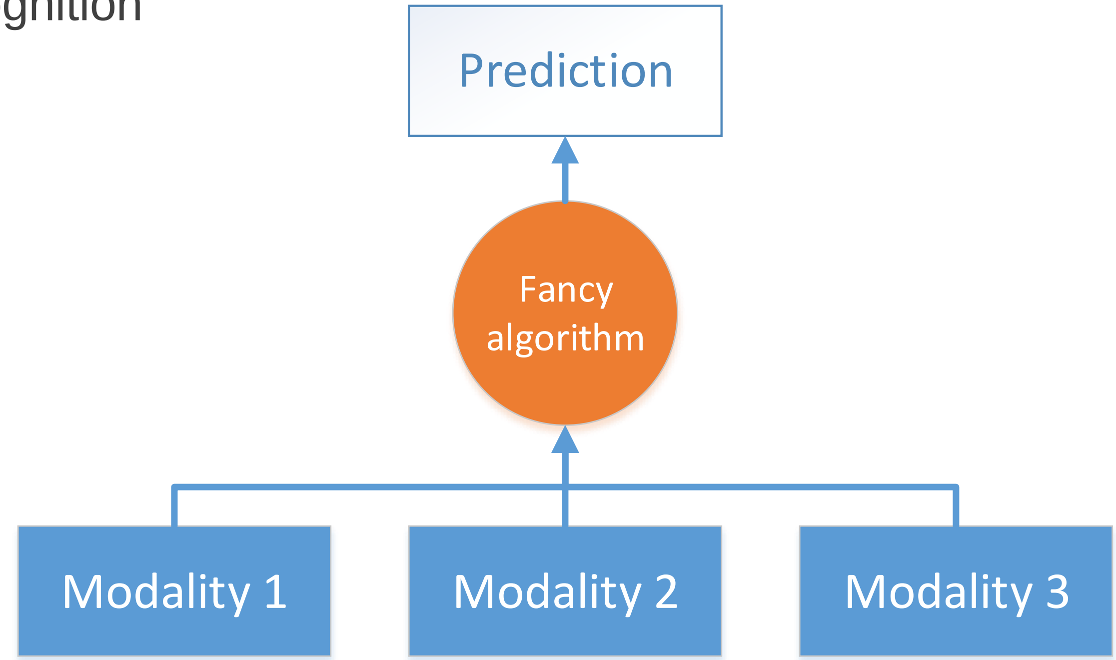


[Pei, Baltrušaitis, Tax and Morency. Temporal Attention-Gated Model for Robust Sequence Classification, **CVPR**, *2017*]

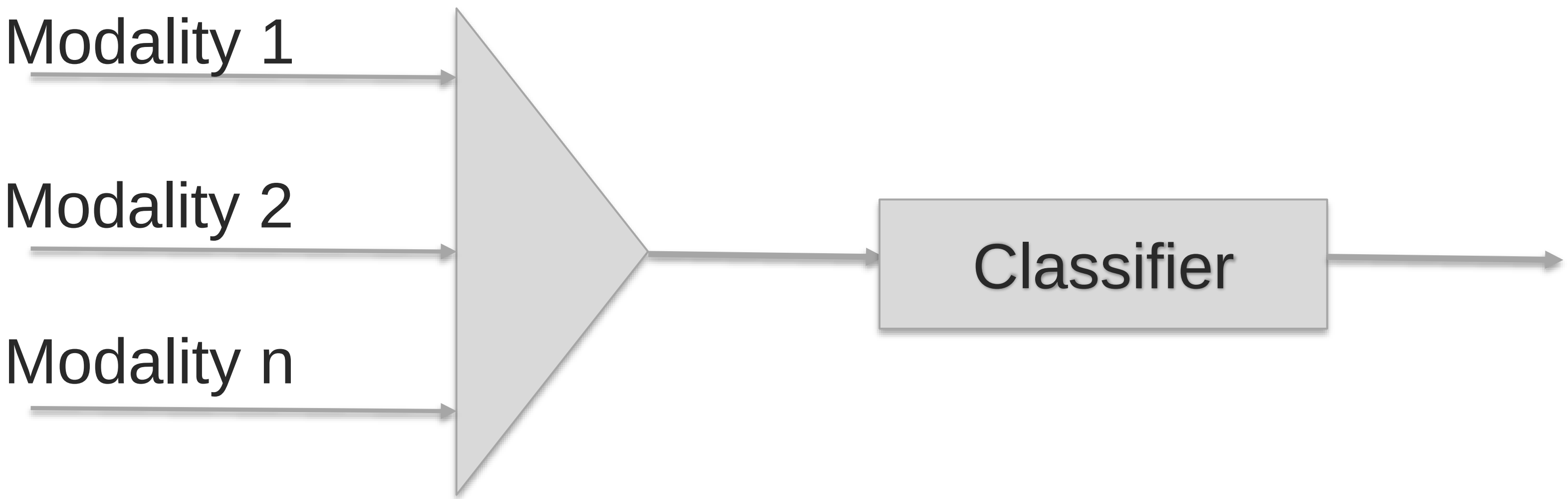
Multimodal Fusion

Multimodal Fusion

- Process of joining information from two or more modalities to perform a prediction
  - One of the earlier and more established problems
  - e.g. audio-visual speech recognition, multimedia event detection, multimodal emotion recognition
- Two major types
- Model Free
  - Early, late, hybrid
- Model Based
  - Kernel Methods
  - Graphical models
  - Neural networks

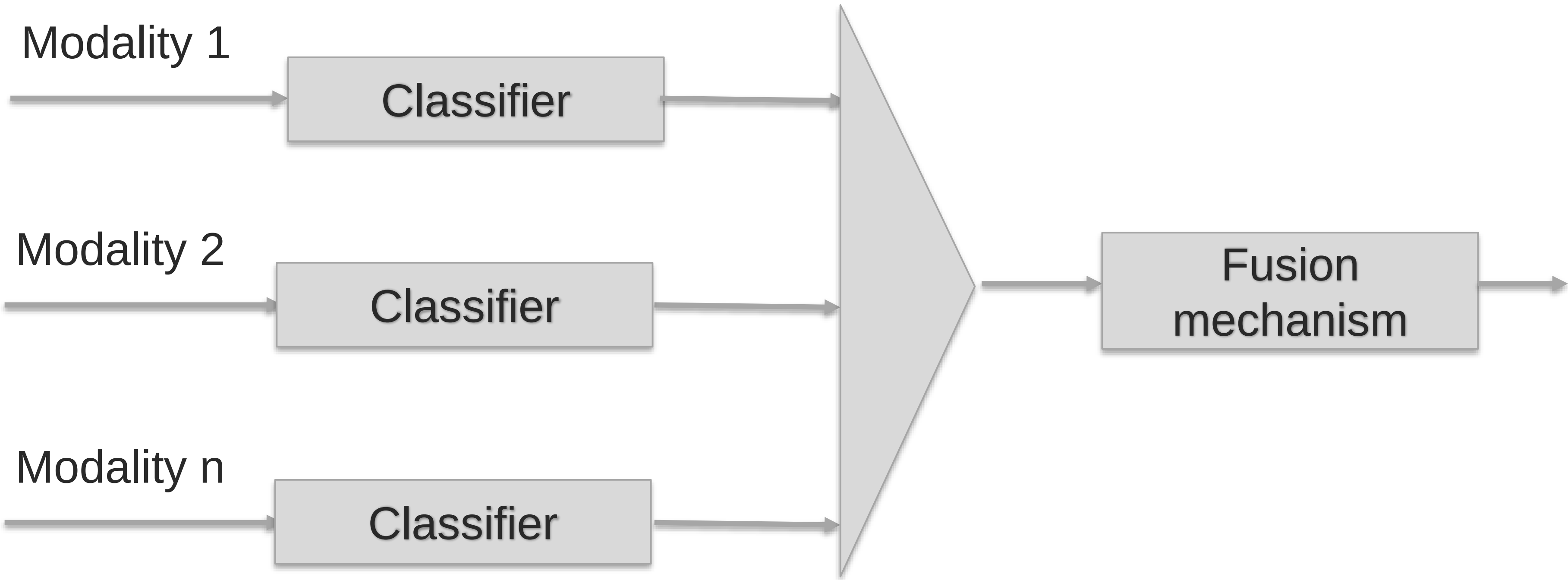


Model free approaches – early fusion



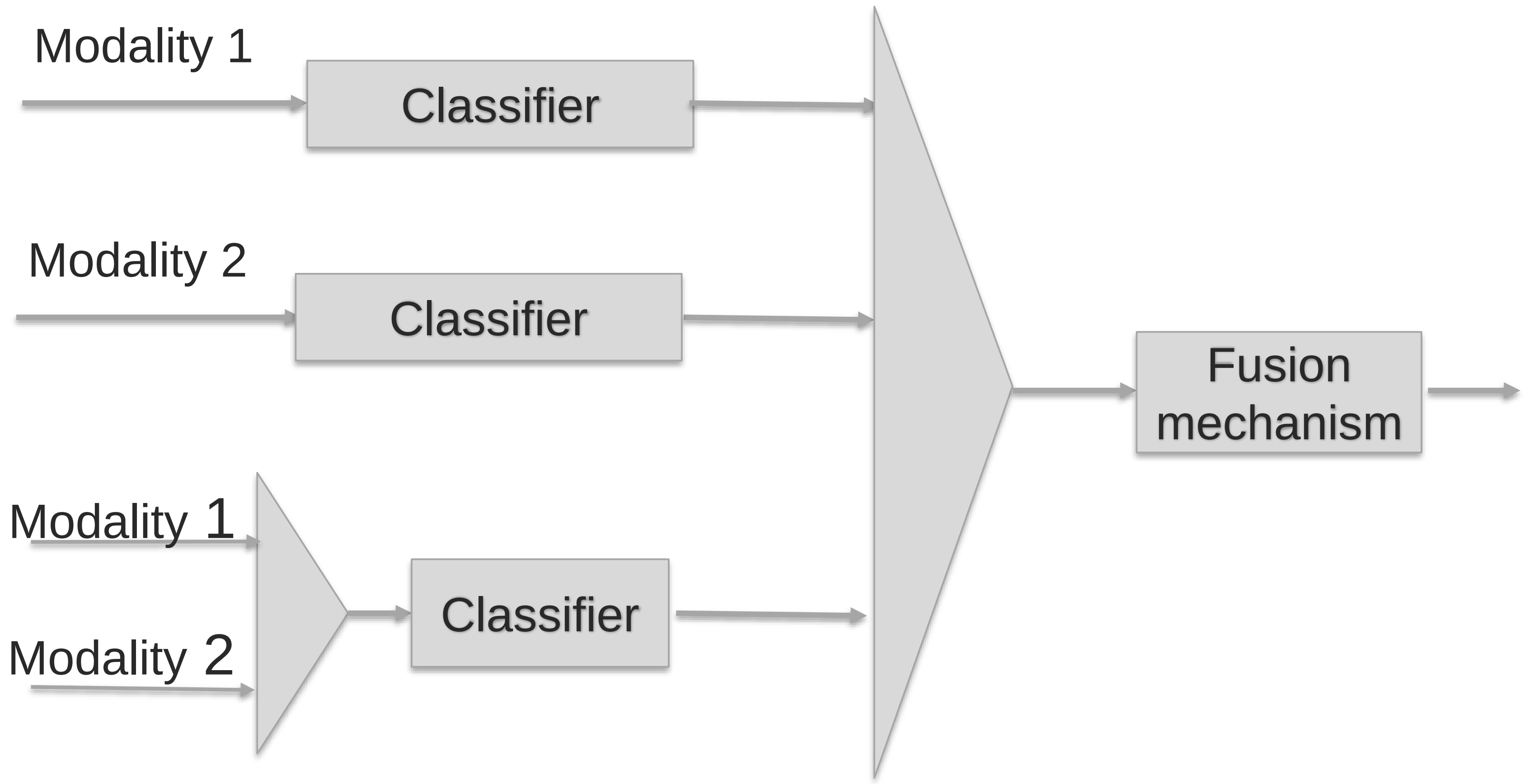
- Easy to implement – just concatenate the features
- Exploit dependencies between features
- Can end up very high dimensional
- More difficult to use if features have different framerates

Model free approaches – late fusion



- Train a unimodal predictor and a multimodal fusion one
- Requires multiple training stages
- Do not model low level interactions between modalities
- Fusion mechanism can be voting, weighted sum or an ML approach

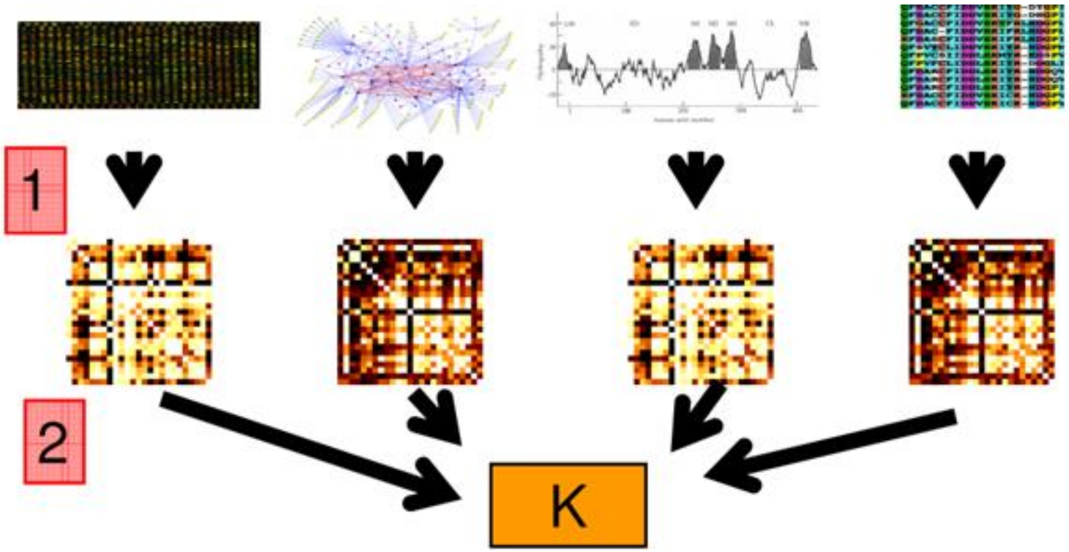
Model free approaches – hybrid fusion



- Combine benefits of both early and late fusion mechanisms

Multiple Kernel Learning

- Pick a family of kernels for each modality and learn which kernels are important for the classification case
- Generalizes the idea of Support Vector Machines
- Works as well for unimodal and multimodal data, very little adaptation is needed



[Lanckriet 2004]

Multimodal Fusion for Sequential Data

Modality-*private* structure

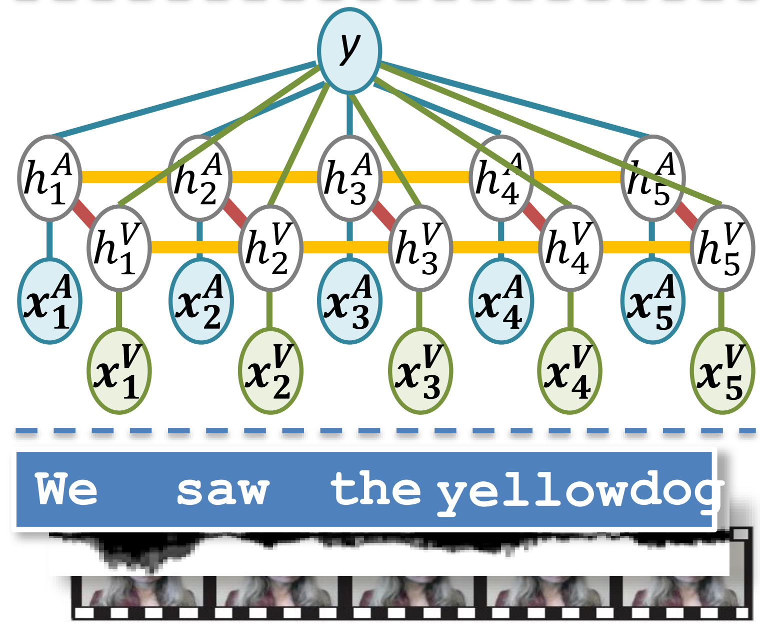
- Internal grouping of observations

Modality-*shared* structure

- Interaction and synchrony

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

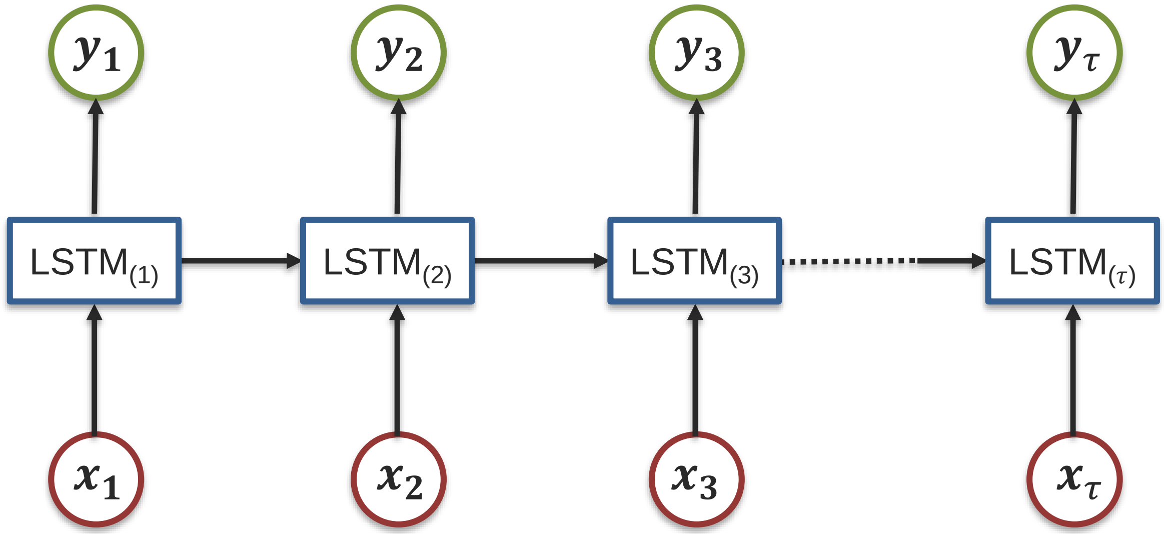
Multi-View Hidden Conditional Random Field



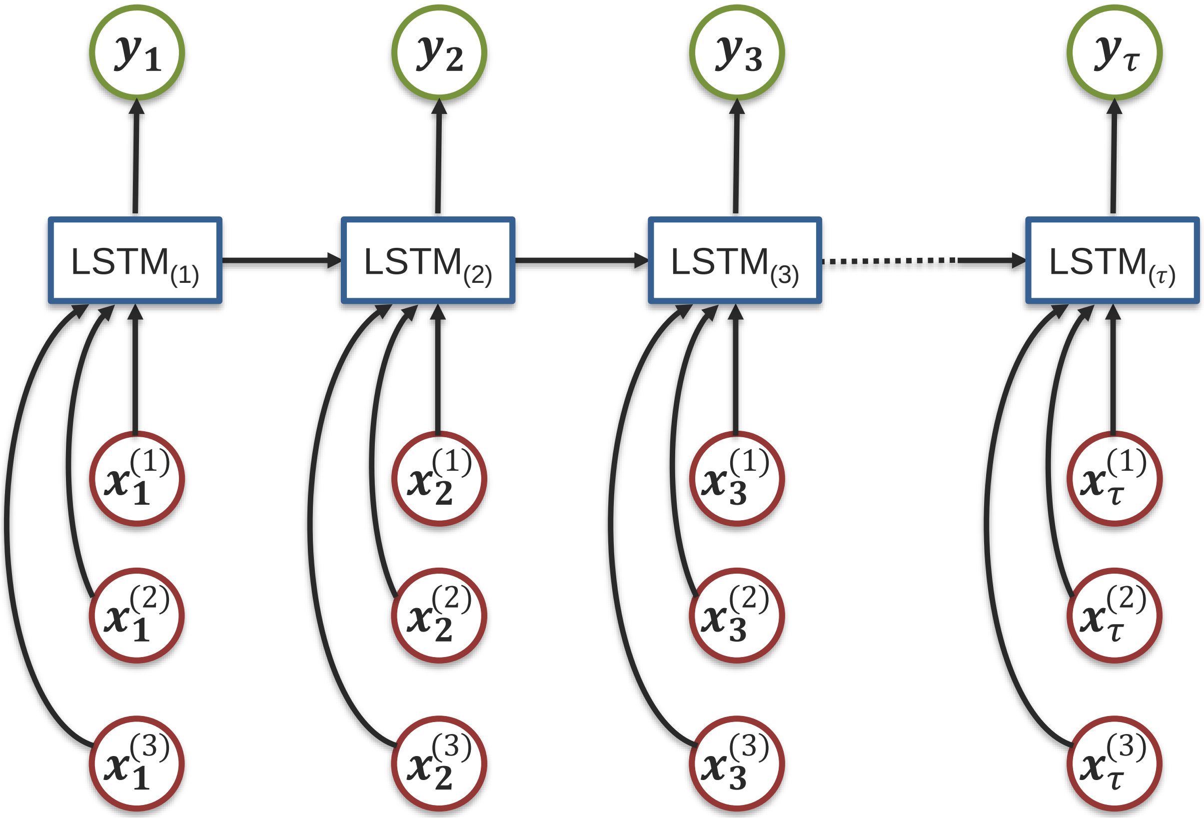
➢ Approximate inference using loopy-belief

[Song, Morency and Davis, CVPR 2012]

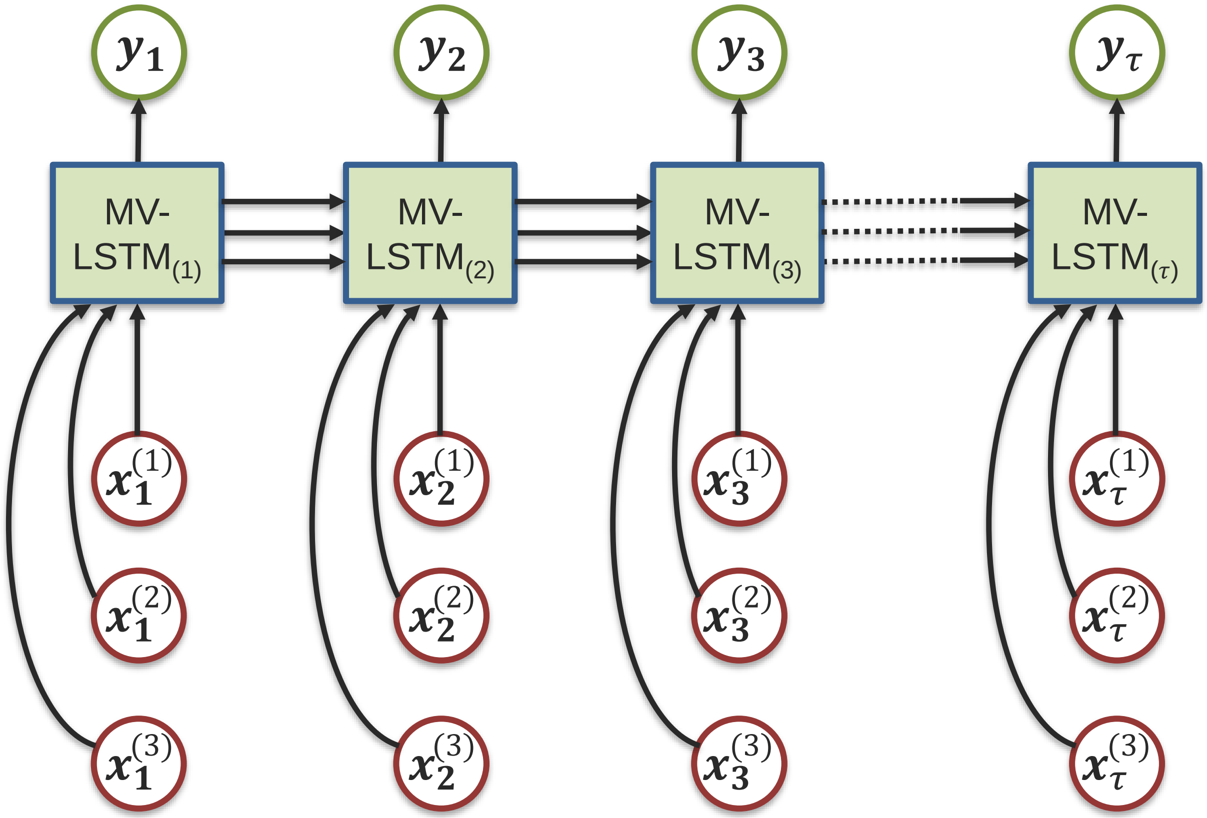
Sequence Modeling with LSTM



Multimodal Sequence Modeling – Early Fusion

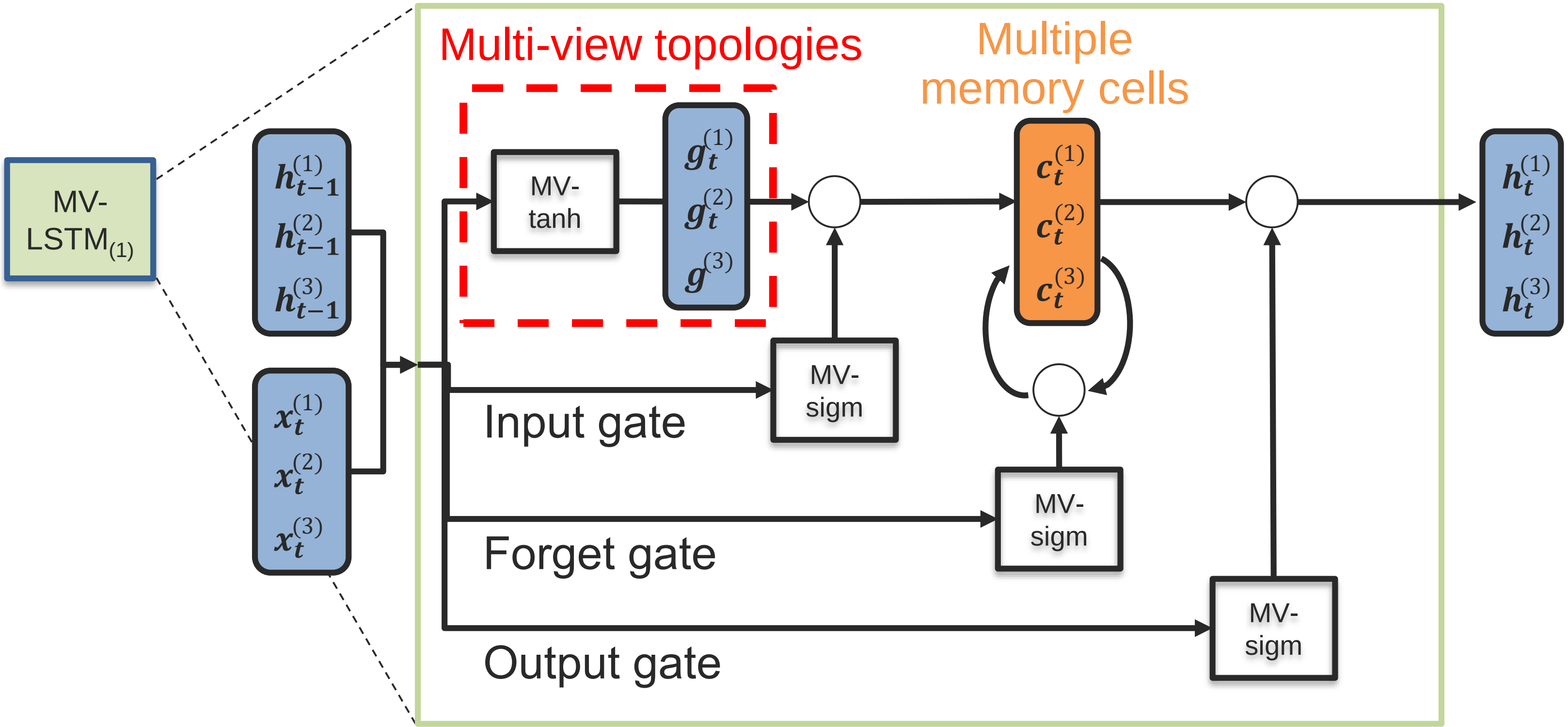


Multi-View Long Short-Term Memory (MV-LSTM)



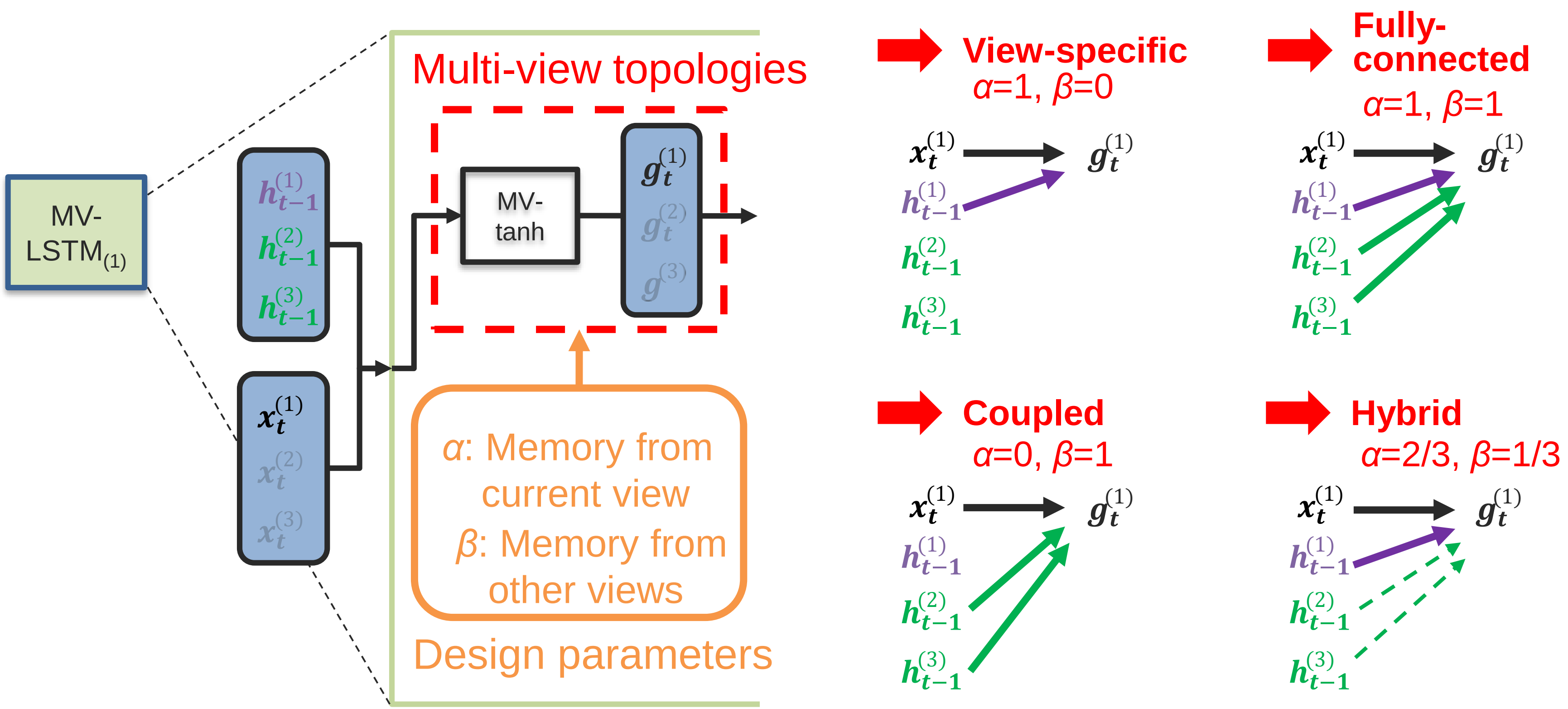
[Shyam, Morency, et al. Extending Long Short-Term Memory for Multi-View Structured Learning, **ECCV**, 2016]

Multi-View Long Short-Term Memory



[Shyam, Morency, et al. Extending Long Short-Term Memory for Multi-View Structured Learning, **ECCV**, 2016]

Topologies for Multi-View LSTM



[Shyam, Morency, et al. Extending Long Short-Term Memory for Multi-View Structured Learning, **ECCV**, 2016]

Multi-View Long Short-Term Memory (MV-LSTM)

Multimodal prediction of children engagement

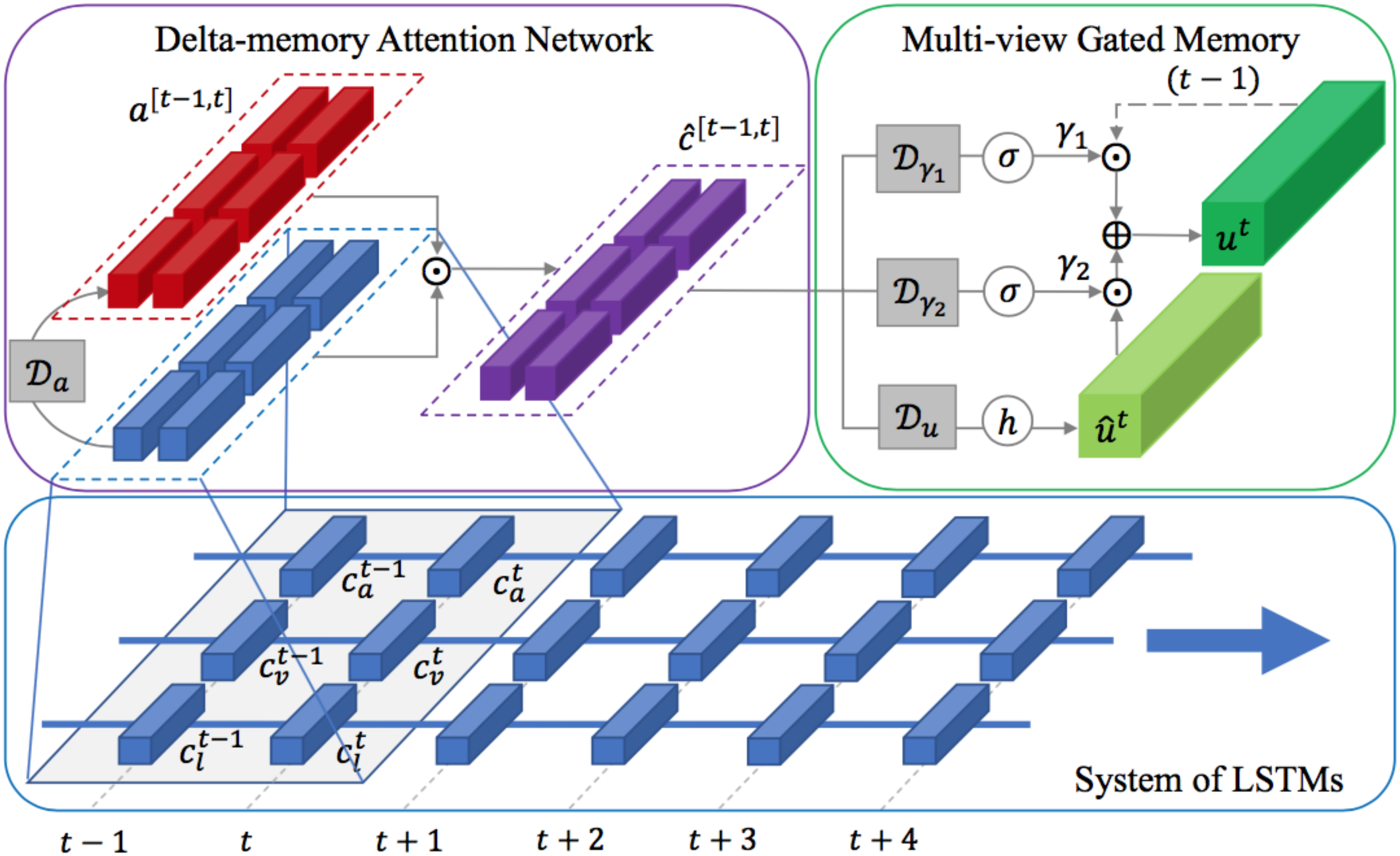
Class labels	Model	Precision	Recall	F1
Easy to engage	LSTM (Early fusion)	0.75	0.81	0.78
	MV-LSTM Full	0.81	0.81	0.81
	MV-LSTM Coupled	0.79	0.81	0.80
	MV-LSTM Hybrid	**0.80**	**0.86**	**0.83**
Difficult to engage	LSTM (Early fusion)	0.63	0.55	0.59
	MV-LSTM Full	0.68	0.68	0.68
	MV-LSTM Coupled	0.67	0.64	0.65
	MV-LSTM Hybrid	**0.74**	**0.64**	**0.68**

[Shyam, Morency, et al. Extending Long Short-Term Memory for Multi-View Structured Learning, **ECCV**, 2016]

Memory Based

- A memory accumulates multimodal information over time.
- From the representations throughout a source network.
- No need to modify the structure of the source network, only attached the memory.

Memory Based



[Zadeh et al., Memory Fusion Network for Multi-view Sequential Learning, AAAI 2018]

Multimodal Machine Learning

Representation

Alignment

Fusion

Translation

Co-Learning

Multimodal Machine Learning: A Survey and Taxonomy

By Tadas Baltrusaitis, Chaitanya Ahuja, and Louis-Philippe Morency

https://arxiv.org/abs/1705.09406

- ☑ **5 core challenges**
- ☑ **37 taxonomic classes**
- ☑ **253 referenced citations**