THE UNIVERSITY OF BRITISH COLUMBIA

Topics in AI (CPSC 532S):
Multimodal Learning with Vision, Language and Sound

Lecture 14: Coordinated Representations and Joint Embeddings

Multimodal Representations

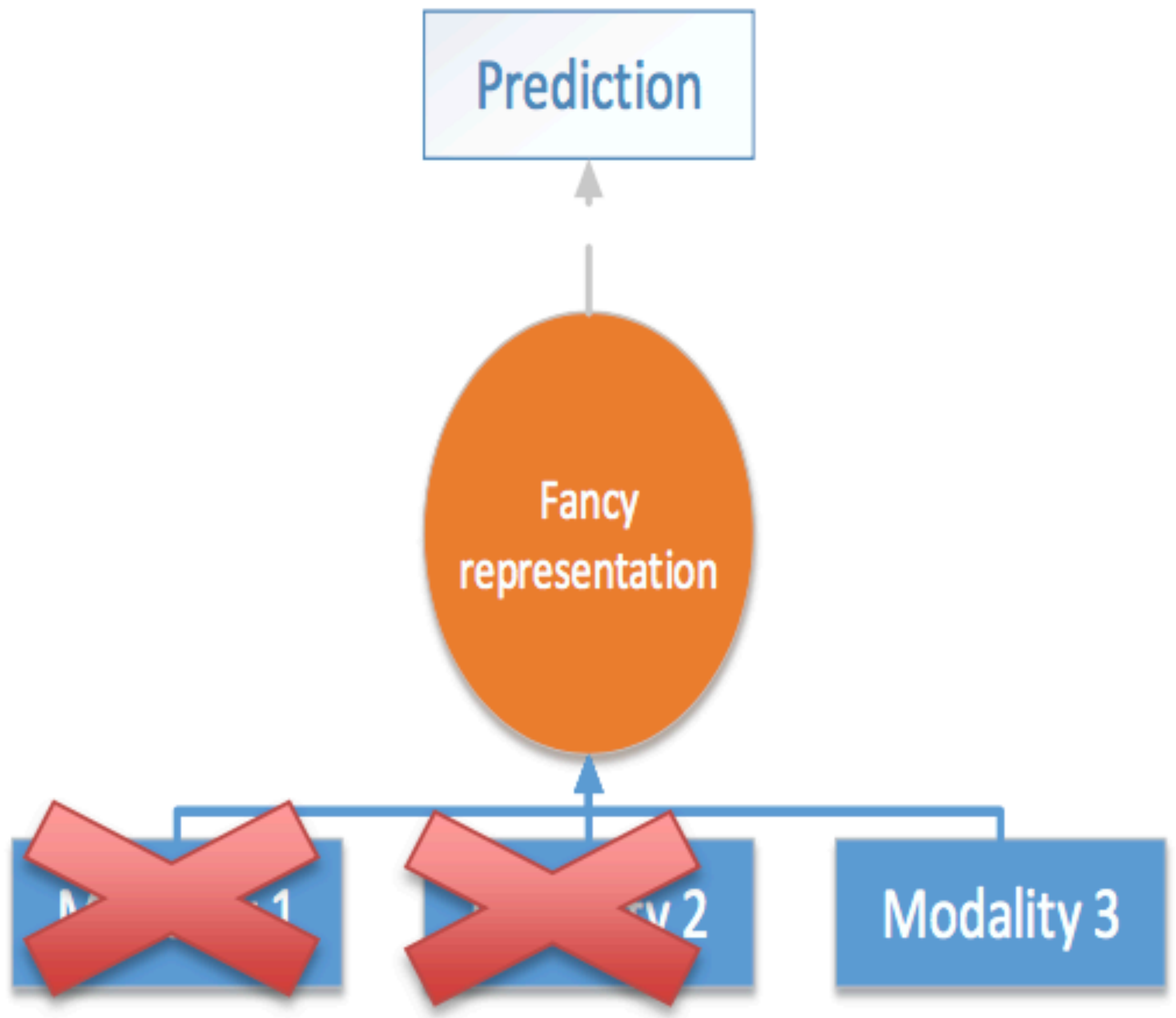
What is a **good** multimodal representation?

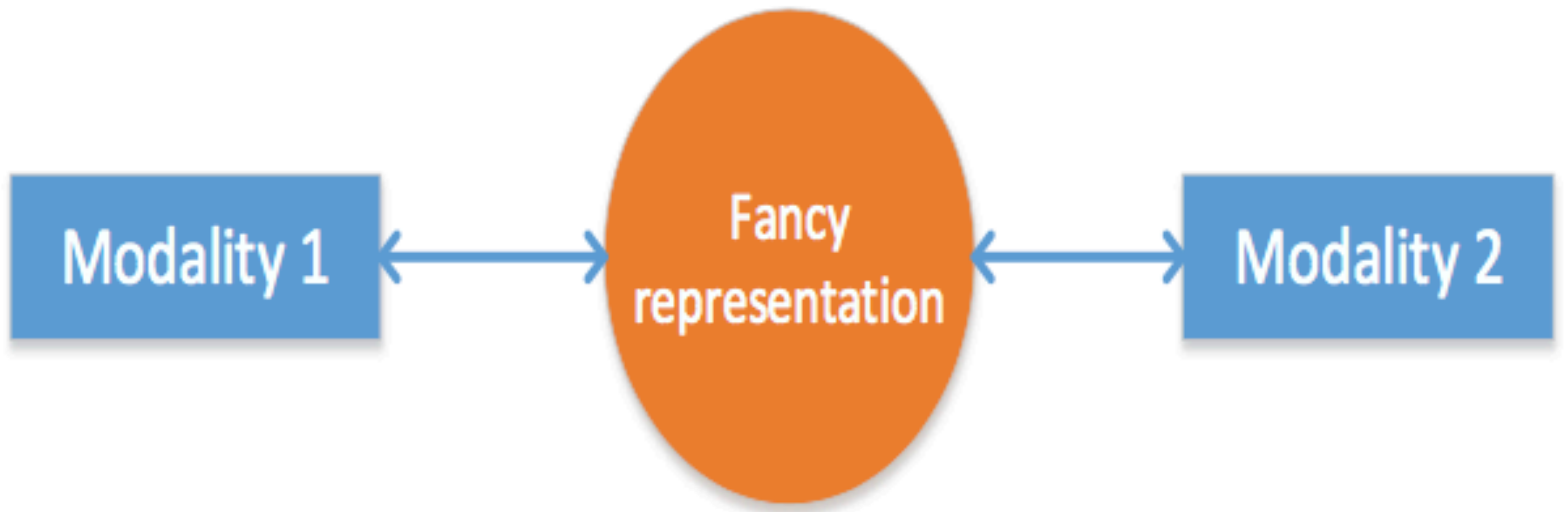
— **Similarity** in the representation (somehow) implies similarity in corresponding concepts
(we saw this in word2vec)

— **Useful** for various **discriminative tasks** (retrieval, mapping, fusion, etc.)

— Possible to obtain **in absence of one or mere modalities**

— **Fill in missing modalities** given others (map or translate between modalities)

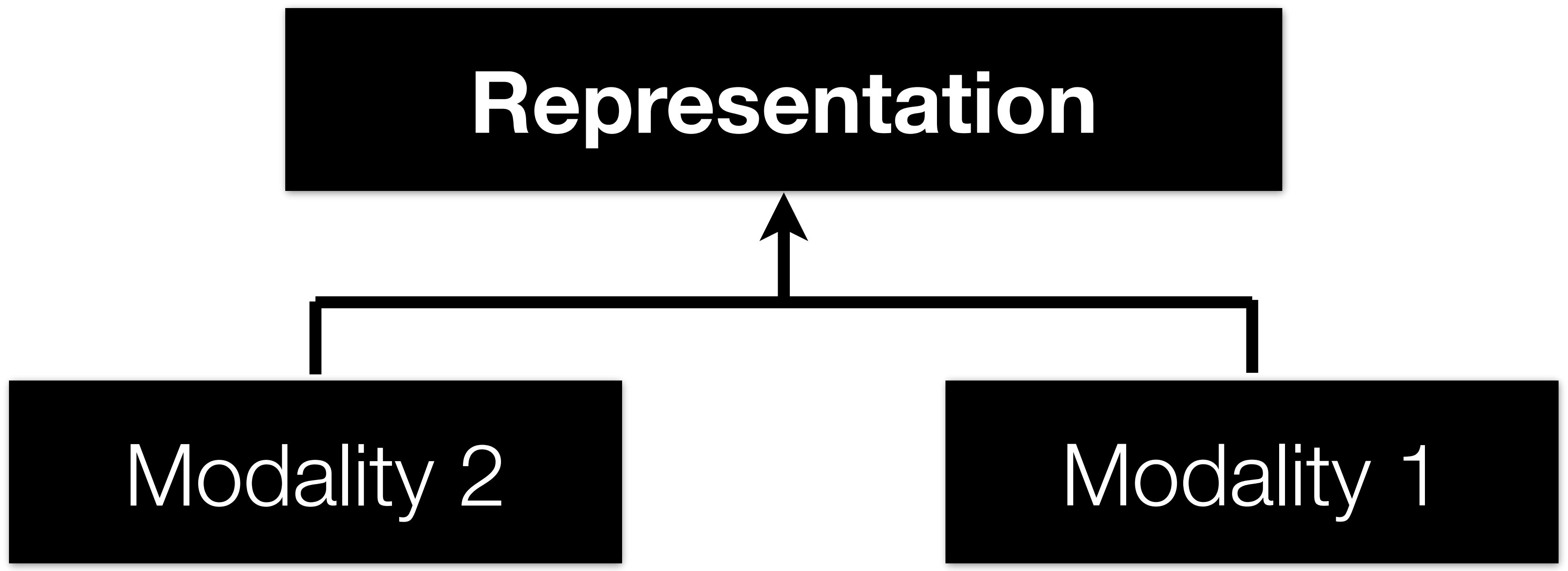




*slide from Louis-Philippe Morency

Multimodal Representation Types

Joint representations:



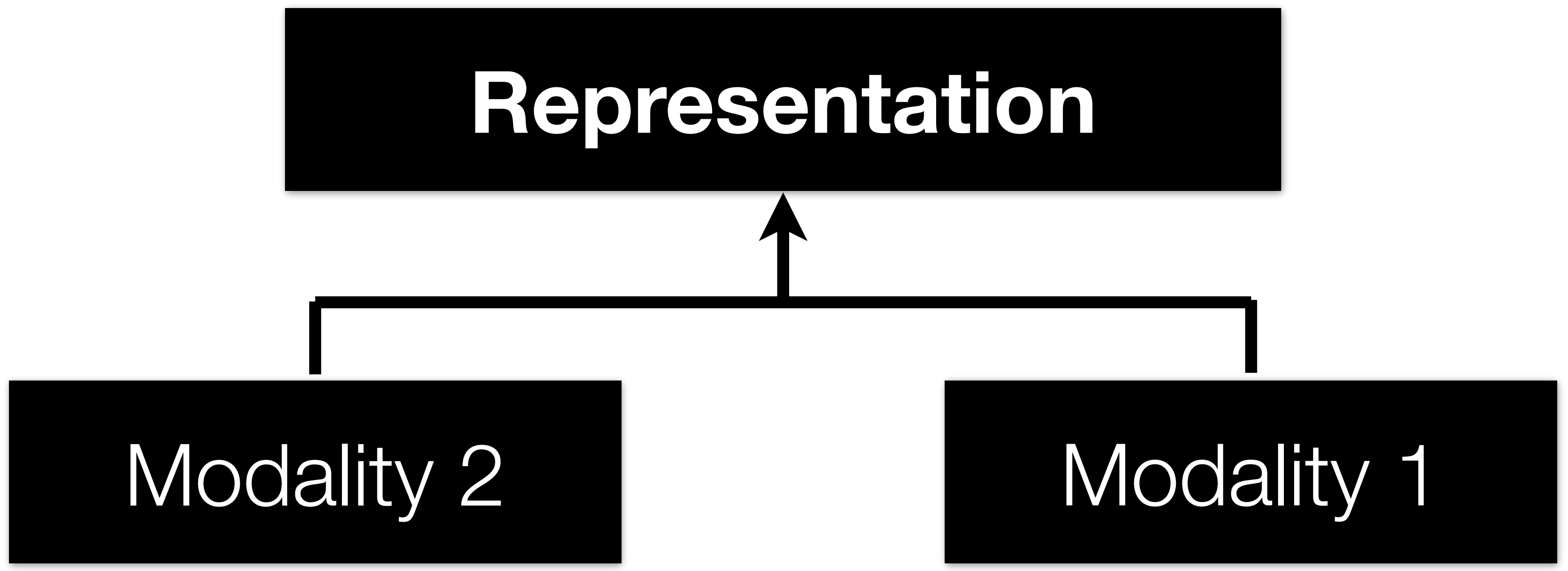
— Simplest version: **modality concatenation** (early fusion)

— Can be learned **supervised** or **unsupervised**

*slide from Louis-Philippe Morency

Multimodal Representation Types

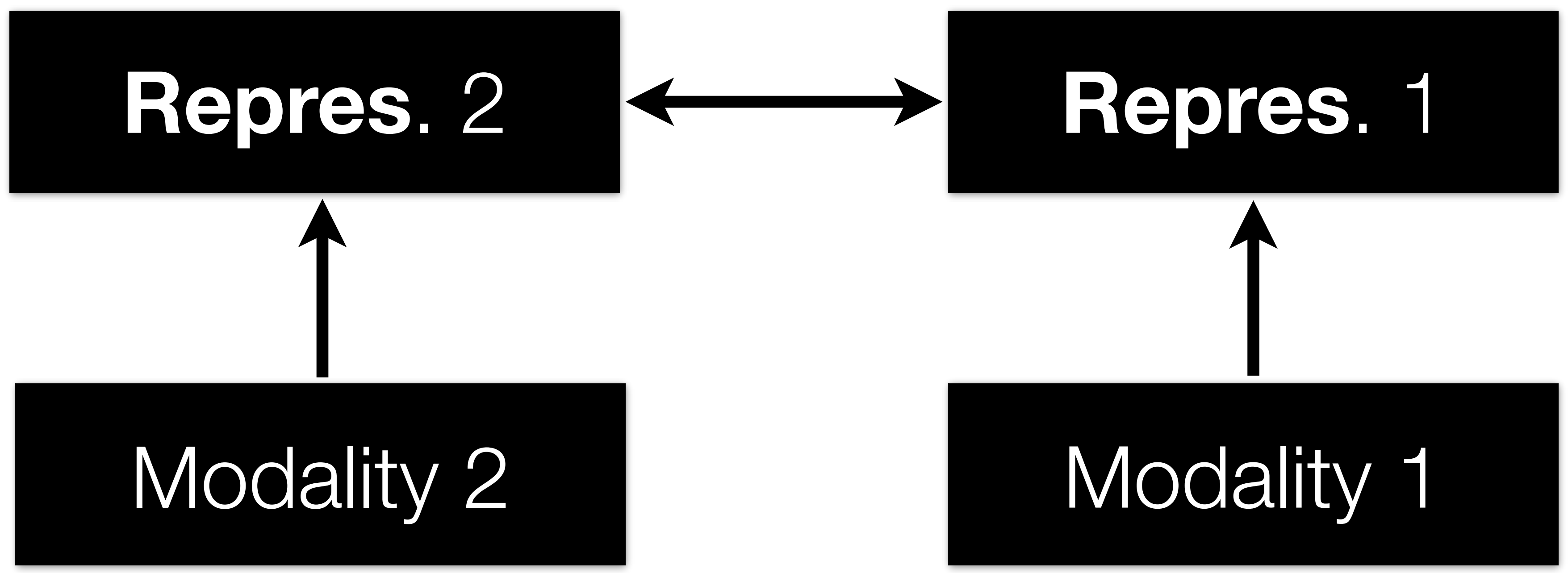
Joint representations:



— Simplest version: **modality concatenation** (early fusion)

— Can be learned **supervised** or **unsupervised**

Coordinated representations:



— **Similarity-based** methods (e.g., cosine distance)

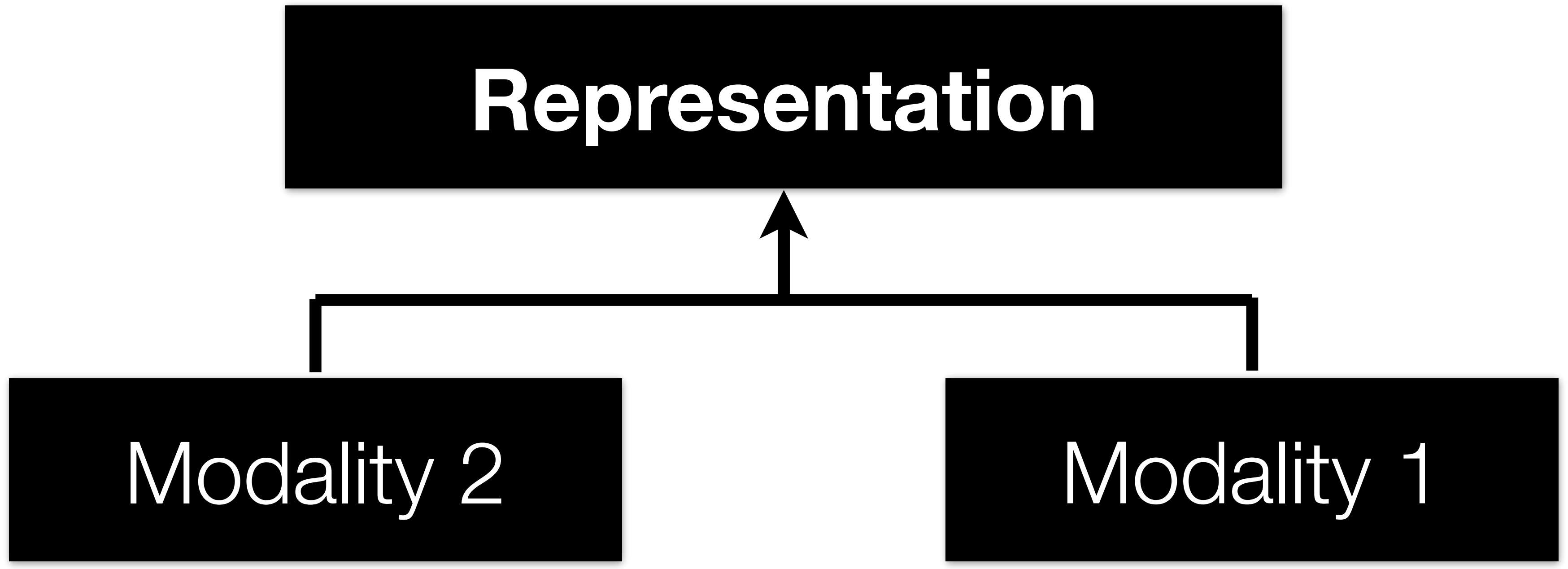
— **Structure constraints** (e.g., orthogonality, sparseness)

— Examples: CCA, joint embeddings

*slide from Louis-Philippe Morency

Multimodal Representation Types

Joint representations:



— Simplest version: **modality concatenation** (early fusion)

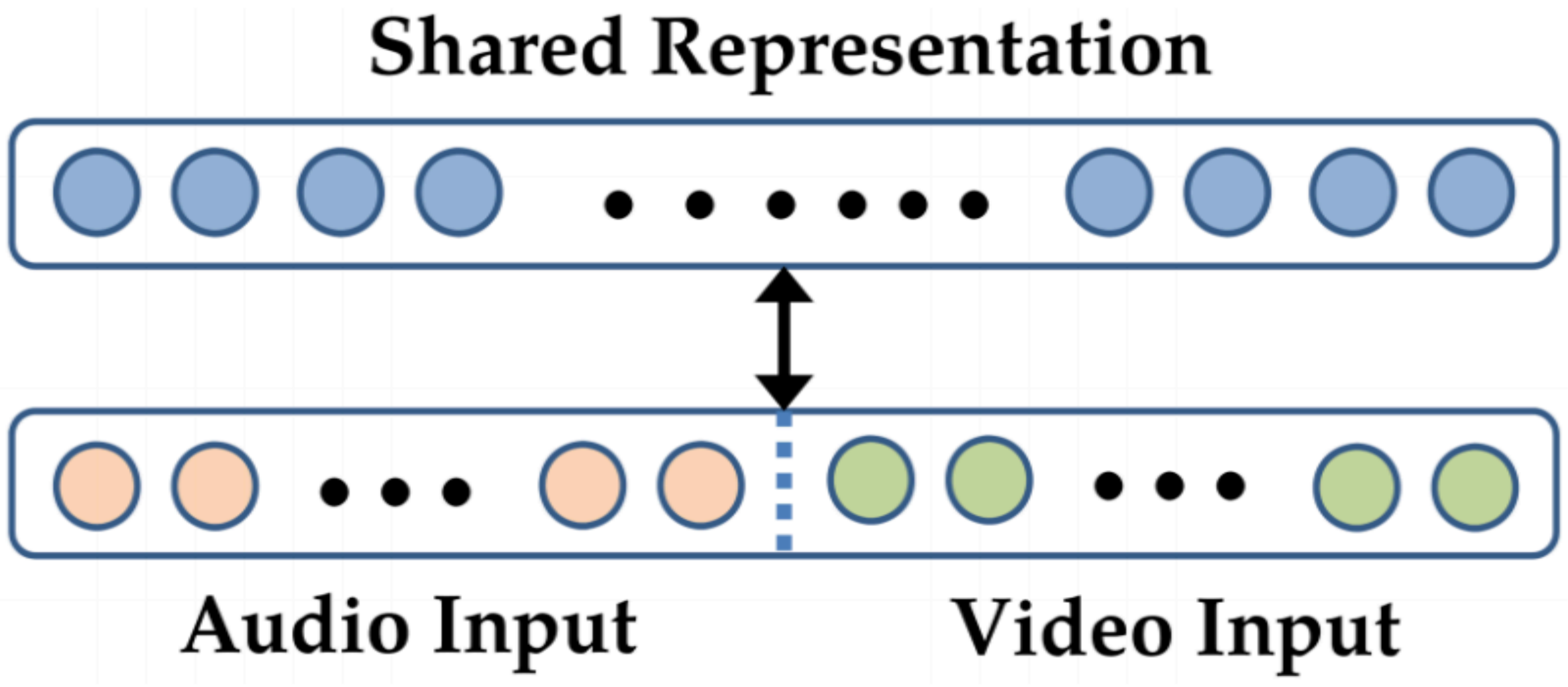
— Can be learned **supervised** or **unsupervised**

*slide from Louis-Philippe Morency

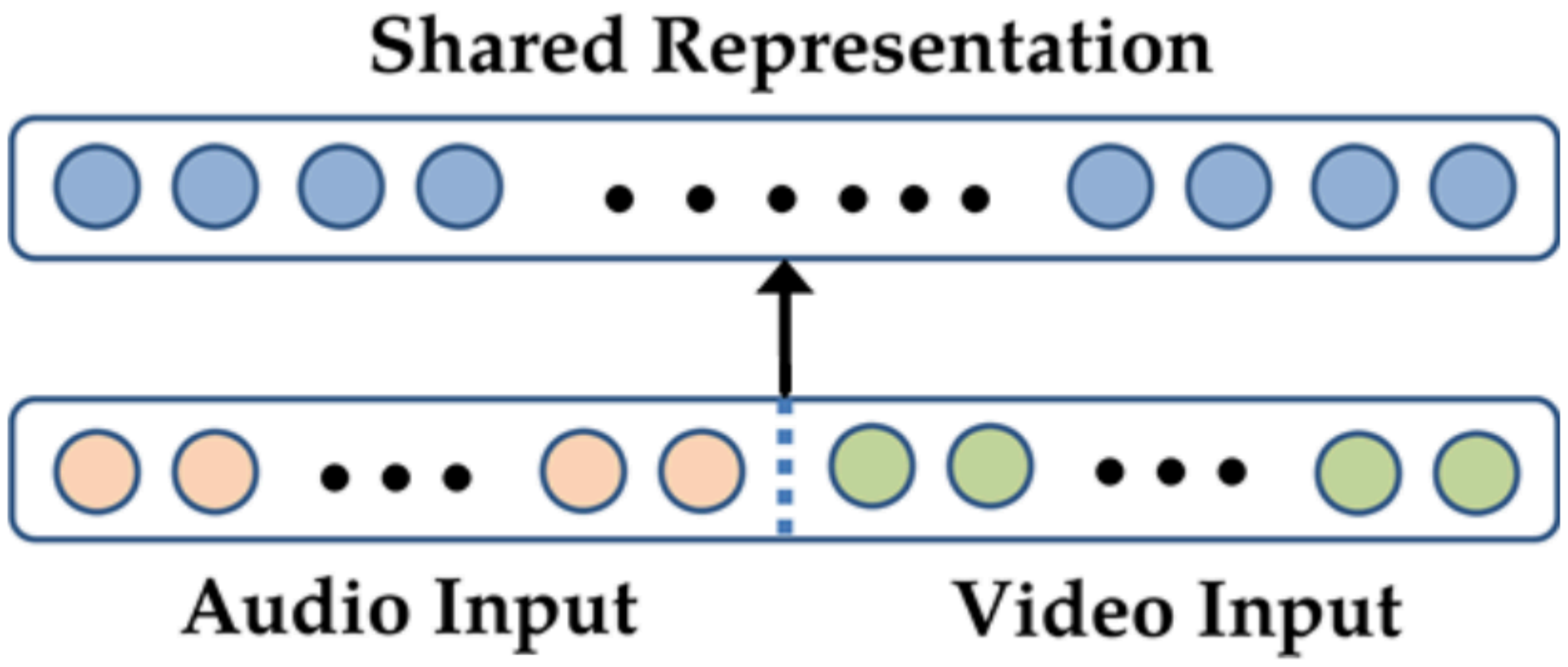
Joint Representation: Simple Multimodal Autoencoders

Concatenating modalities is fine, but requires both modalities at test time

No ability to ensure there is indeed **sharing** in the representations space



Shallow RBM



Shallow Autoencoder

Joint Representation: Deep Multimodal Autoencoders

[Ngiam et al., 2011]

Each **modality** can be pre-trained

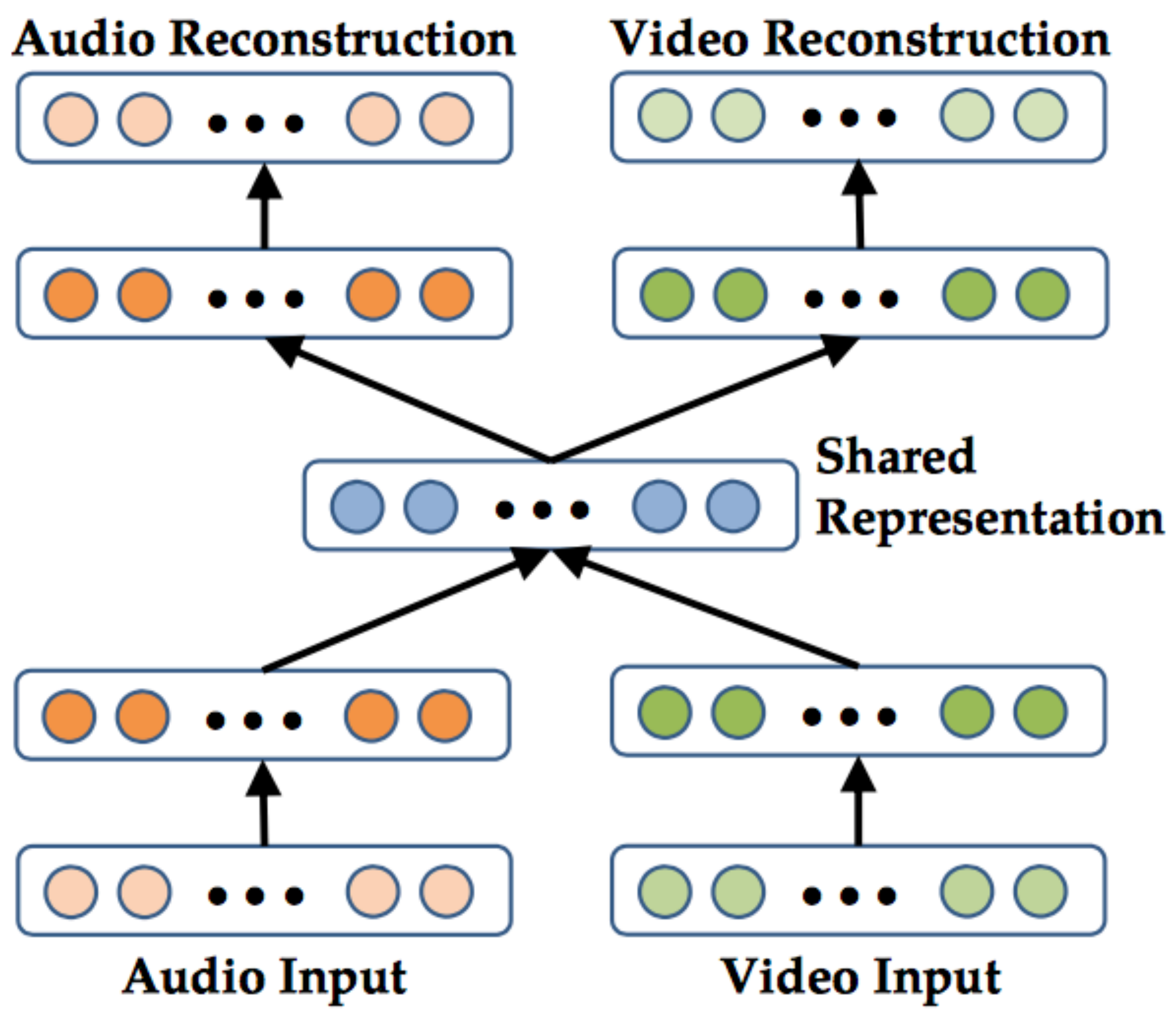
— using denoising autoencoder

To train the model, **reconstruct both modalities** using

— both Audio & Video

— just Audio

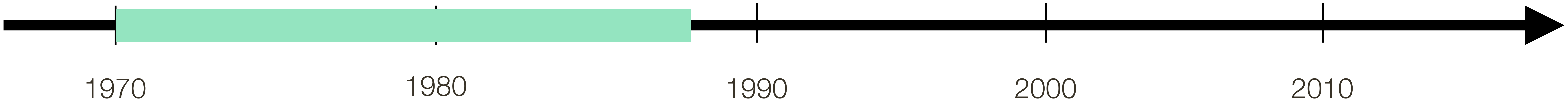
— just Video



*slide from Louis-Philippe Morency

Multimodal Research: Historical Perspective

McGurk Effect (1976)

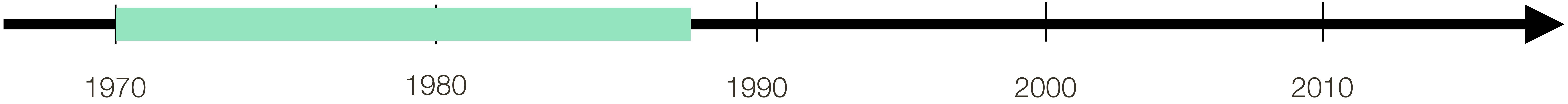


* video credit: **OK Science**

* Adopted from slides by Louis-Philippe Morency

Multimodal Research: Historical Perspective

McGurk Effect (1976)



* video credit: **OK Science**

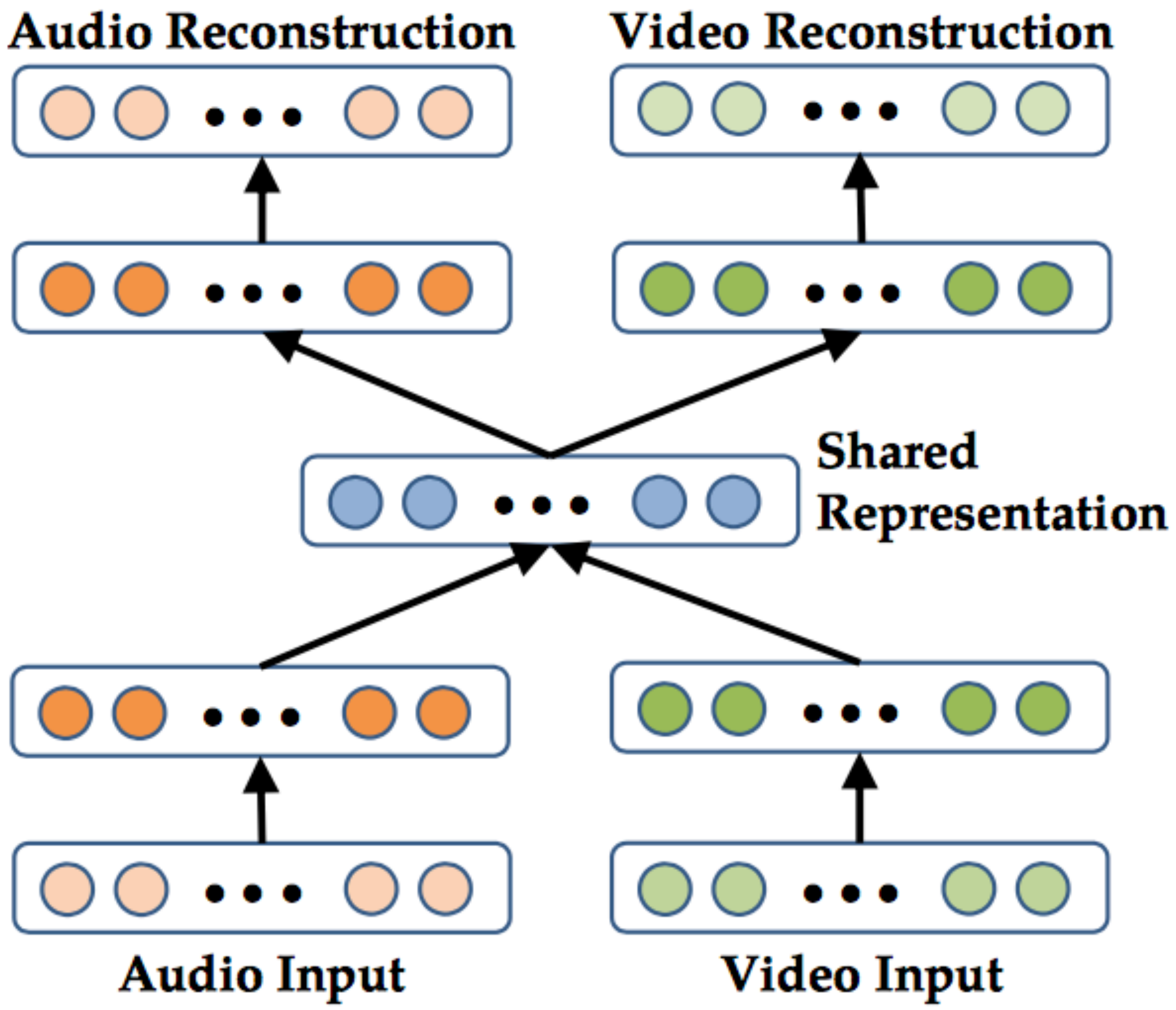
* Adopted from slides by Louis-Philippe Morency

Joint Representation: Deep Multimodal Autoencoders

[Ngiam et al., 2011]

Table 3: McGurk Effect

Audio / Visual Setting	**Model prediction**		
	/ga/	*/ba/*	*/da/*
Visual */ga/*, Audio */ga/*	82.6%	2.2%	15.2%
Visual */ba/*, Audio */ba/*	4.4%	89.1%	6.5%
Visual */ga/*, Audio */ba/*	28.3%	13.0%	58.7%



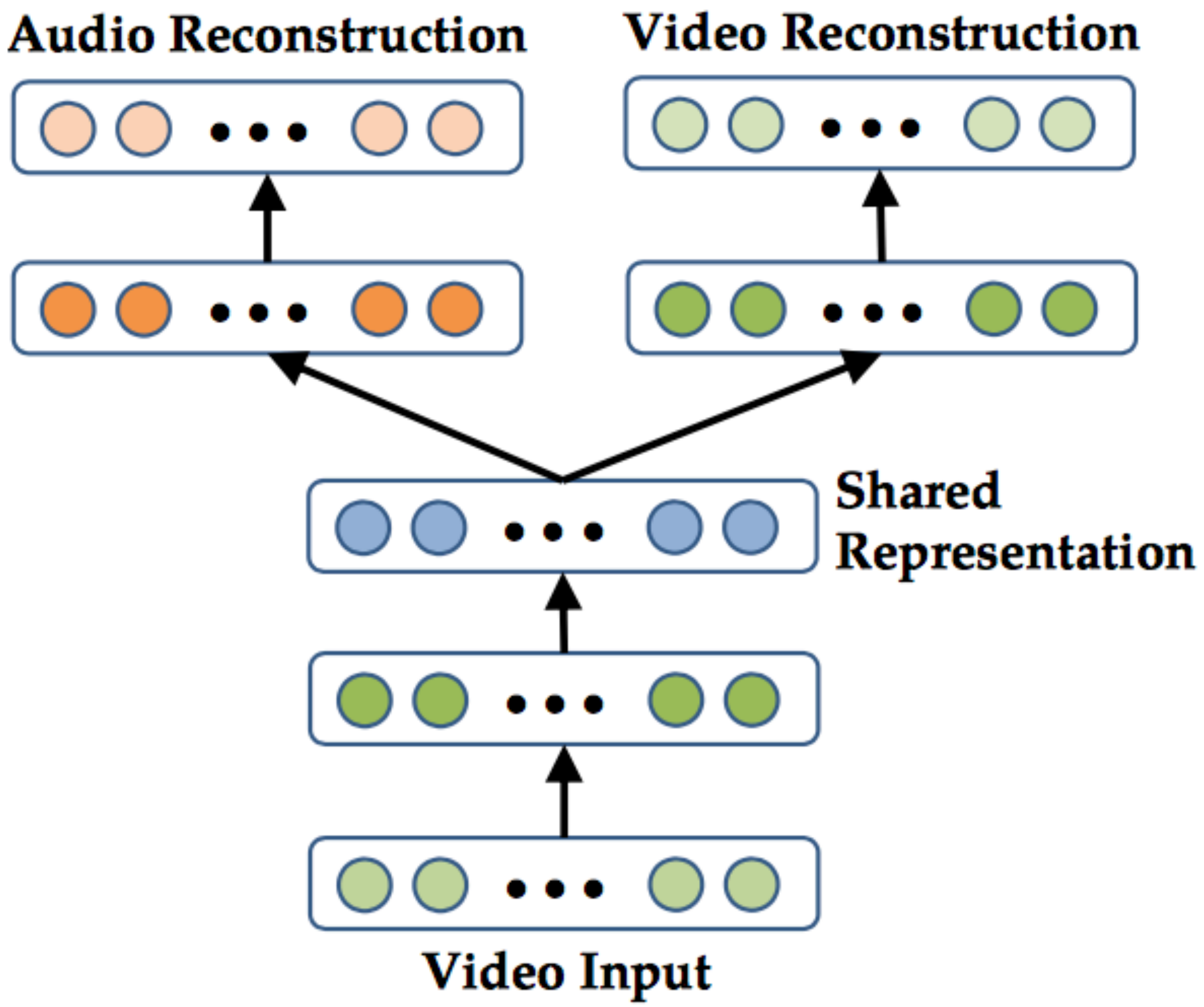
*slide from Louis-Philippe Morency

Joint Representation: Deep Multimodal Autoencoders

[Ngiam et al., 2011]

Useful when you know you may only be conditioning on one modality at test time

Can be regarded as a form of **regularization**



*slide from Louis-Philippe Morency

Supervised Joint Representation

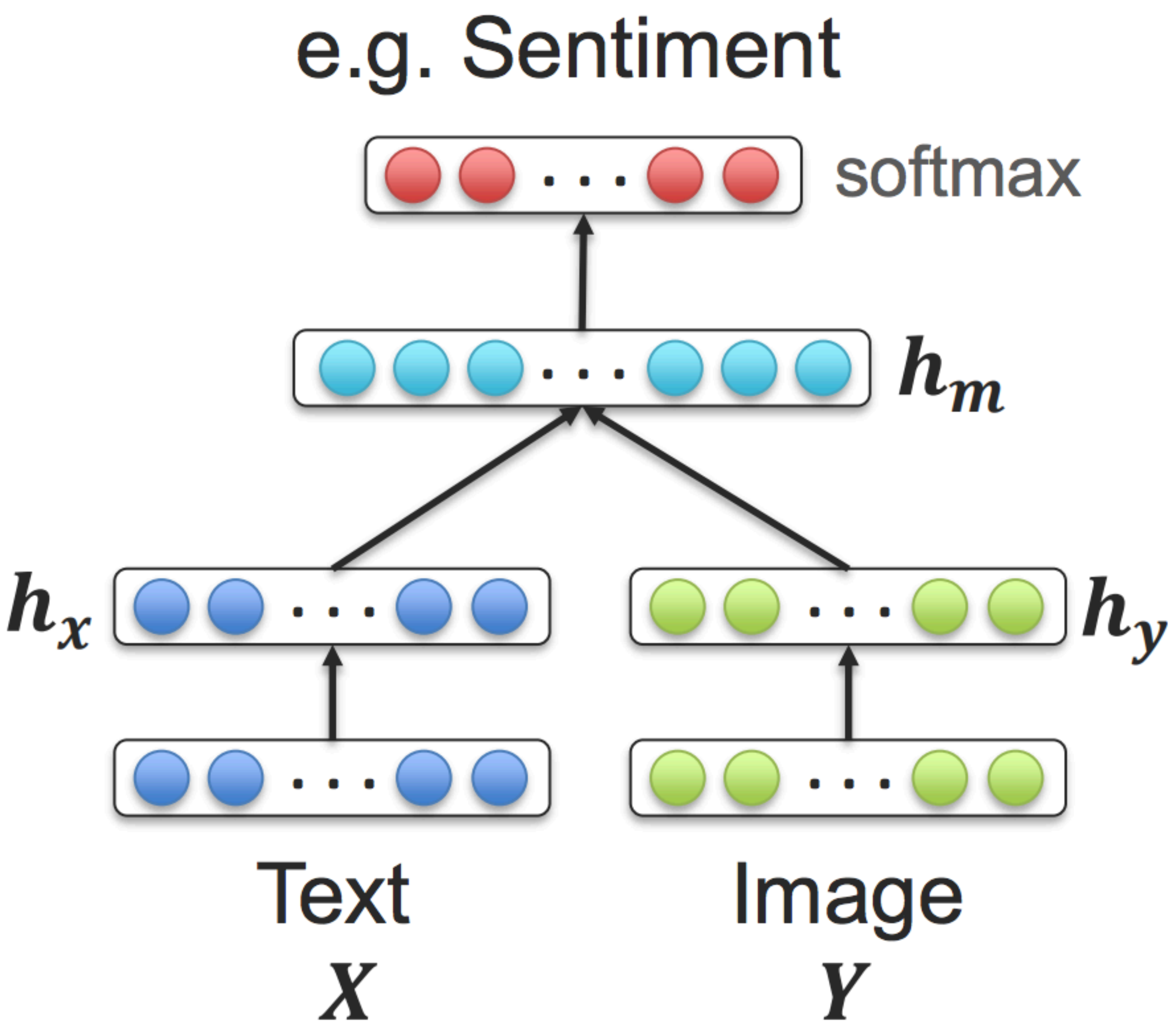
For supervised leaning tasks, we need to join unimodal representations

— Simple **concatenation**

— Element-wise **multiplicative** interactions

— many many others

Encoder-decoder Architectures



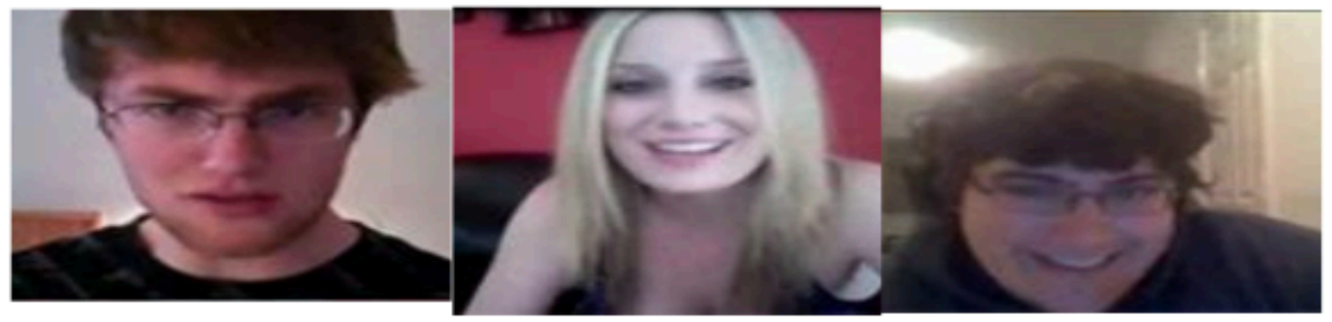
*slide from Louis-Philippe Morency

Multi-modal Sentiment Analysis

For supervised leaning tasks, we need to join unimodal representations

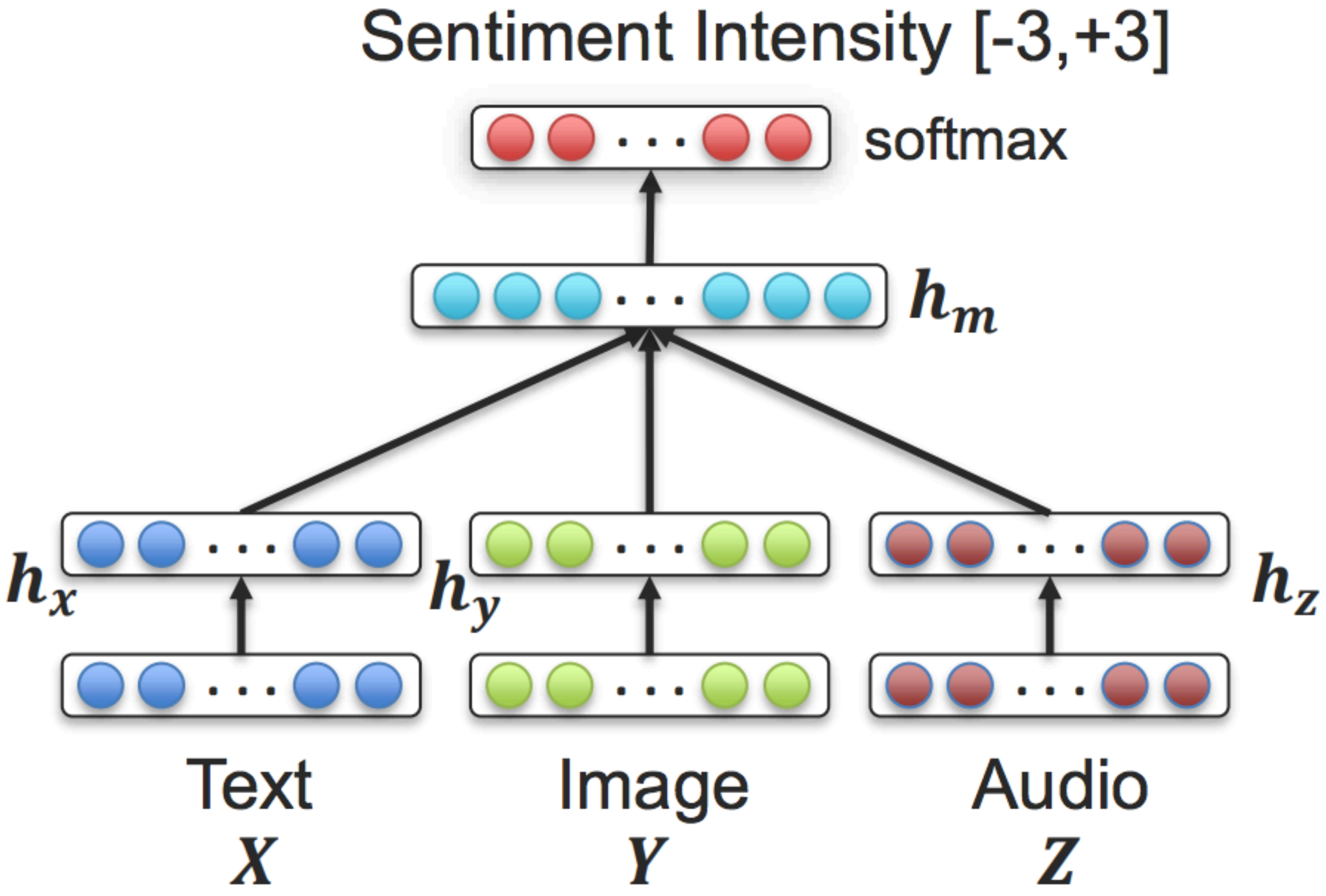
— Simple **concatenation**

MOSI dataset (Zadeh et al, 2016)



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

$$\mathbf{h}_m = \sigma(\mathbf{W} \cdot [\mathbf{h}_x, \mathbf{h}_y, \mathbf{h}_z]^T)$$



*slide from Louis-Philippe Morency

Bilinear Pooling

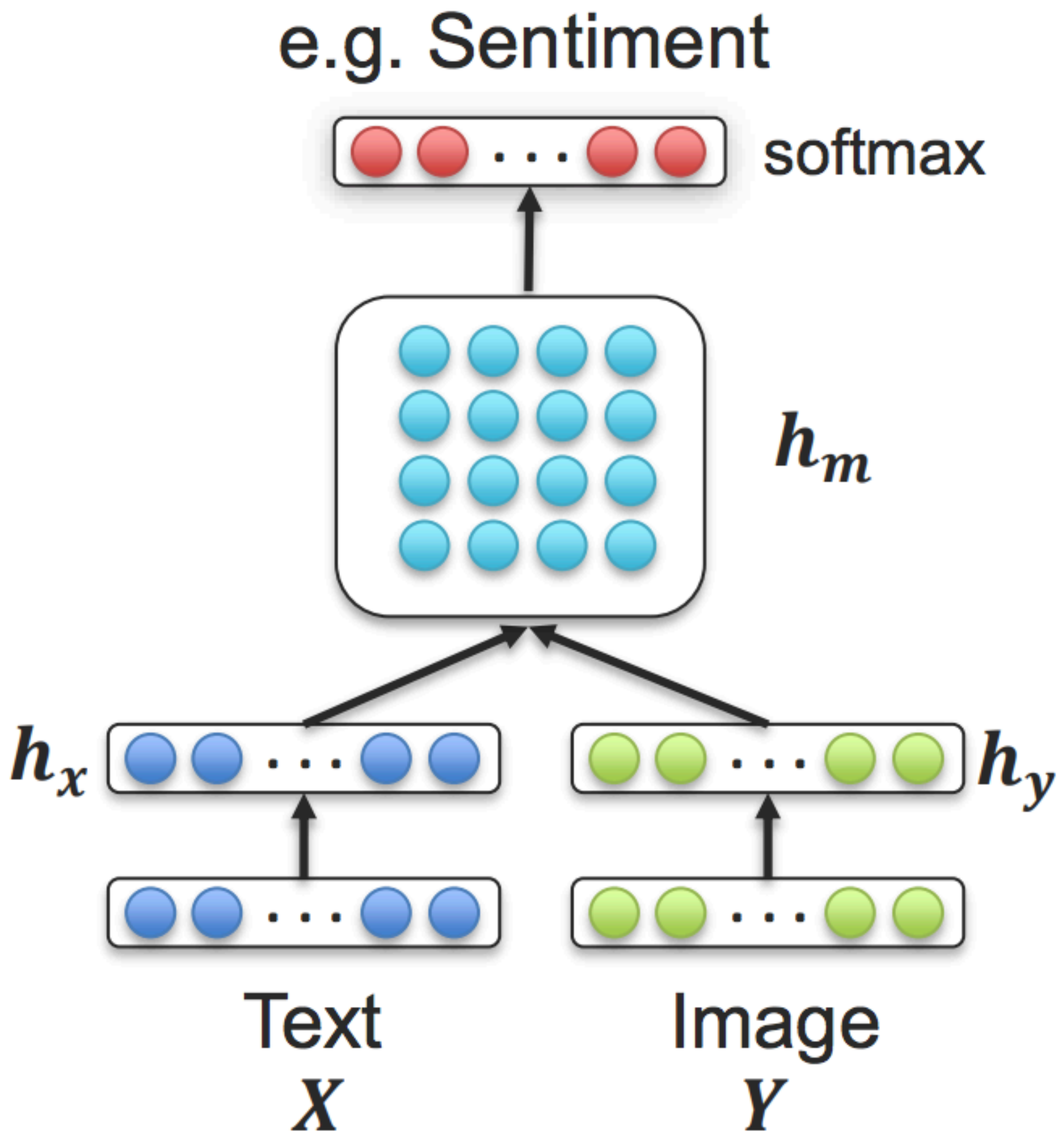
For supervised leaning tasks, we need to join unimodal representations

— Simple **concatenation**

— Element-wise **multiplicative** interactions

$$\mathbf{h}_m = \mathbf{h}_x \otimes \mathbf{h}_y$$

[Tenenbaum and Freeman, 2000]



*slide from Louis-Philippe Morency

Multimodal Tensor Fusion Network (TFN)

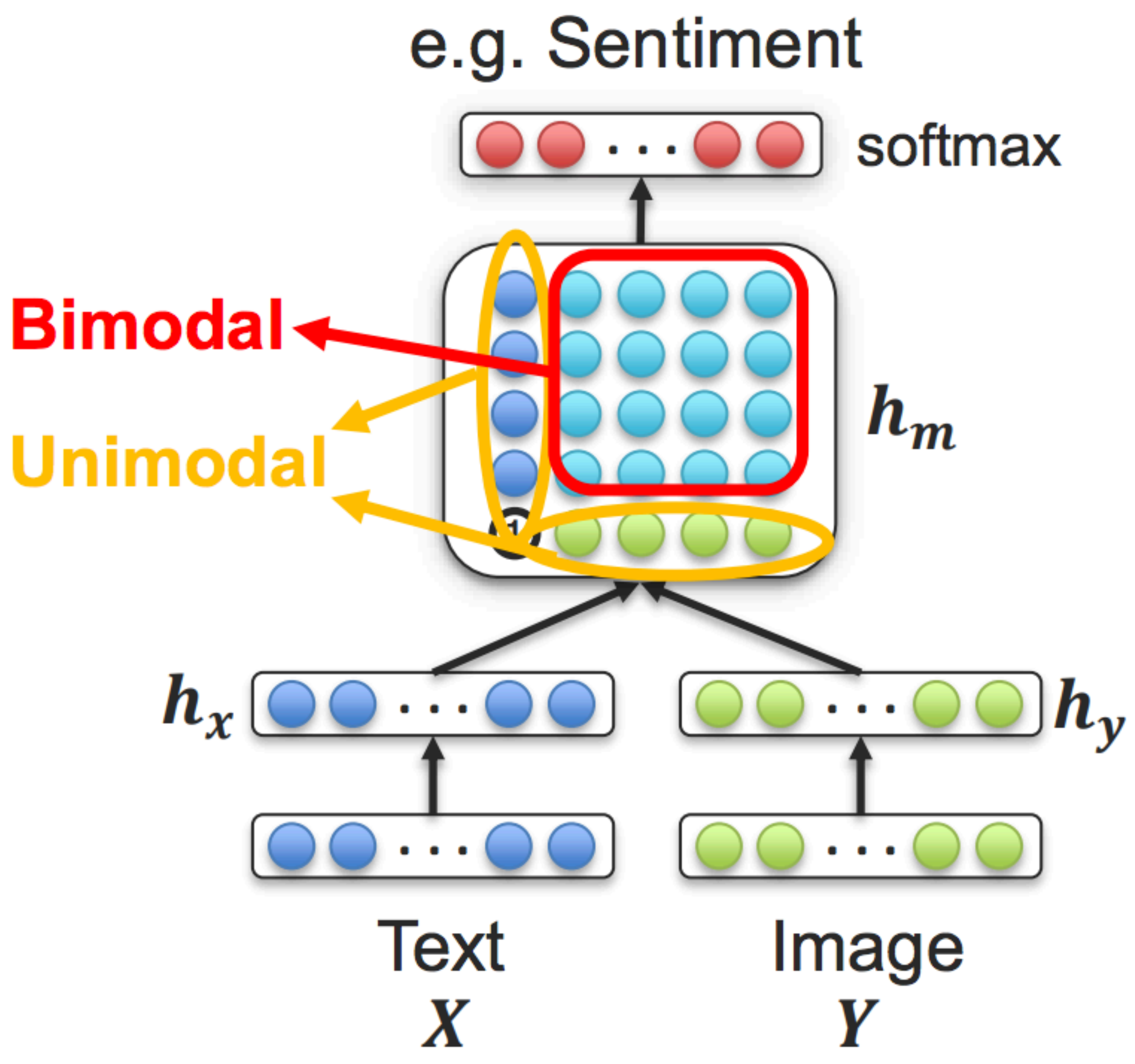
For supervised leaning tasks, we need to join unimodal representations

— Simple **concatenation**

— Element-wise **multiplicative** interactions

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

[Zadeh, Jones and Morency, EMNLP 2017]



*slide from Louis-Philippe Morency

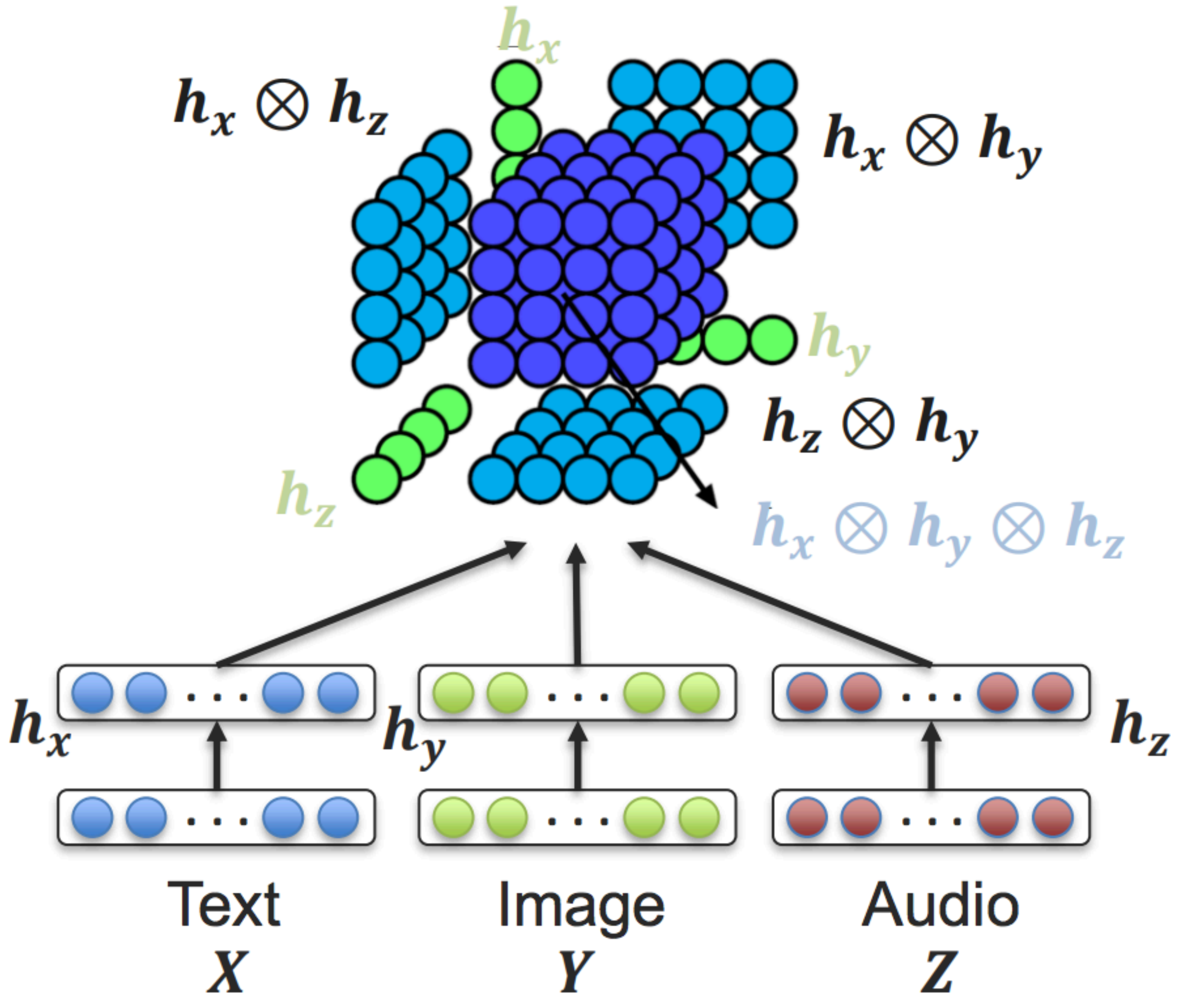
Multimodal Tensor Fusion Network (TFN)

For supervised leaning tasks, we need to join unimodal representations

— Simple **concatenation**

— Element-wise **multiplicative** interactions

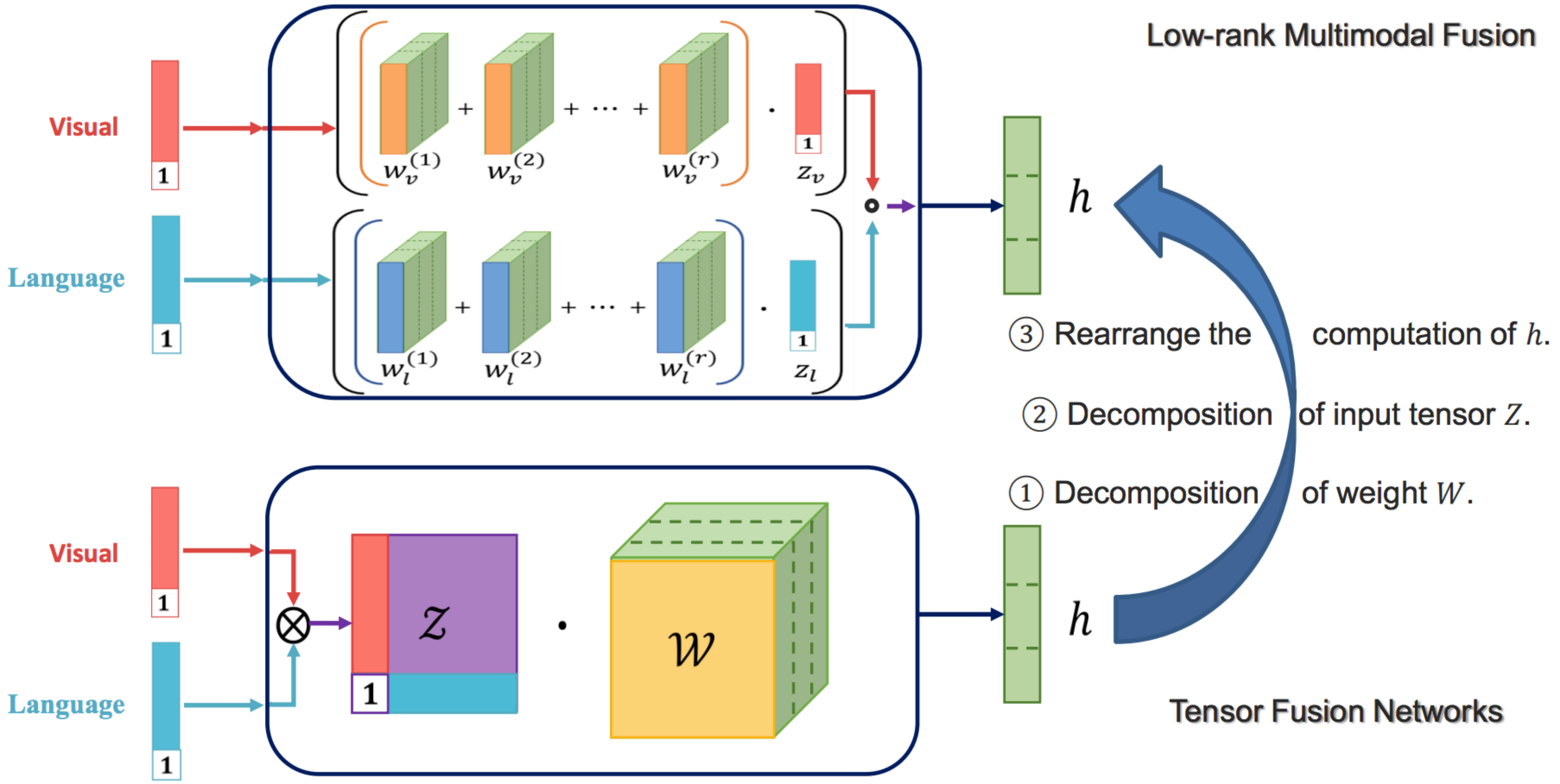
$$\mathbf{h}_m = \begin{bmatrix} \mathbf{h}_x \\ 1 \end{bmatrix} \otimes \begin{bmatrix} \mathbf{h}_y \\ 1 \end{bmatrix} \otimes \begin{bmatrix} \mathbf{h}_z \\ 1 \end{bmatrix}$$



[Zadeh, Jones and Morency, EMNLP 2017]

*slide from Louis-Philippe Morency

Low-rank Tensor Fusion



Tucker tensor decomposition leads to MUTAN fusion

[Ben-younes et al., ICCV 2017]

*slide from Louis-Philippe Morency

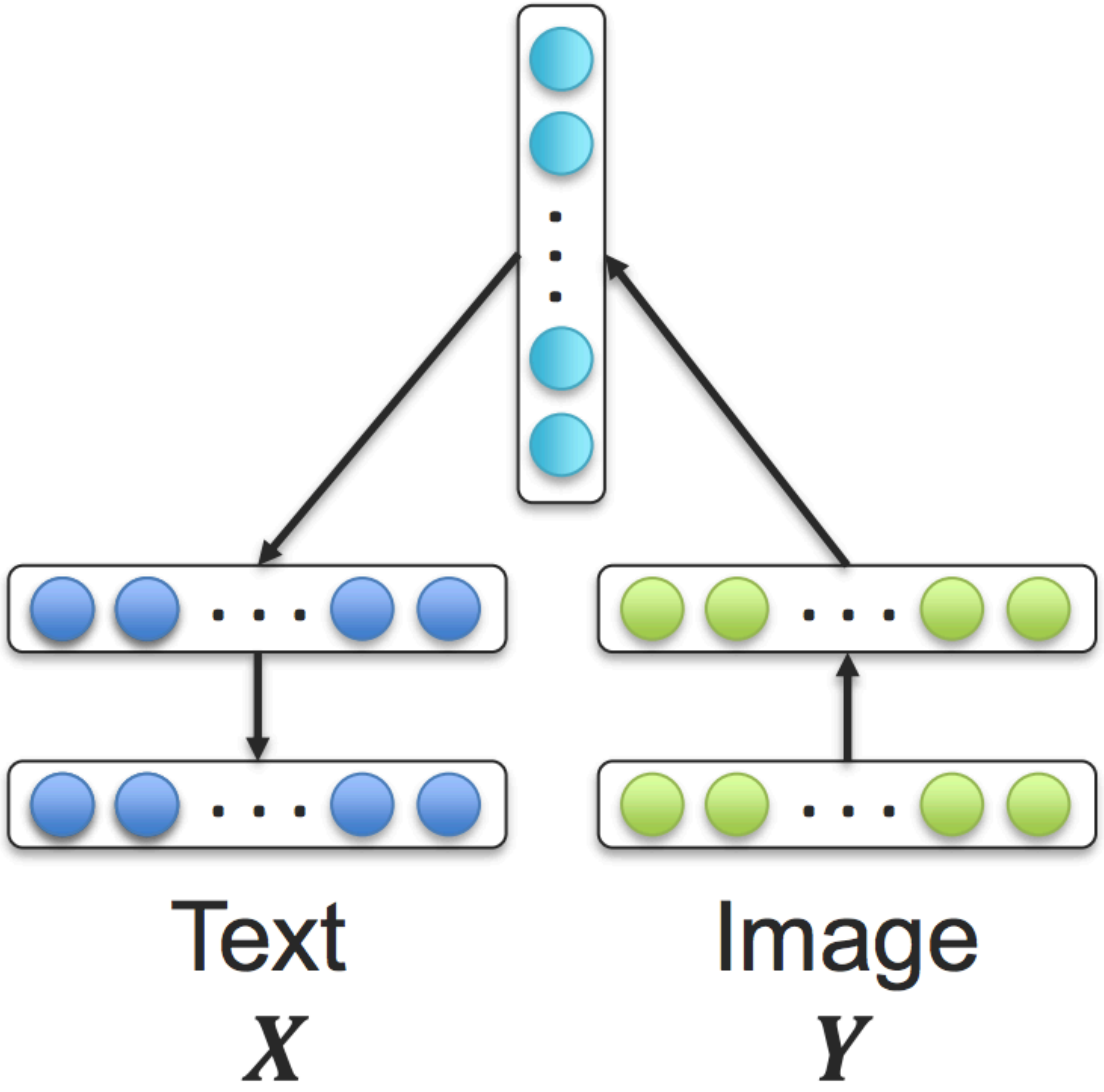
Supervised Joint Representation

For supervised leaning tasks, we need to join unimodal representations

— Simple **concatenation**

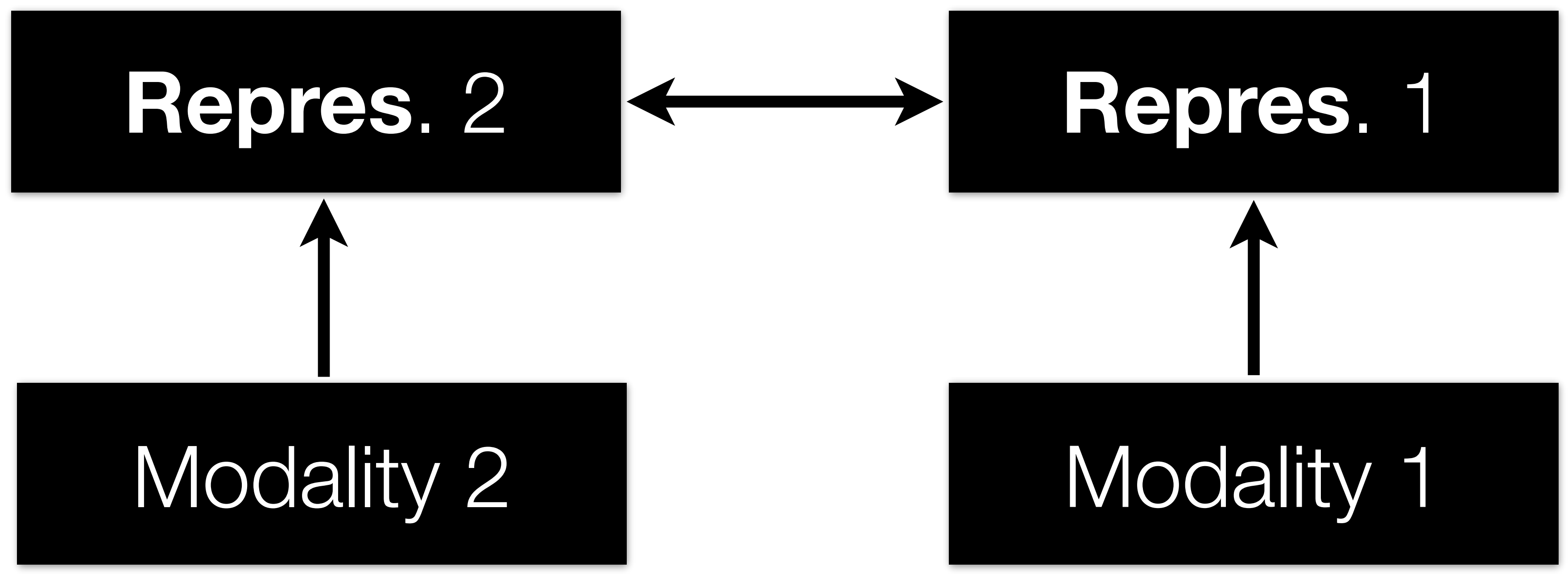
— Element-wise **multiplicative** interactions

Encoder-decoder Architectures



Multimodal Representation Types

Coordinated representations:



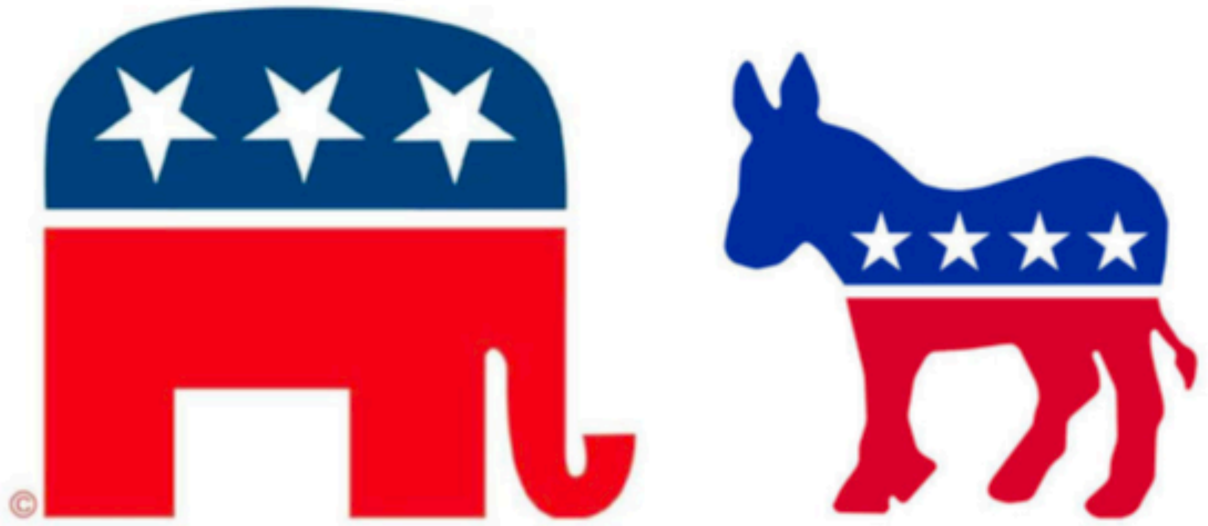
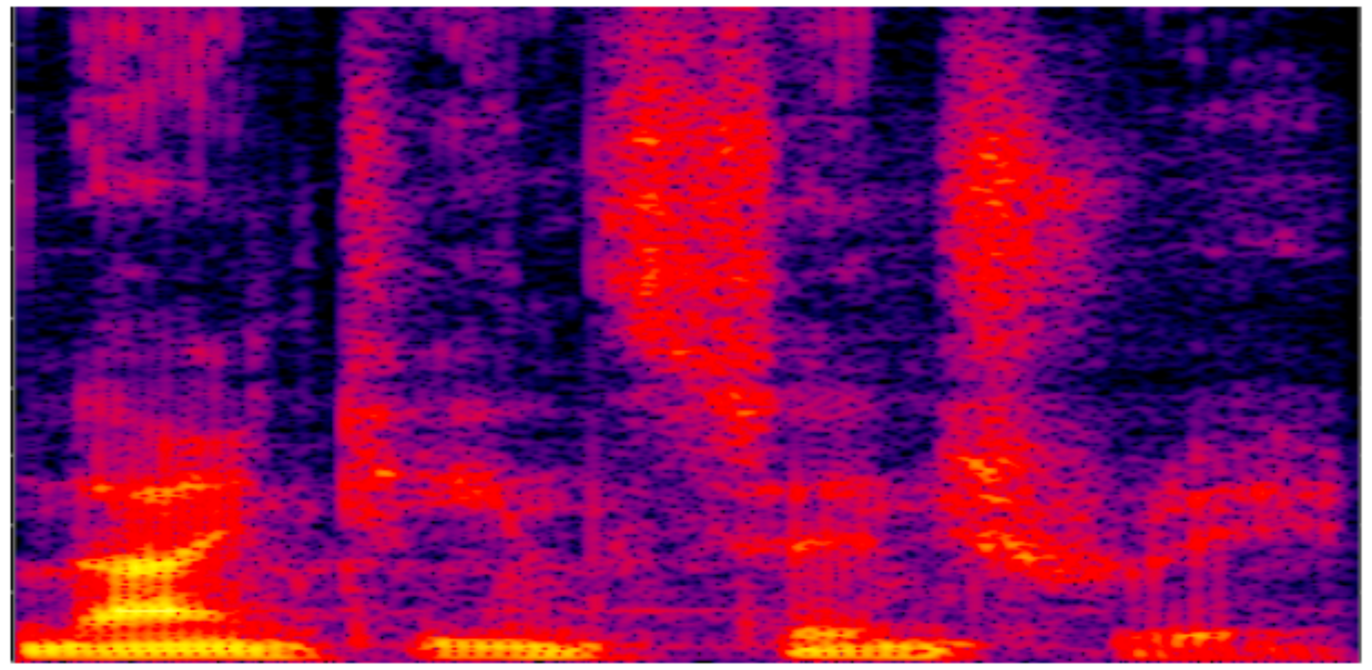
— **Similarity-based** methods (e.g., cosine distance)

— **Structure constraints** (e.g., orthogonality, sparseness)

— Examples: CCA, joint embeddings

*slide from Louis-Philippe Morency

Data with **Multiple Views**

$x_1^{(i)}$	$x_2^{(i)}$
 demographic properties	 responses to survey
 audio features at time i	 video features at time i

*slide from Andrew, Arora, Bilmes, Livescu

Correlated Representations

Goal: Find representations $f_1(\mathbf{x}_1), f_2(\mathbf{x}_2)$ for each view that maximize correlation:

$$\mathbf{corr}(f_1(\mathbf{x}_1), f_2(\mathbf{x}_2)) = \frac{\mathbf{cov}(f_1(\mathbf{x}_1), f_2(\mathbf{x}_2))}{\sqrt{\mathbf{var}(f_1(\mathbf{x}_1)) \cdot \mathbf{var}(f_2(\mathbf{x}_2))}}$$

*slide from Andrew, Arora, Bilmes, Livescu

Correlated Representations

Goal: Find representations $f_1(\mathbf{x}_1), f_2(\mathbf{x}_2)$ for each view that maximize correlation:

$$\mathbf{corr}(f_1(\mathbf{x}_1), f_2(\mathbf{x}_2)) = \frac{\mathbf{cov}(f_1(\mathbf{x}_1), f_2(\mathbf{x}_2))}{\sqrt{\mathbf{var}(f_1(\mathbf{x}_1)) \cdot \mathbf{var}(f_2(\mathbf{x}_2))}}$$

Finding correlated representations can be **useful** for

- Gaining insights into the data
- Detecting of asynchrony in test data
- Removing noise uncorrelated across views
- Translation or retrieval across views

*slide from Andrew, Arora, Bilmes, Livescu

Correlated Representations

Goal: Find representations $f_1(\mathbf{x}_1), f_2(\mathbf{x}_2)$ for each view that maximize correlation:

$$\mathbf{corr}(f_1(\mathbf{x}_1), f_2(\mathbf{x}_2)) = \frac{\mathbf{cov}(f_1(\mathbf{x}_1), f_2(\mathbf{x}_2))}{\sqrt{\mathbf{var}(f_1(\mathbf{x}_1)) \cdot \mathbf{var}(f_2(\mathbf{x}_2))}}$$

Finding correlated representations can be **useful** for

— Gaining insights into the data

— Detecting of asynchrony in test data

— Removing noise uncorrelated across views

— Translation or retrieval across views

Has been **applied widely** to problems in computer vision, speech, NLP, medicine, chemometrics, metrology, neurology, etc.

*slide from Andrew, Arora, Bilmes, Livescu

CCA: Canonical Correlation Analysis

Classical technique to find **linear** correlated representations, i.e.,

$$f_1(\mathbf{x}_1) = \mathbf{W}_1^T \mathbf{x}_1 \qquad \mathbf{W}_1 \in \mathbb{R}^{d_1 \times k}$$

where

$$f_2(\mathbf{x}_2) = \mathbf{W}_2^T \mathbf{x}_2 \qquad \mathbf{W}_2 \in \mathbb{R}^{d_2 \times k}$$

*slide from Andrew, Arora, Bilmes, Livescu

CCA: Canonical Correlation Analysis

Classical technique to find **linear** correlated representations, i.e.,

$$f_1(\mathbf{x}_1) = \mathbf{W}_1^T \mathbf{x}_1 \qquad \text{where} \qquad \mathbf{W}_1 \in \mathbb{R}^{d_1 \times k}$$

$$f_2(\mathbf{x}_2) = \mathbf{W}_2^T \mathbf{x}_2 \qquad \qquad \qquad \mathbf{W}_2 \in \mathbb{R}^{d_2 \times k}$$

The first columns $(\mathbf{w}_{1,:1}, \mathbf{w}_{2,:1})$ of the matrices $\mathbf{W}_1$ and $\mathbf{W}_2$ are found to maximize the **correlation of the projections**:

$$(\mathbf{w}_{1,:1}, \mathbf{w}_{2,:1}) = \arg\max \mathbf{corr}(\mathbf{w}_{1,:1}^T \mathbf{X}_1, \mathbf{w}_{2,:1}^T \mathbf{X}_2)$$

*slide from Andrew, Arora, Bilmes, Livescu

CCA: Canonical Correlation Analysis

Classical technique to find **linear** correlated representations, i.e.,

$$f_1(\mathbf{x}_1) = \mathbf{W}_1^T \mathbf{x}_1 \qquad \mathbf{W}_1 \in \mathbb{R}^{d_1 \times k}$$

where

$$f_2(\mathbf{x}_2) = \mathbf{W}_2^T \mathbf{x}_2 \qquad \mathbf{W}_2 \in \mathbb{R}^{d_2 \times k}$$

The first columns $(\mathbf{w}_{1,:1}, \mathbf{w}_{2,:1})$ of the matrices $\mathbf{W}_1$ and $\mathbf{W}_2$ are found to maximize the **correlation of the projections**:

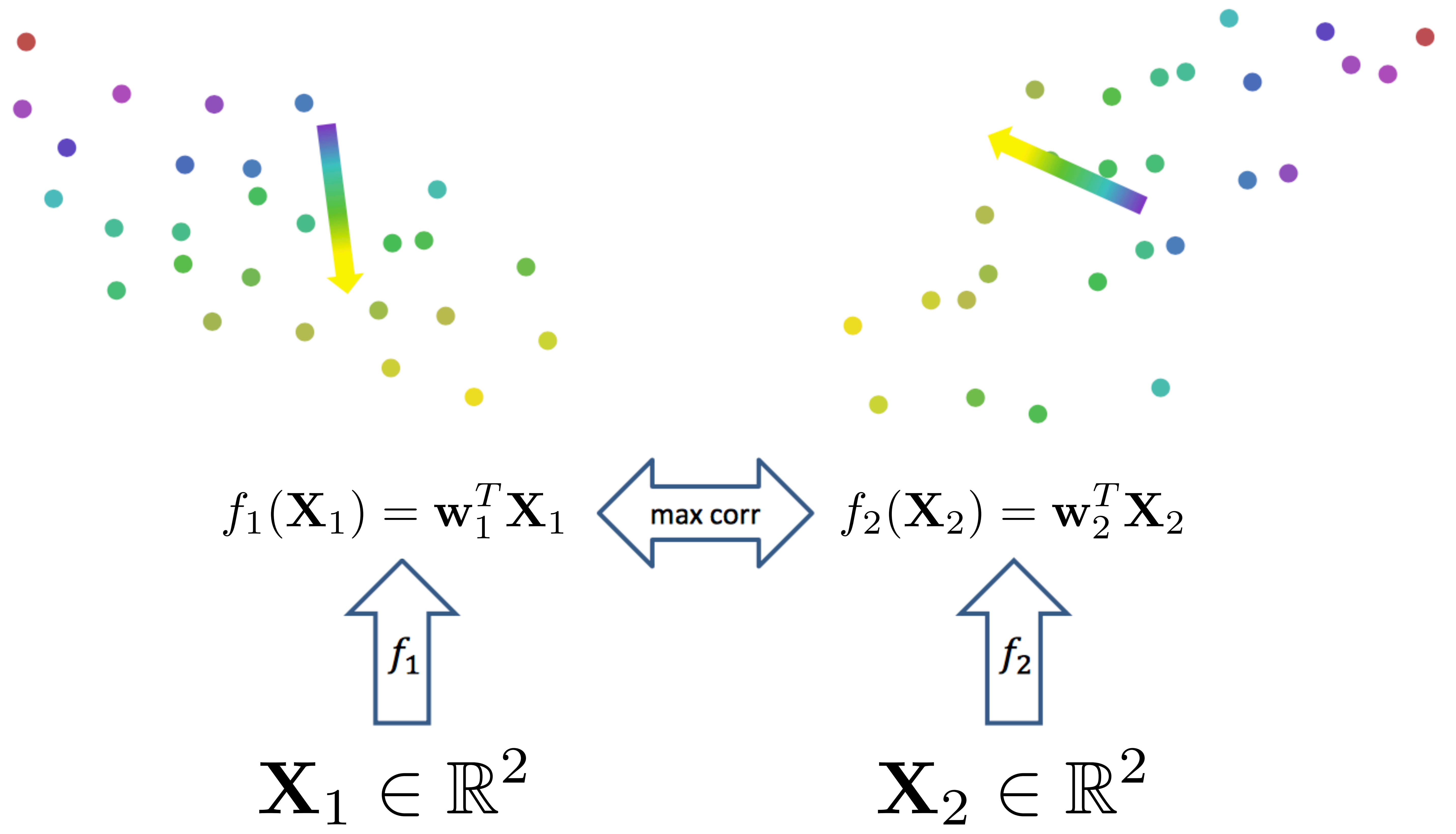
$$(\mathbf{w}_{1,:1}, \mathbf{w}_{2,:1}) = \arg\max \mathbf{corr}(\mathbf{w}_{1,:1}^T \mathbf{X}_1, \mathbf{w}_{2,:1}^T \mathbf{X}_2)$$

Subsequent pairs are constrained to be **uncorrelated with previous components** (i.e., for $j < i$)

$$\mathbf{corr}(\mathbf{w}_{1,:i}^T \mathbf{X}_1, \mathbf{w}_{1,:j}^T \mathbf{X}_1) = \mathbf{corr}(\mathbf{w}_{2,:i}^T \mathbf{X}_2, \mathbf{w}_{2,:j}^T \mathbf{X}_2) = 0$$

*slide from Andrew, Arora, Bilmes, Livescu

CCA Illustration



Two views of each instance have the same color

*slide from Andrew, Arora, Bilmes, Livescu

CCA: Canonical Correlation Analysis

1. Estimate **covariance matrix** with regularization:

$$\Sigma_{11} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)^T + r_1\mathbf{I}$$

$$\Sigma_{12} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)^T$$

$$\Sigma_{12} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)^T$$

$$\Sigma_{22} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)^T + r_2\mathbf{I}$$

*slide from Andrew, Arora, Bilmes, Livescu

CCA: Canonical Correlation Analysis

1. Estimate **covariance matrix** with regularization:

$$\Sigma_{11} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)^T + r_1\mathbf{I}$$

$$\Sigma_{12} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)^T$$

$$\Sigma_{12} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)^T$$

$$\Sigma_{22} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)^T + r_2\mathbf{I}$$

$$\Sigma = \left[\begin{array}{c|c} \Sigma_{11} & \Sigma_{12} \\ \hline \Sigma_{12} & \Sigma_{22} \end{array}\right] \quad \xRightarrow{\mathbf{W}_1^* \quad \mathbf{W}_2^*} \quad \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & \lambda_1 & 0 & 0 \\ 0 & 1 & 0 & 0 & \lambda_2 & 0 \\ 0 & 0 & 1 & 0 & 0 & \lambda_3 \\ \hline \lambda_1 & 0 & 0 & 1 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 & 1 & 0 \\ 0 & 0 & \lambda_3 & 0 & 0 & 1 \end{array}\right]$$

CCA: Canonical Correlation Analysis

1. Estimate **covariance matrix** with regularization:

$$\Sigma_{11} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)^T + r_1\mathbf{I} \qquad \Sigma_{12} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)^T$$

$$\Sigma_{12} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)^T \qquad \Sigma_{22} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)^T + r_2\mathbf{I}$$

2. Form **normalized covariance** matrix: $\mathbf{T} = \Sigma_{11}^{-1/2}\Sigma_{12}\Sigma_{22}^{-1/2}$ and its singular value decomposition $\mathbf{T} = \mathbf{UDV}^T$

*slide from Andrew, Arora, Bilmes, Livescu

CCA: Canonical Correlation Analysis

1. Estimate **covariance matrix** with regularization:

$$\Sigma_{11} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)^T + r_1\mathbf{I} \qquad \Sigma_{12} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)^T$$

$$\Sigma_{12} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)^T \qquad \Sigma_{22} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)^T + r_2\mathbf{I}$$

2. Form **normalized covariance** matrix: $\mathbf{T} = \Sigma_{11}^{-1/2}\Sigma_{12}\Sigma_{22}^{-1/2}$ and its singular value decomposition $\mathbf{T} = \mathbf{UDV}^T$

3. **Total correlation** at k is $\sum_{i=1}^{k} D_{ii}$

*slide from Andrew, Arora, Bilmes, Livescu

CCA: Canonical Correlation Analysis

1. Estimate **covariance matrix** with regularization:

$$\Sigma_{11} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)^T + r_1\mathbf{I} \qquad \Sigma_{12} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)^T$$

$$\Sigma_{12} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_1^{(i)} - \bar{\mathbf{x}}_1)(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)^T \qquad \Sigma_{22} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)(\mathbf{x}_2^{(i)} - \bar{\mathbf{x}}_2)^T + r_2\mathbf{I}$$

2. Form **normalized covariance** matrix: $\mathbf{T} = \Sigma_{11}^{-1/2}\Sigma_{12}\Sigma_{22}^{-1/2}$ and its singular value decomposition $\mathbf{T} = \mathbf{U}\mathbf{D}\mathbf{V}^T$

3. **Total correlation** at k is $\sum_{i=1}^{k} D_{ii}$

4. The optimal projection matrices are:

$$\mathbf{W}_1^* = \Sigma_{11}^{-1/2}\mathbf{U}_k$$

$$\mathbf{W}_2^* = \Sigma_{22}^{-1/2}\mathbf{V}_k$$

where $\mathbf{U}_k$ is the first k columns of $\mathbf{U}$.

*slide from Andrew, Arora, Bilmes, Livescu

KCCA: Kernel CCA

There maybe **non-linear** functions $f_1(\mathbf{x}_1), f_2(\mathbf{x}_2)$ that produce more highly correlated (better) representations than linear projections

Kernel CCA is a principal method for finding such function

— Learns functions from any reproducing kernel Hilbert space

— May use different kernels for each view

Using **RBF** (Gaussian) kernel in KCCA is akin to finding sets of instances that form clusters in both views

*slide from Andrew, Arora, Bilmes, Livescu

KCCA vs. CCA

Pros:

— More complex function space of KCCA can yield dramatically higher correlations

Cons:

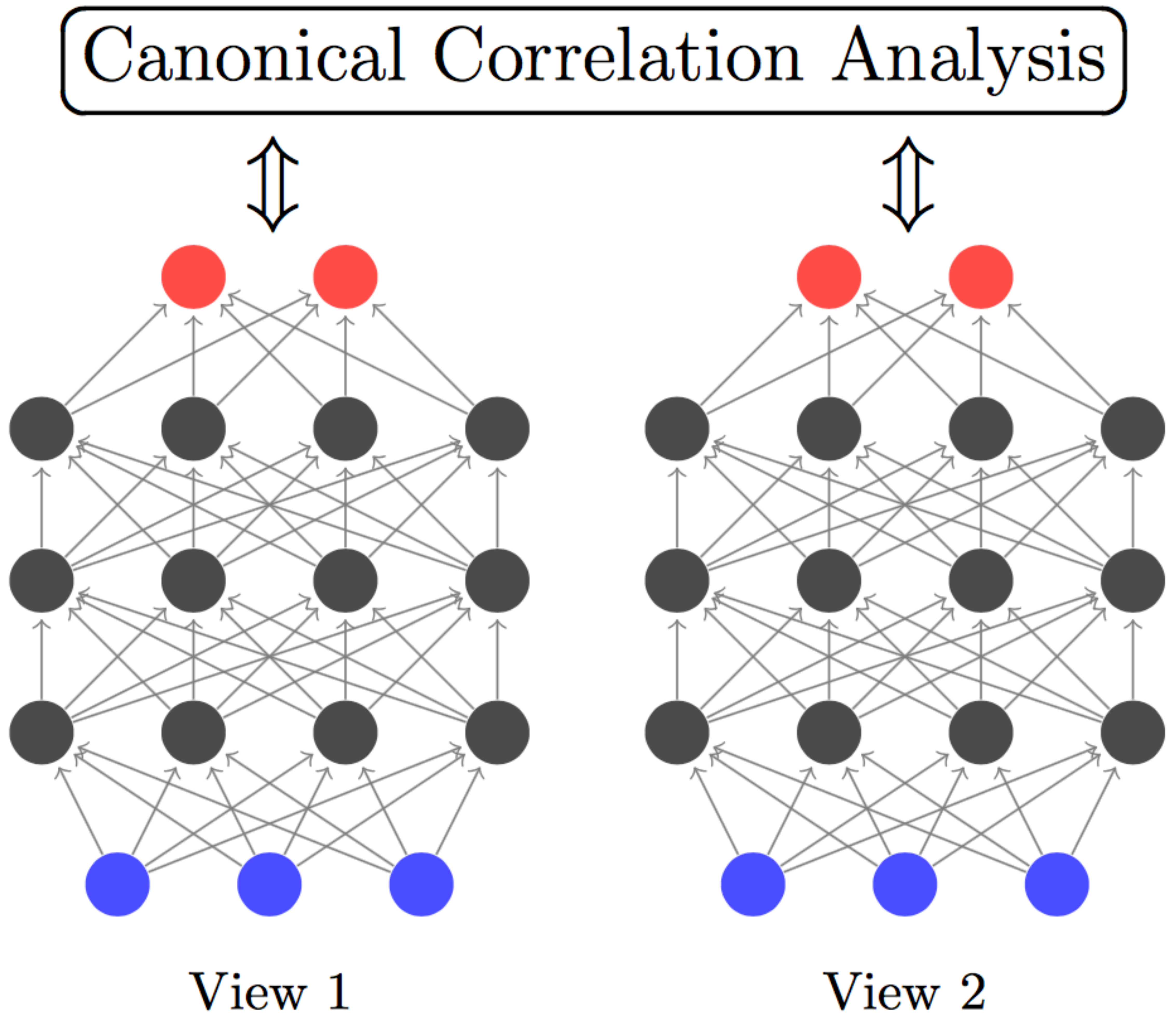
— KCCA is slower to train

— For KCCA training set must be stored and referenced at test time

— KCCA model is more difficult to interpret

*slide from Andrew, Arora, Bilmes, Livescu

Deep CCA



*slide from Andrew, Arora, Bilmes, Livescu

Benefits of Deep CCA

Pros:

- Better suited for natural, real-world data
- **Parametric model**
  - The training set can be disregarded once the model is learned
  - Computational speed at test time is fast

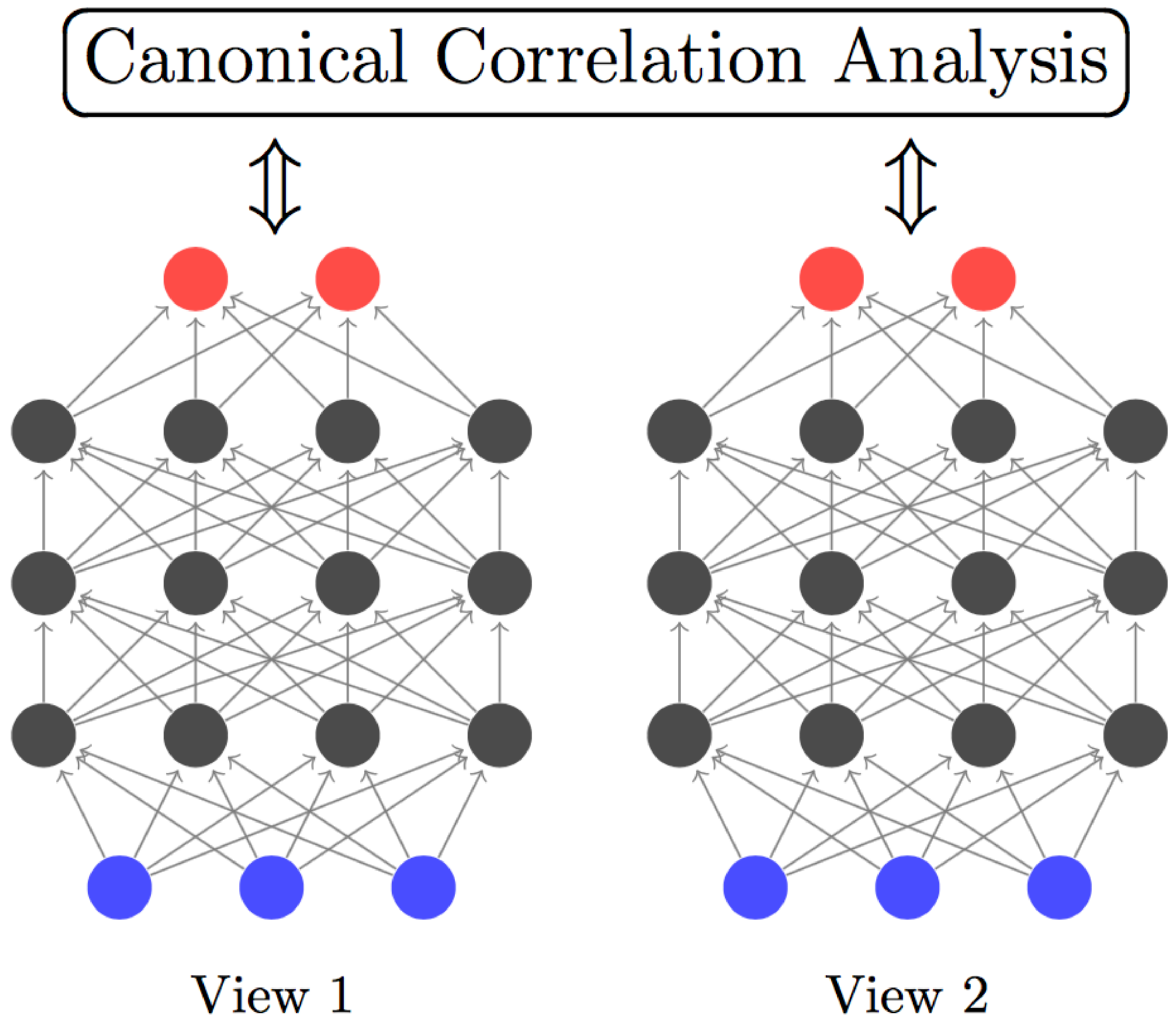
*slide from Andrew, Arora, Bilmes, Livescu

Deep CCA: Training

Training a Deep CCA model:

1. **Pretrain** the layers of **each side** individually

2. **Jointly fine-tune** all parameters to maximize the total correlation of the output layers. Requires computing correlation gradient:
   - Forward propagate activations on both sides.
   - Compute correlation and its gradient w.r.t. output layers.
   - Backpropagate gradient on both sides.



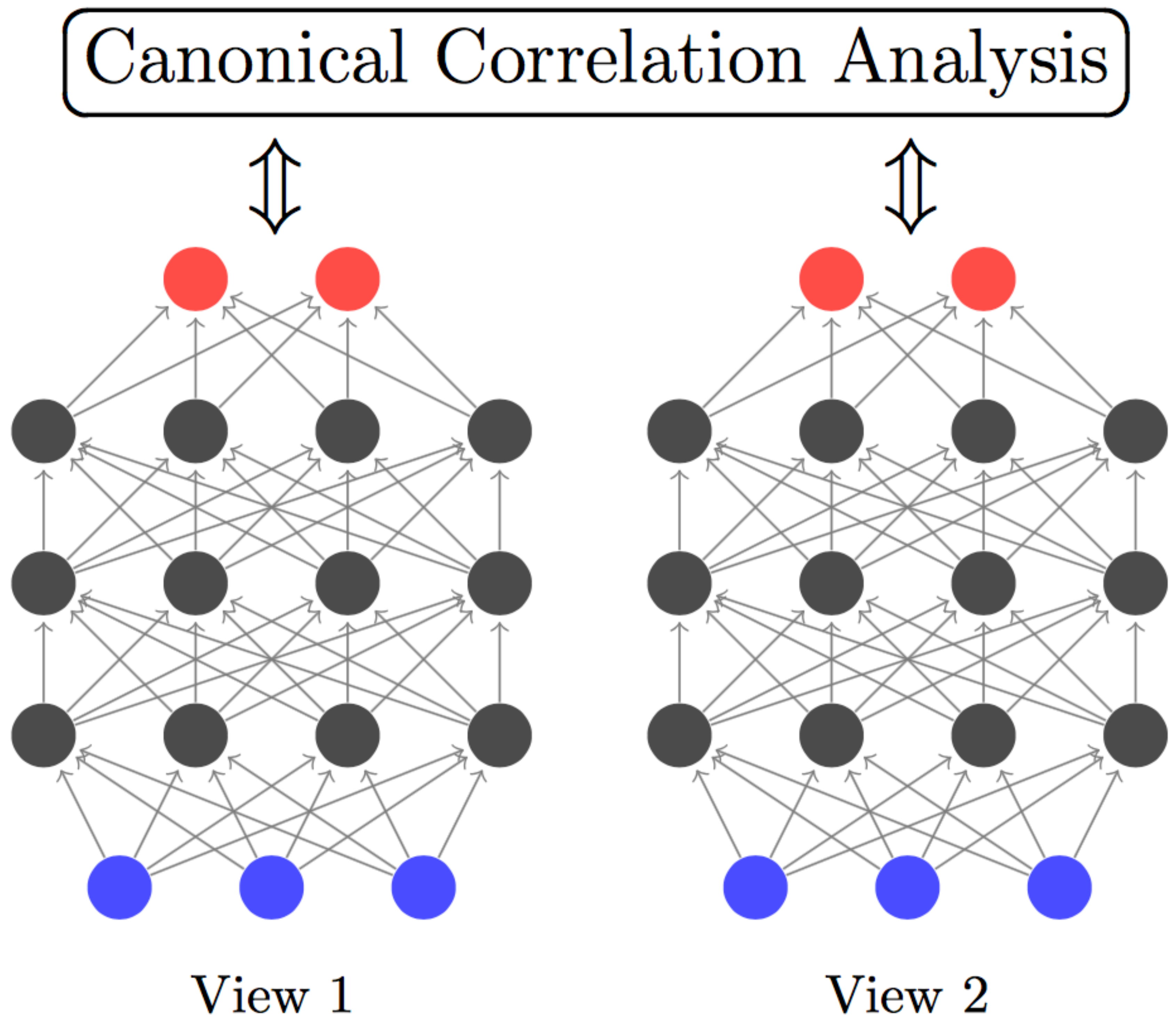
*slide from Andrew, Arora, Bilmes, Livescu

Deep CCA: Training

Training a Deep CCA model:

1. **Pretrain** the layers of **each side** individually

2. **Jointly fine-tune** all parameters to maximize the total correlation of the output layers. Requires computing correlation gradient:
   - Forward propagate activations on both sides.
   - Compute correlation and its gradient w.r.t. output layers.
   - Backpropagate gradient on both sides.



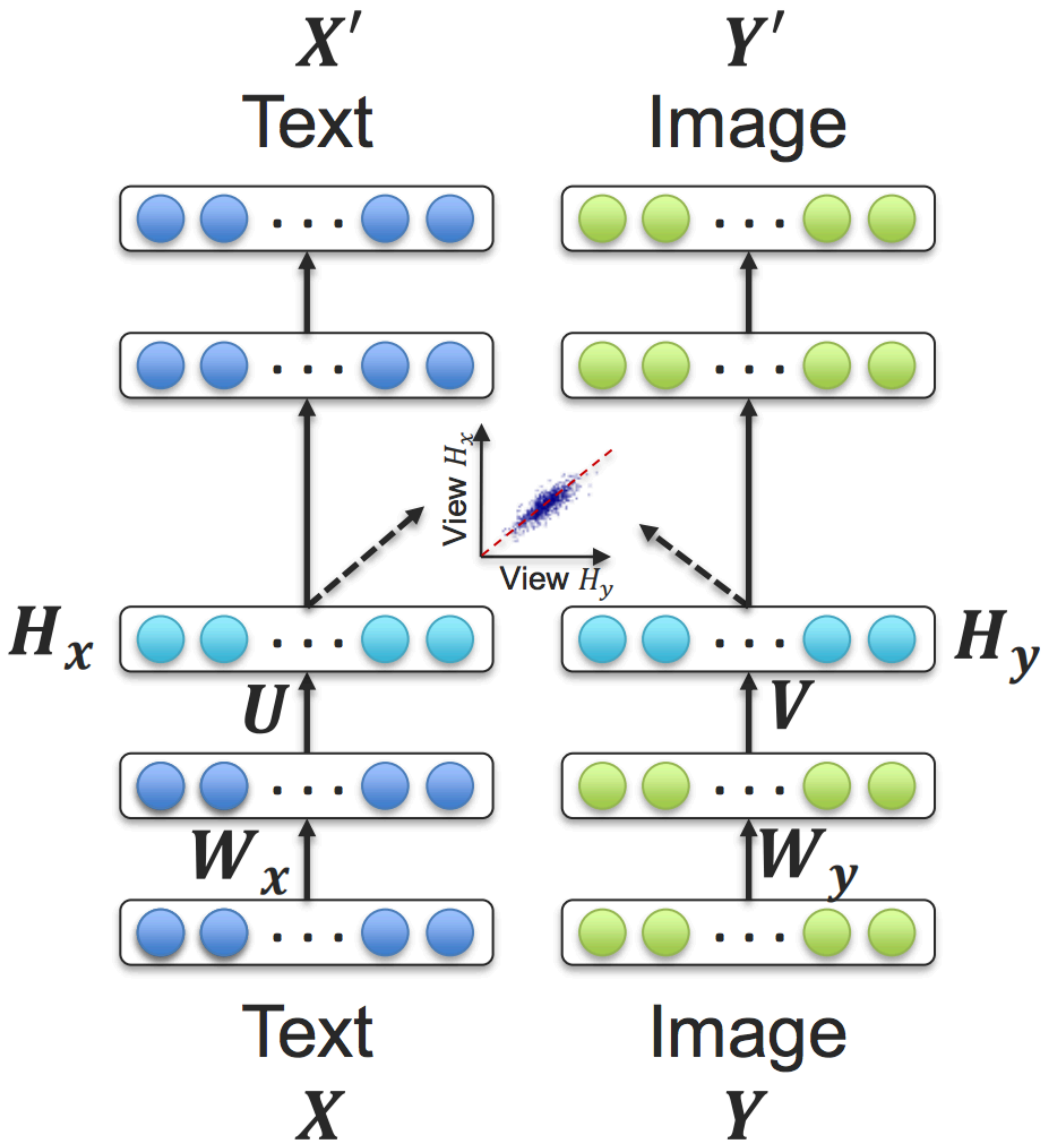
Correlation is a population objective, so instead of one instance (or minibatch) training, requires L-BFGS second-order method (with full-batch)

*slide from Andrew, Arora, Bilmes, Livescu

Deep Canonically Correlated Autoencoders (DCCAE)

Jointly optimize for DCCA and auto encoders loss functions

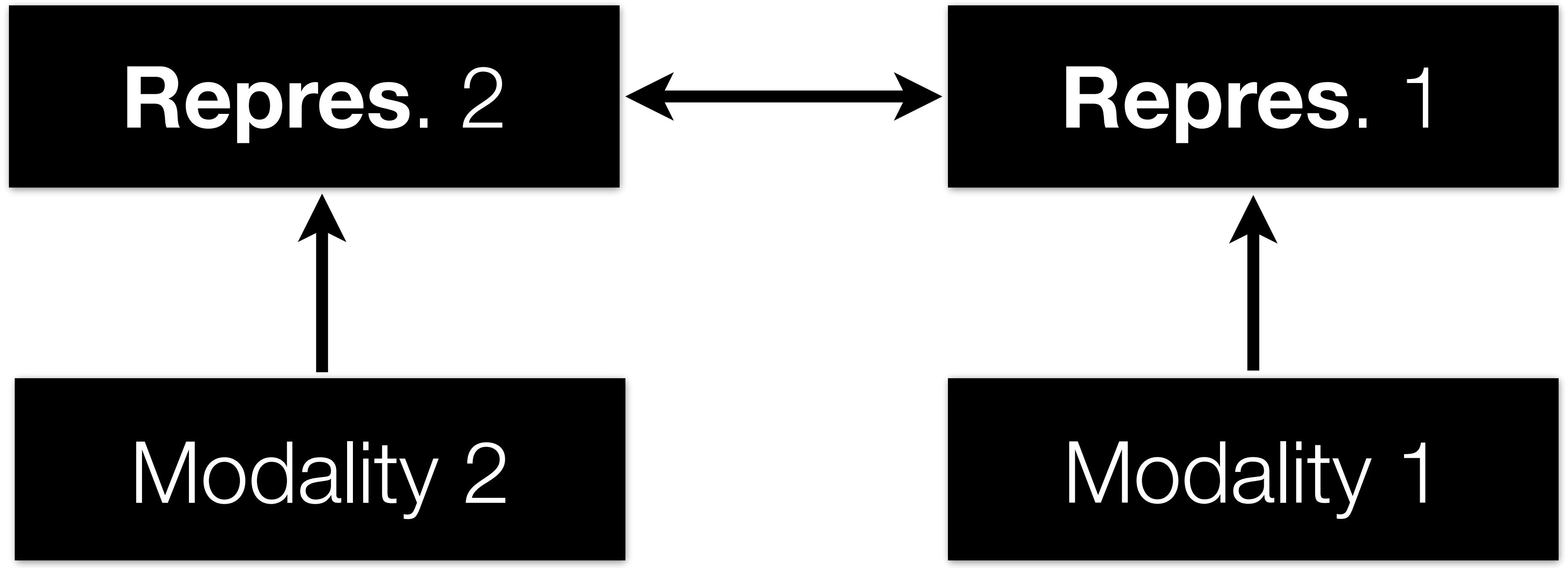
— A trade-off between multi-view correlation and reconstruction error from individual views



[Wang et al., ICML 2015]

Multimodal Representation Types

Coordinated representations:



— **Similarity-based** methods (e.g., cosine distance)

— **Structure constraints** (e.g., orthogonality, sparseness)

— Examples: CCA, joint embeddings

*slide from Louis-Philippe Morency

Correlated Representations vs. **Joint Embeddings**

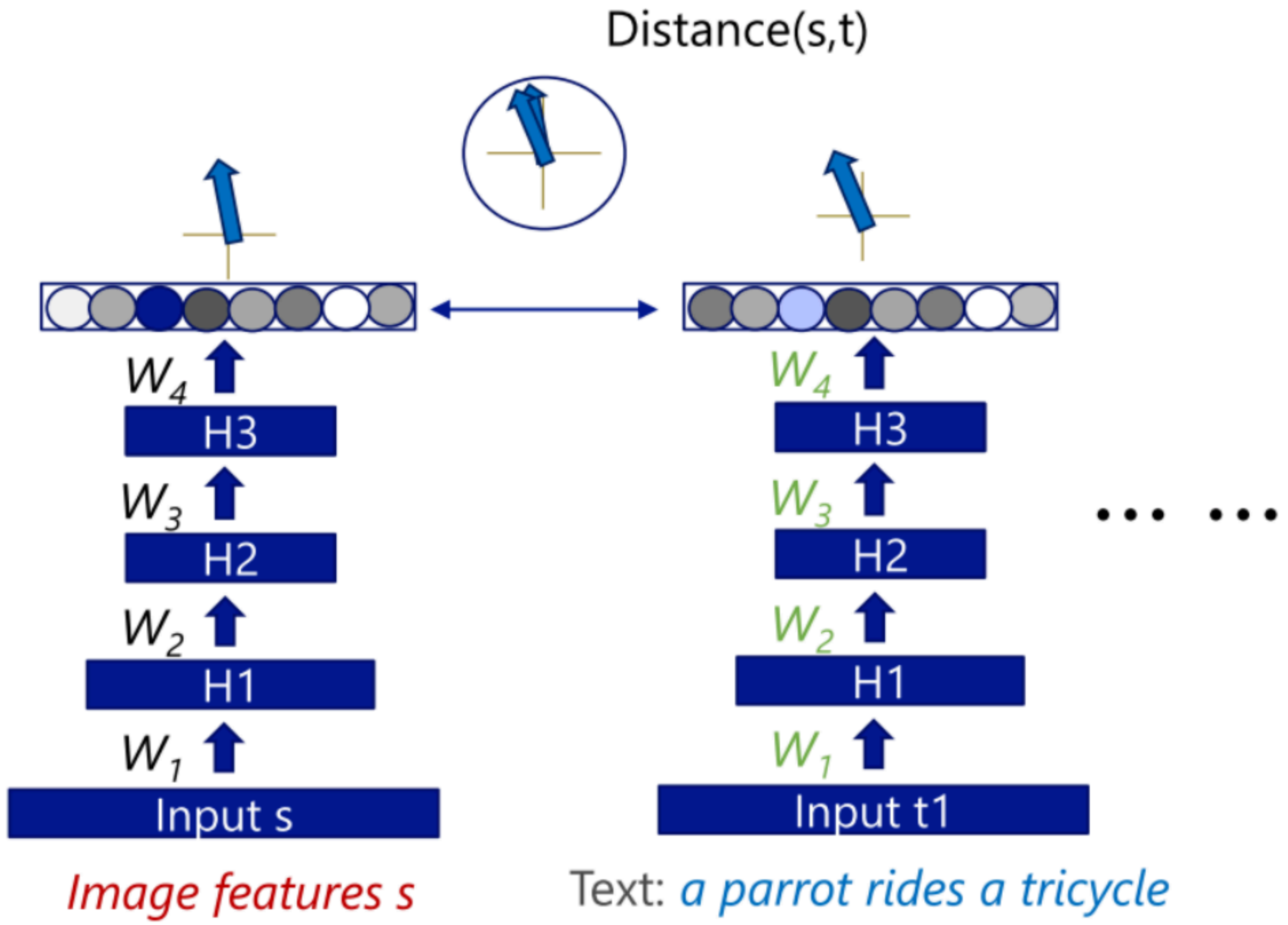
Correlated Representations: Find representations $f_1(\mathbf{x}_1), f_2(\mathbf{x}_2)$ for each view that maximize correlation:

$$\mathbf{corr}(f_1(\mathbf{x}_1), f_2(\mathbf{x}_2)) = \frac{\mathbf{cov}(f_1(\mathbf{x}_1), f_2(\mathbf{x}_2))}{\sqrt{\mathbf{var}(f_1(\mathbf{x}_1)) \cdot \mathbf{var}(f_2(\mathbf{x}_2))}}$$

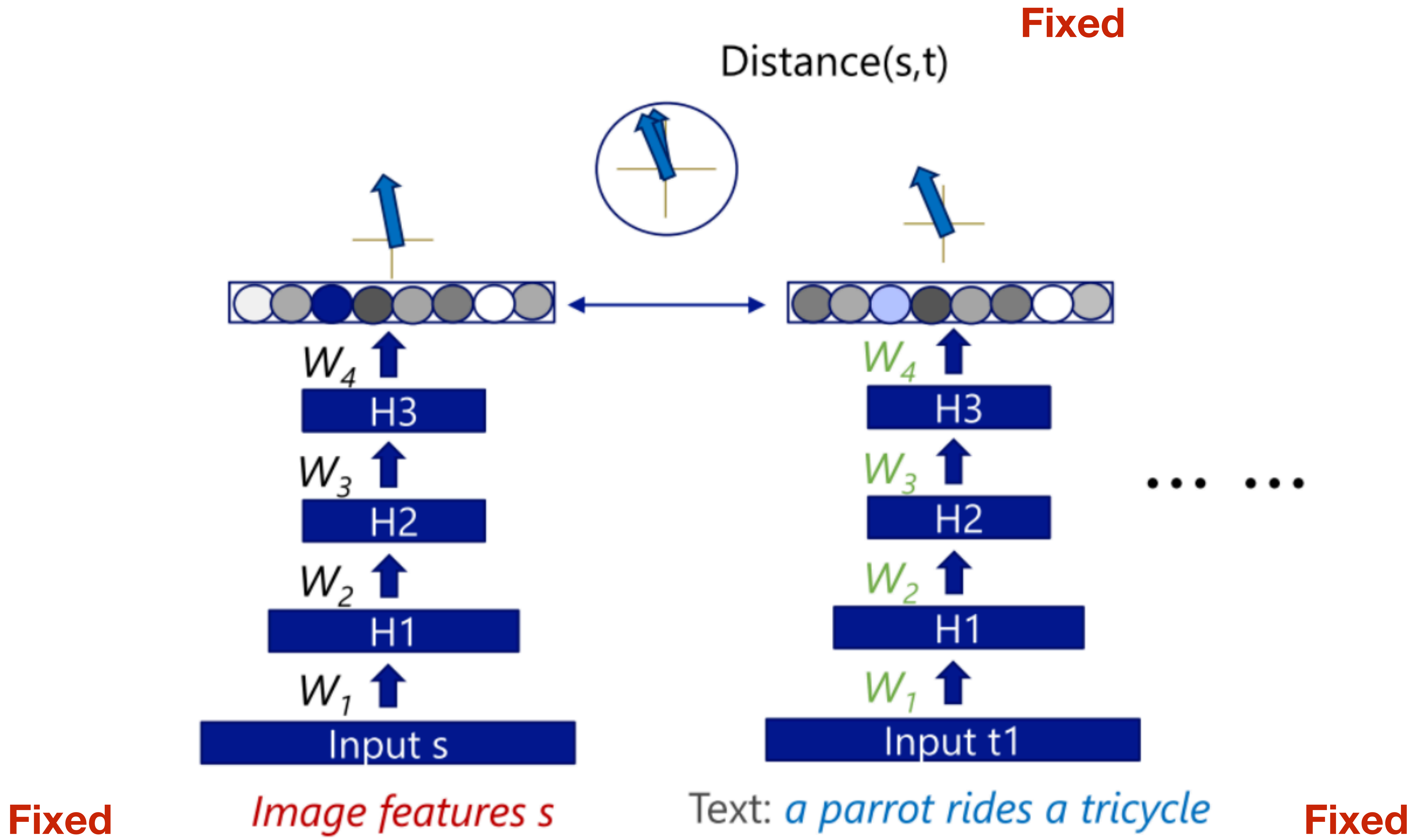
Joint Embeddings: Models that minimize distance between ground truth pairs of samples:

$$min_{f_1, f_2} D\left(f_1(\mathbf{x}_1^{(i)}), f_2(\mathbf{x}_2^{(i)})\right)$$

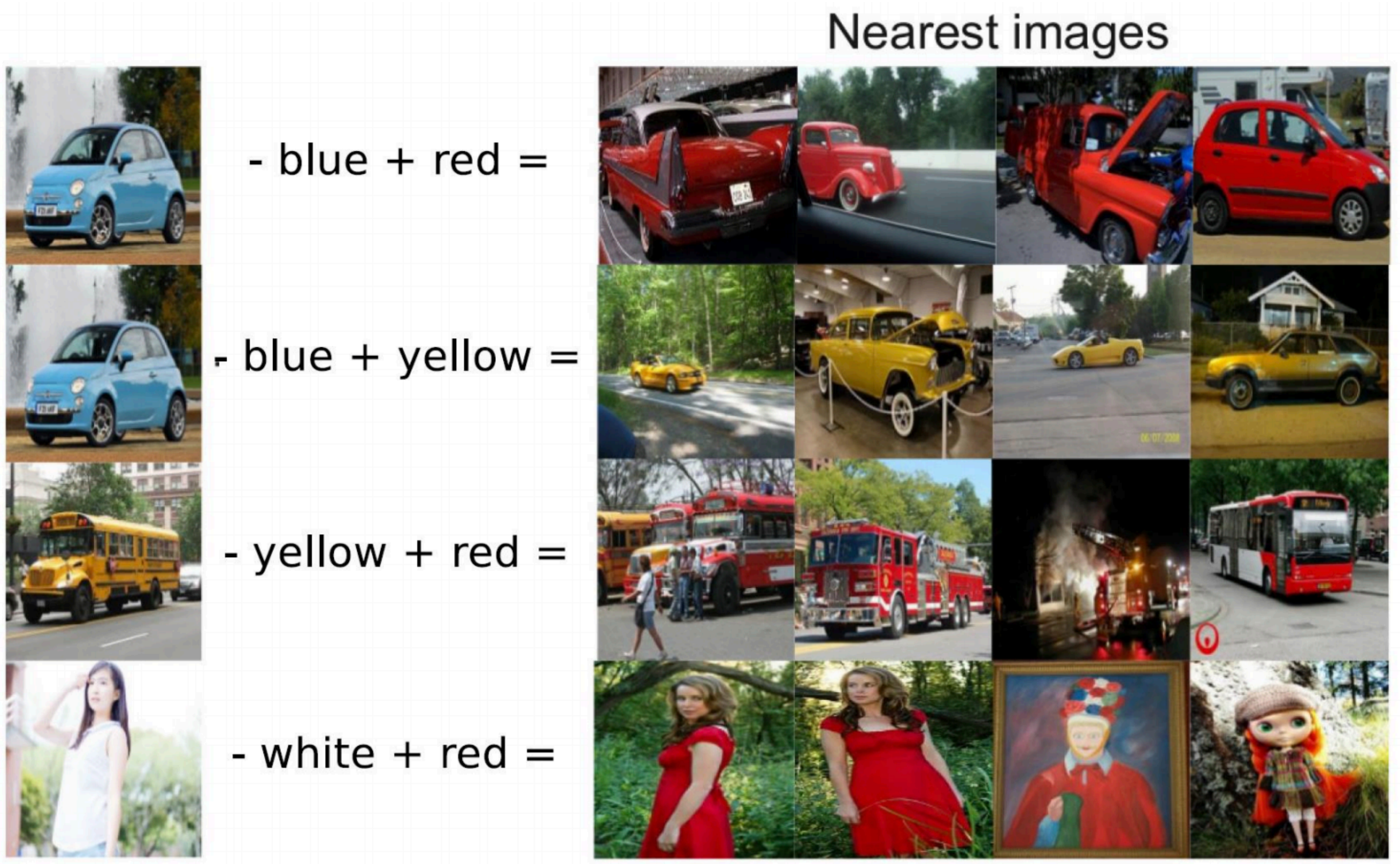
Joint **Embeddings**



Joint **Embeddings**

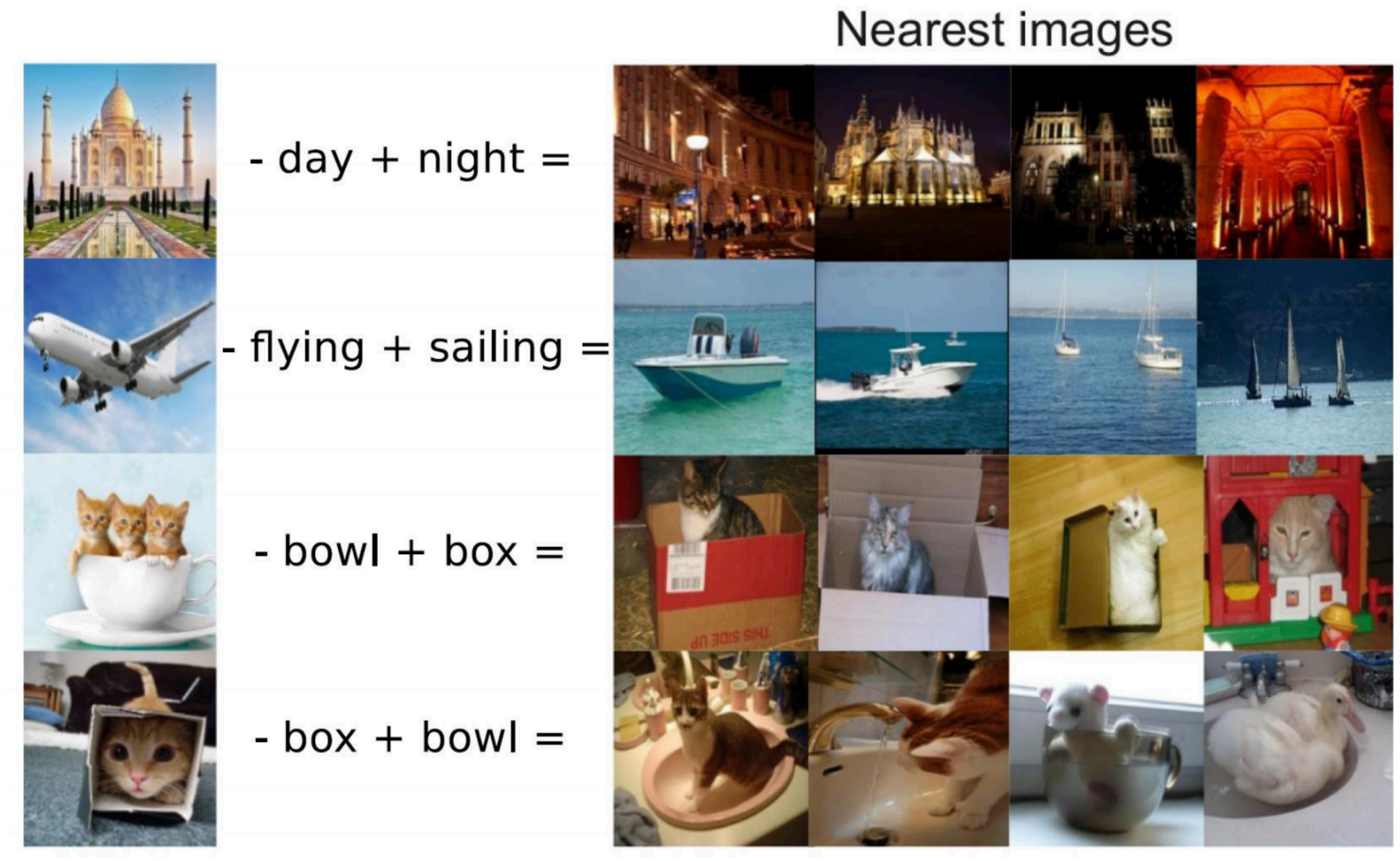


Joint **Embeddings**



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

Joint **Embeddings**



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