CMPT 983

Grounded Natural Language Understanding

January 21, 2021

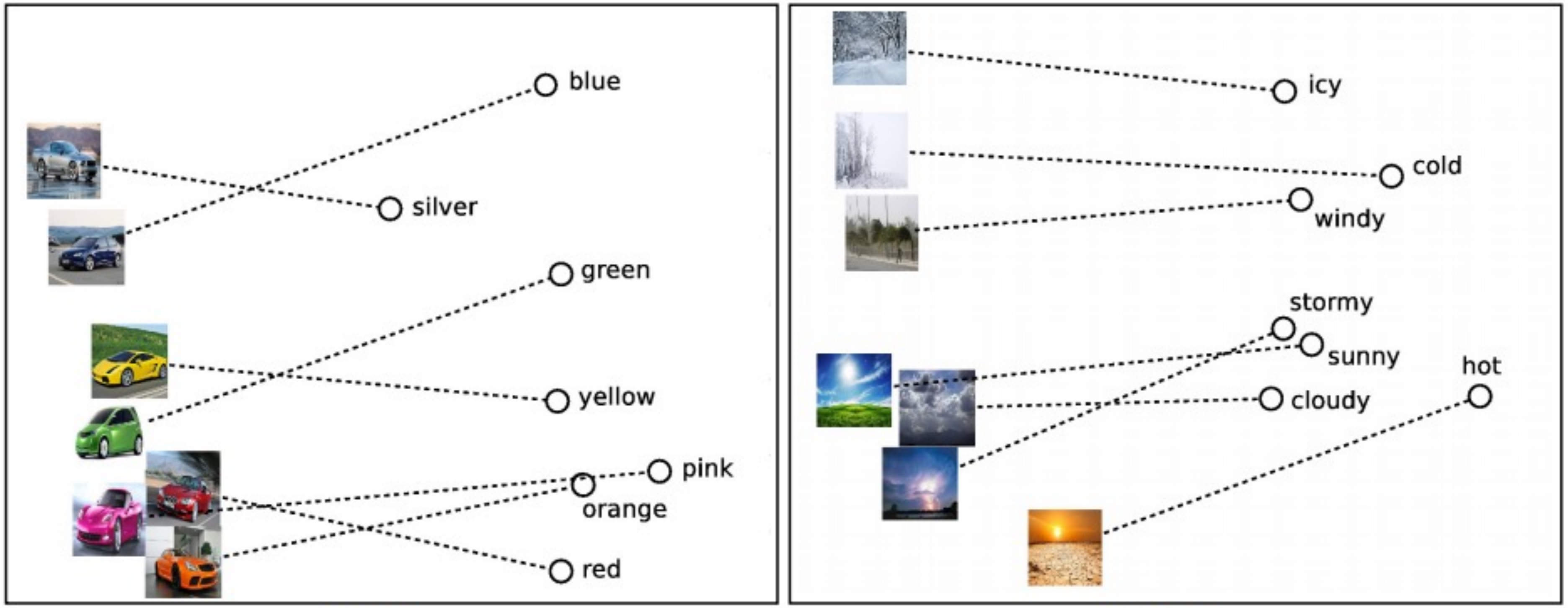
Multimodal representations

Today

- Multimodal representations
  - Joint representations
  - Correlated representations
- Applications using multimodal representations
  - Retrieval
  - Translation

Multimodal representations

Multimodal Embeddings



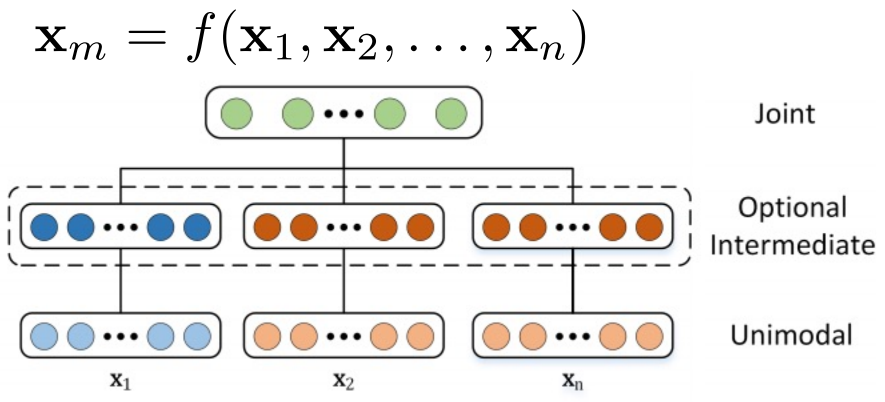
(a) Colors (b) Weather

Figure 5: PCA projection of the 300-dimensional word and image representations for (a) cars and colors and (b) weather and temperature.

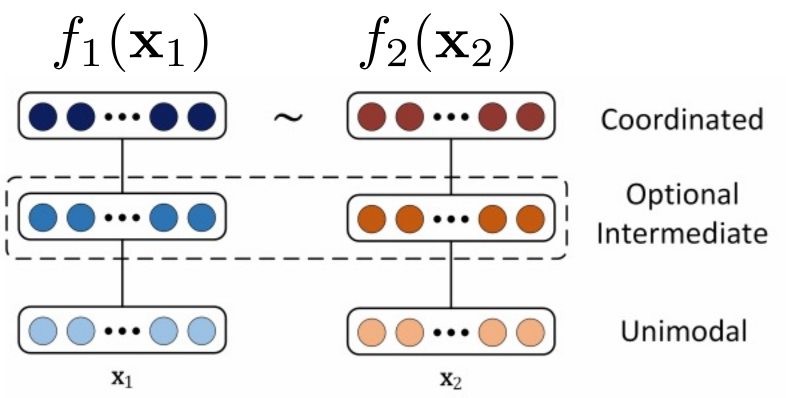
"Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models"
[Kiros, Salakhutdinov, Zemel TACL 2015]

Multimodal representations

- Joint (fused) representations
  - Single combined representation space
  - 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 embedding



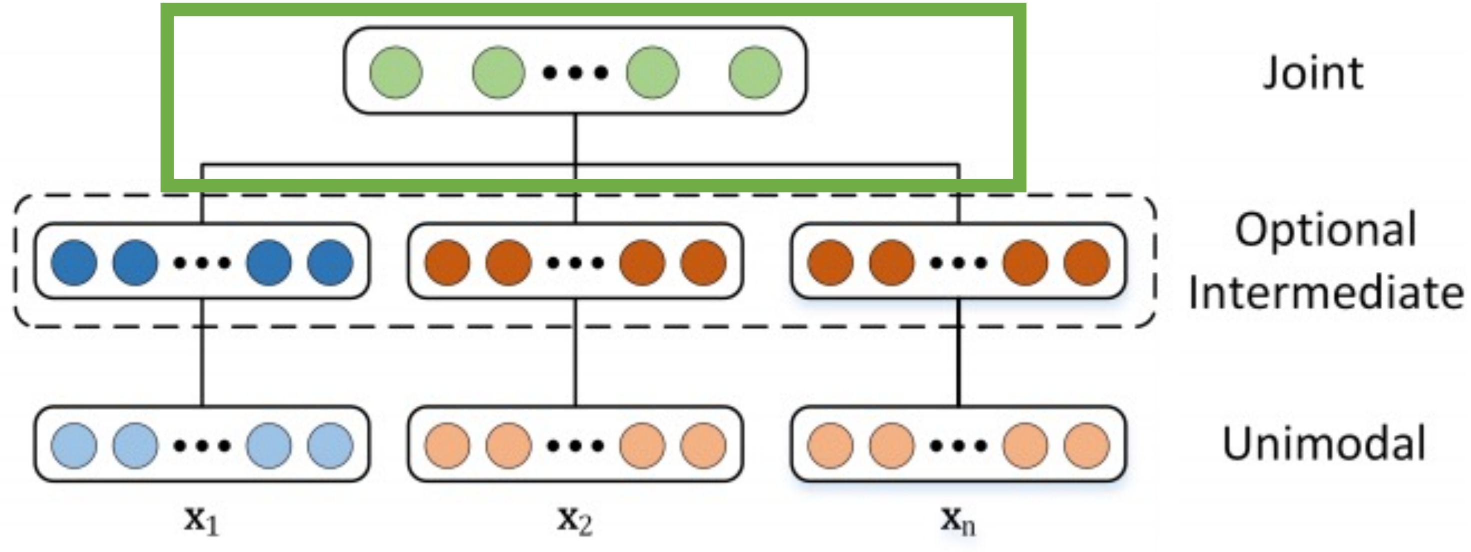
- Representations can be trained end-to-end for a task

Adapted from slide by: Louis-Philippe Morency

Joint representation

- Simplest version: modality concatenation (early fusion)
- More complex: Deep multimodal autoencoders

$$\mathbf{x}_m = f(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$$



Joint representation: Early fusion

Fusion of features / representation

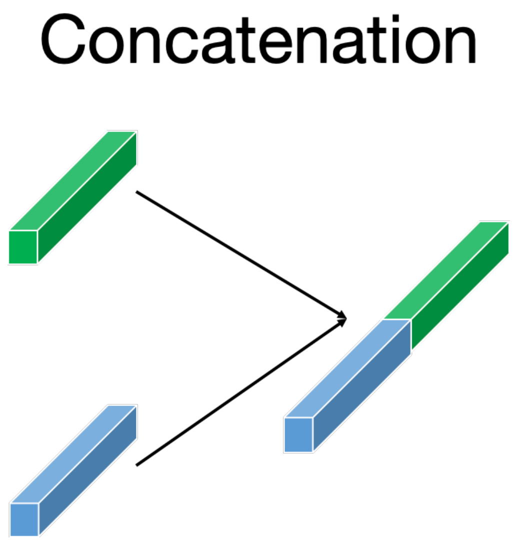
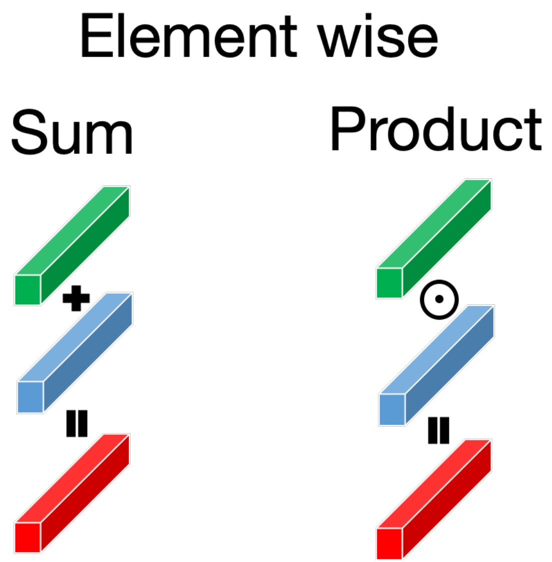
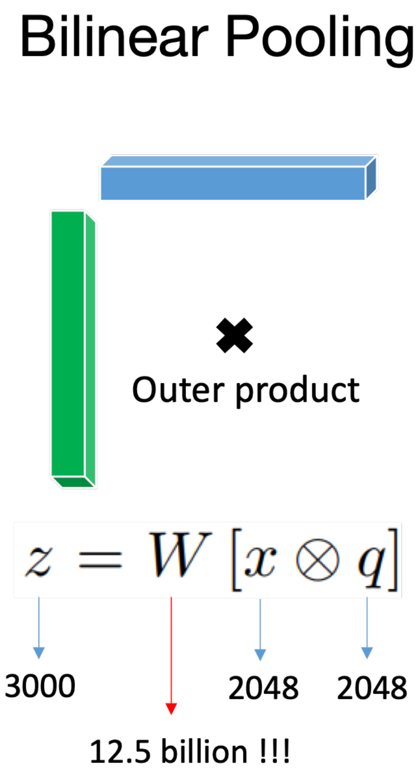


Image credit: Qi Wu





All elements can interact.
More flexible, but lots of weights!

Adapted from slide by: Stefan Lee

Joint representation: Early fusion

Compact Bilinear Pooling

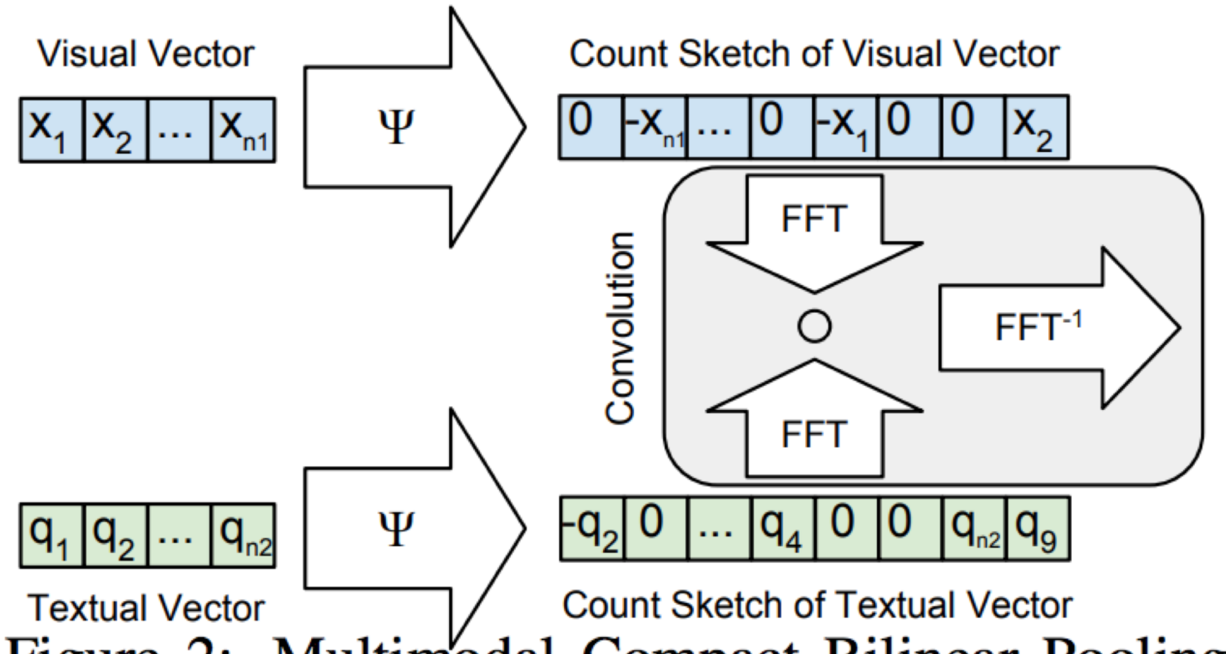


Figure 2: Multimodal Compact Bilinear Pooling

Project outer product to a lower dimensional space

Avoid direct computation of other product

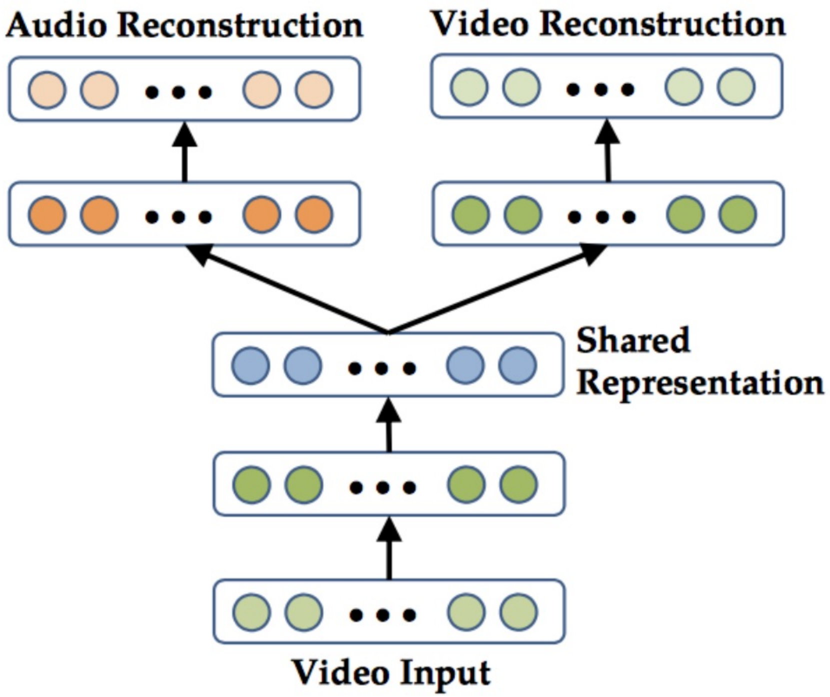
Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding https://arxiv.org/pdf/1606.01847.pdf

Slide credit: Stefan Lee

Joint representation: Autoencoders

Deep Multimodal Autoencoders

- Useful for conditioning on one modality at test time
- Can be regarded as a form of regularization

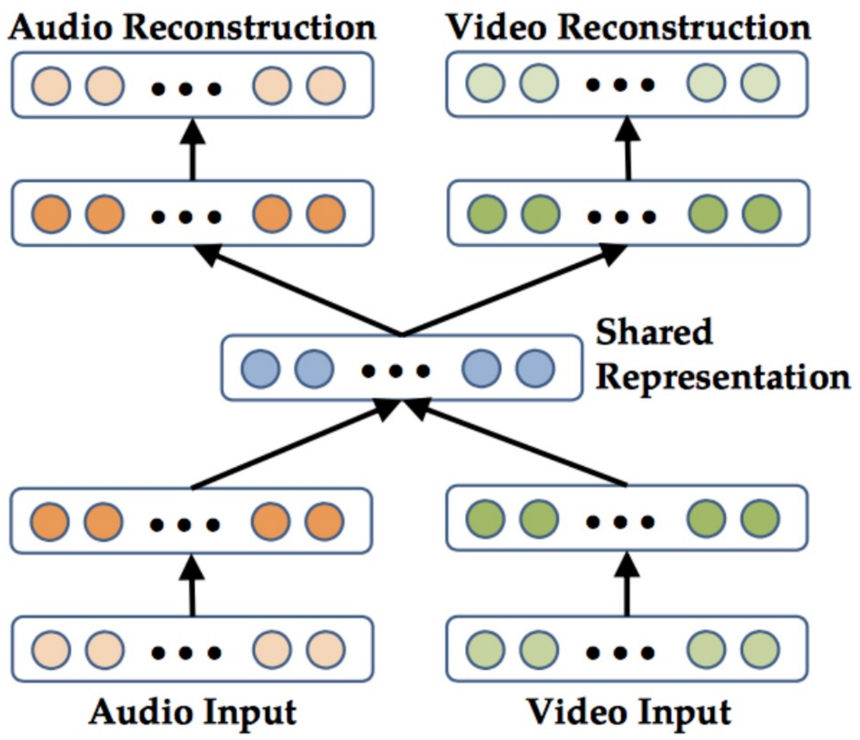


Multimodal deep learning
[Ngiam et al, ICML 2011]

Joint representation: Autoencoders

Deep Multimodal Autoencoders

- Each modality can be pre-trained
  - using denoising autoencoder
- To train the model, reconstruct both modalities using
  - both Audio & Video
  - just Audio
  - just Video



Multimodal deep learning
[Ngiam et al, ICML 2011]

Correlated representations

Canonical correlation analysis (CCA)

- 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

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

Canonical Correlation Analysis (CCA)

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

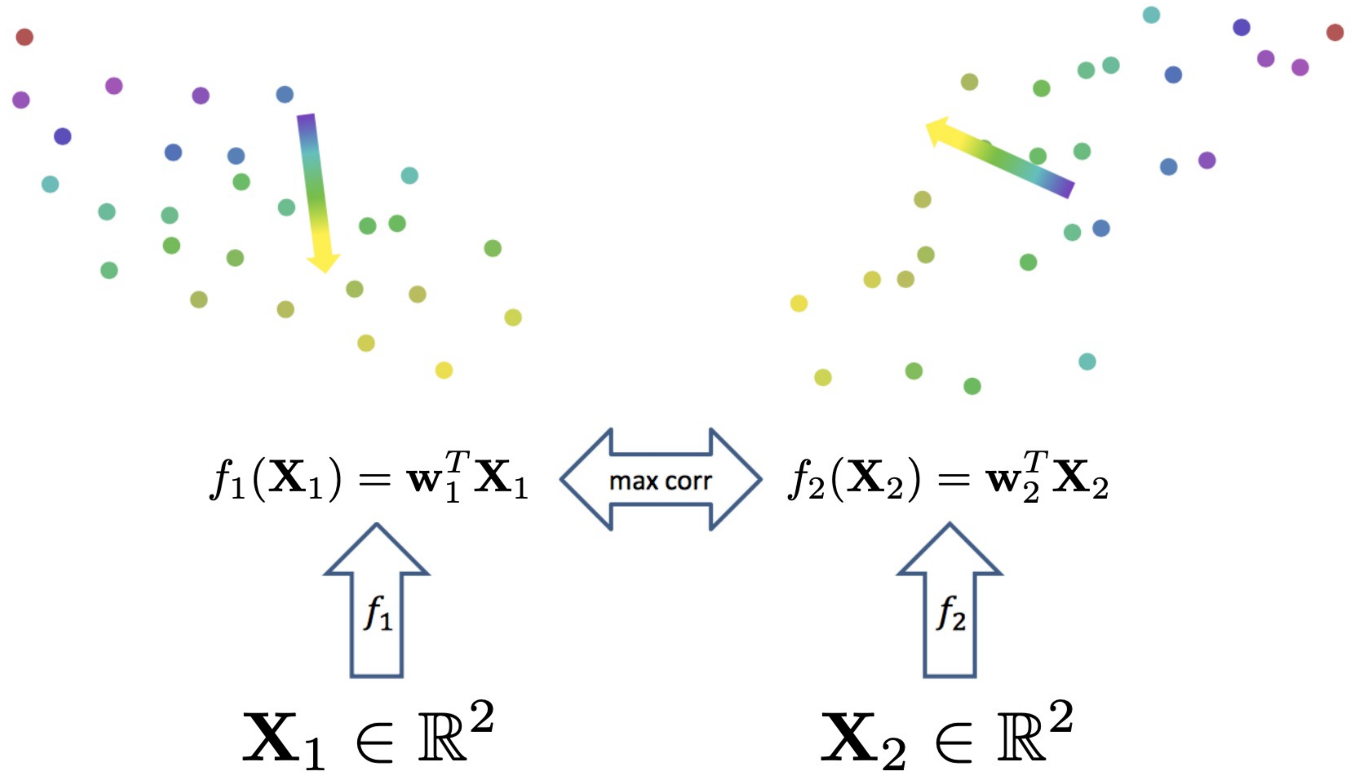
$$\mathbf{corr}(f_1(\mathbf{x}_1), f_2(\mathbf{x}_2)) = \frac{\mathbf{cov}(f_1(\mathbf{x}_1), f_2(\mathbf{x}_2))}{\sqrt{\mathbf{var}(f_1(\mathbf{x}_1)) \cdot \mathbf{var}(f_2(\mathbf{x}_2))}}$$

- 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 credit: Andrew, Arora, Bilmes, Livescu

Linear CCA

- Projections of representation



Two views of each instance have the same color

Linear CCA

- Classical technique to find linear correlated representations

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

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

- Select values for the first columns $(\mathbf{w}_{1,:1}, \mathbf{w}_{2,:1})$ of the matrices $\mathbf{W}_1$ and $\mathbf{W}_2$ 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 credit: Andrew, Arora, Bilmes, Livescu

Linear CCA

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:

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

$$\mathbf{W}_2^* = \Sigma_{11}^{-1/2}\mathbf{V}_k$$

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

Slide credit: Andrew, Arora, Bilmes, Livescu

Kernel CCA

Use non-linear functions for $f_1(\mathbf{x}_1), f_2(\mathbf{x}_2)$

- 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

- Pros:
  - Allow for non-linear functions
  - Can produce more highly correlated representations

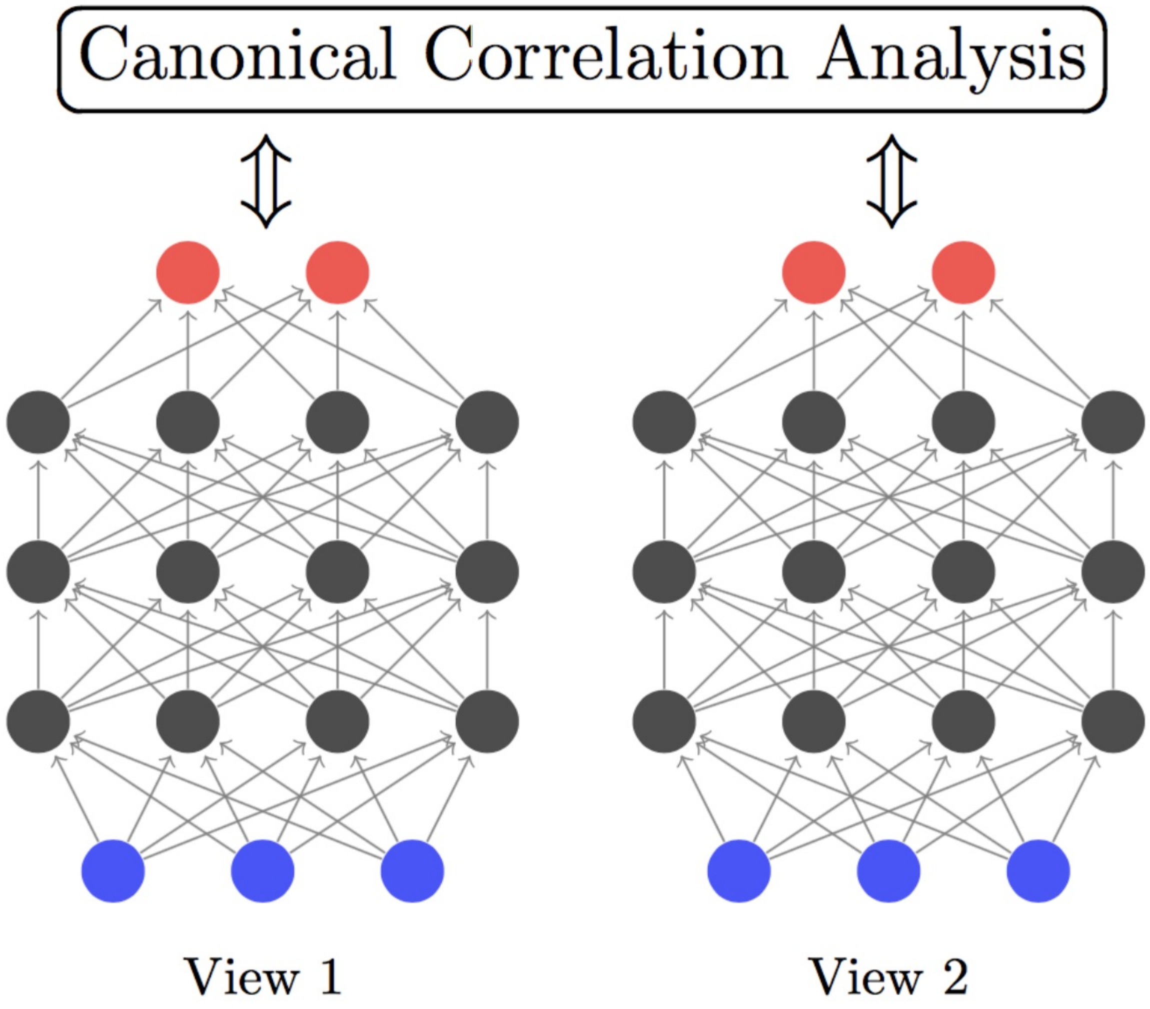
- Cons:
  - KCCA is slower to train
  - KCCA model is more difficult to interpret
  - Training set need to be stored and referenced at test time

Deep CCA

- Use neural network to represent $f_1(\mathbf{x}_1), f_2(\mathbf{x}_2)$
- Can be trained end-to-end for a task

Compared with KCCA

- Training set can be disregarded once the model is learned
- Computational speed at test time is fast



Slide credit: Andrew, Arora, Bilmes, Livescu

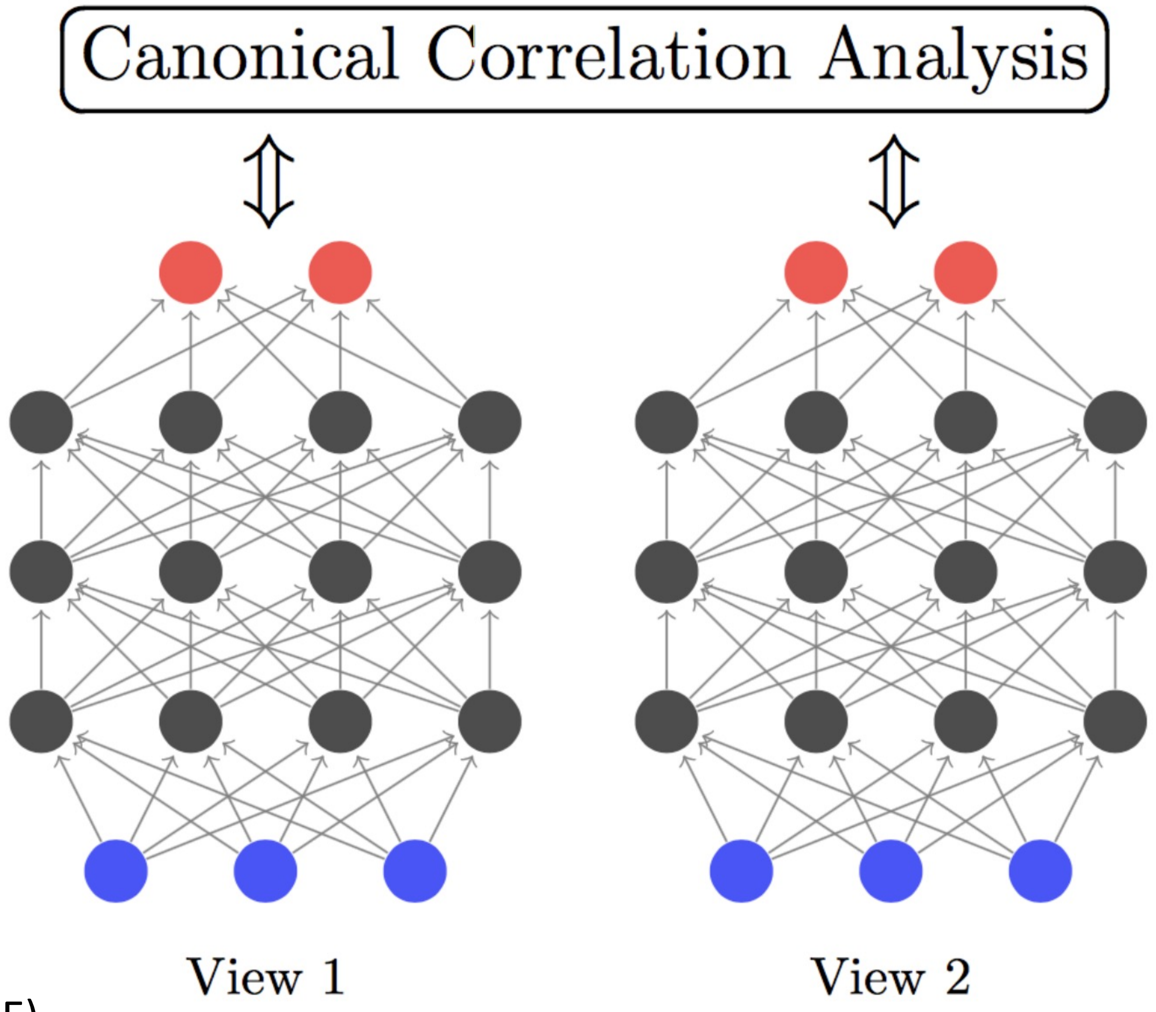
Deep CCA

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)



Extensions: Deep canonically correlated autoencoders (DCCAE)

Slide credit: Andrew, Arora, Bilmes, Livescu

Correlated representations

Canonical correlation analysis (CCA)

- 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

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

Adapted from slide by: Leonid Sigal

Discriminative Embeddings

Images and **class labels** are embedded into the same space

Image Embedding

$$\Psi(I_i) = \mathbf{W} \cdot CNN(I_i; \mathbf{\Theta}) : \mathbb{R}^D \rightarrow \mathbb{R}^d$$

Label Embedding

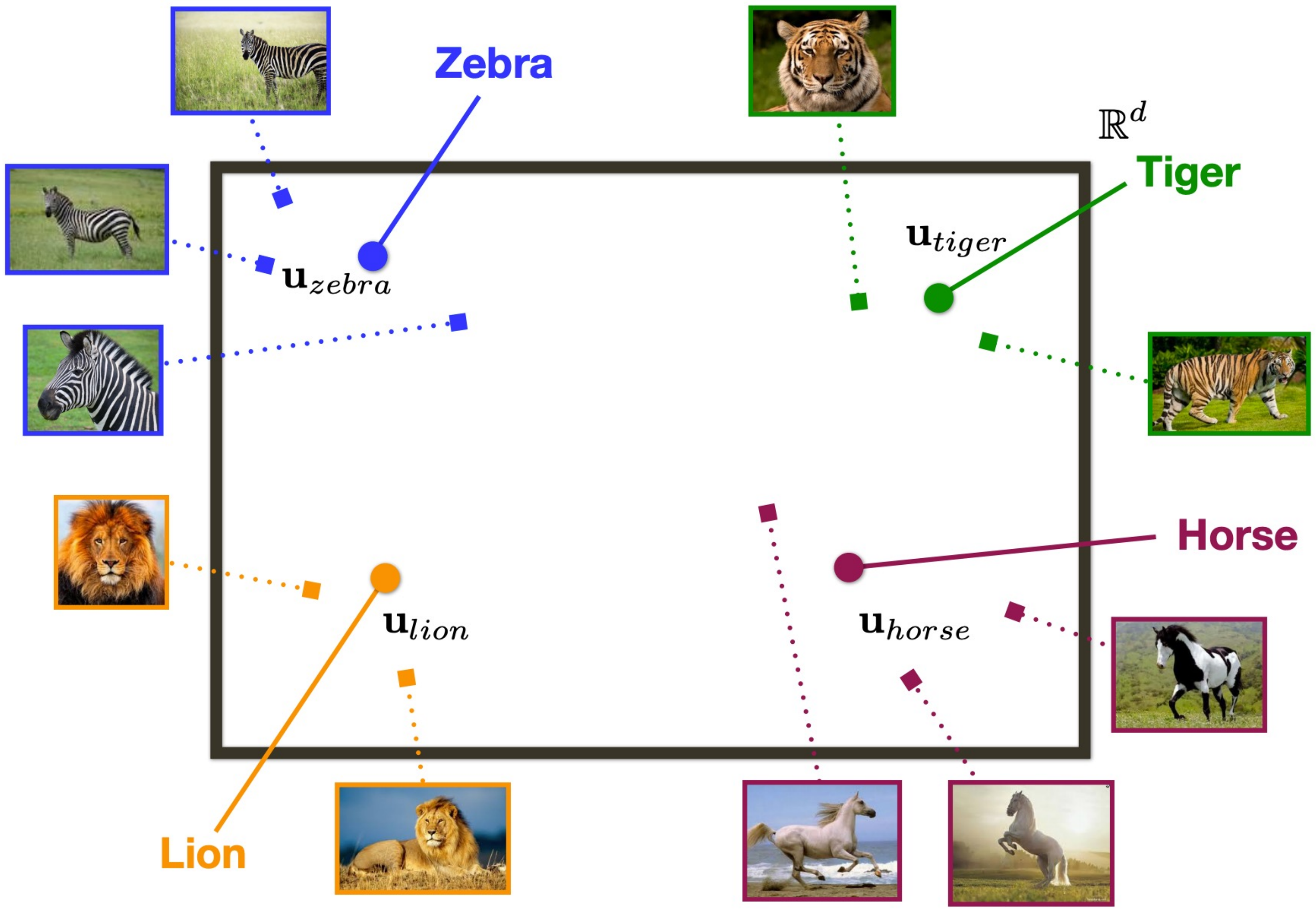
$$\Psi_L(word_i) = \mathbf{u}_i : \{1, ..., L\} \rightarrow \mathbb{R}^d$$

Similarity in Embedding Space

$$D(\mathbf{u}, \mathbf{u}') = \frac{\mathbf{u}}{||\mathbf{u}||} \cdot \frac{\mathbf{u}'}{||\mathbf{u}'||}$$

Can use different distances

$$D(\mathbf{u}, \mathbf{u}') = ||\mathbf{u} - \mathbf{u}'||_2^2$$



Adapted from slide by: Leonid Sigal

Discriminative Embeddings

Train network to minimize distance directly!

Correct label (more similar)

Other labels (less similar)

$$\mathcal{L}_{\mathrm{C}} = \sum \max(0, \alpha - S(\Psi(I_i), \mathbf{u}_{y_i}) + S(\Psi(I_i), \mathbf{u}_{y_c}))$$

Take care with signs depending on if D represents similarity or distance

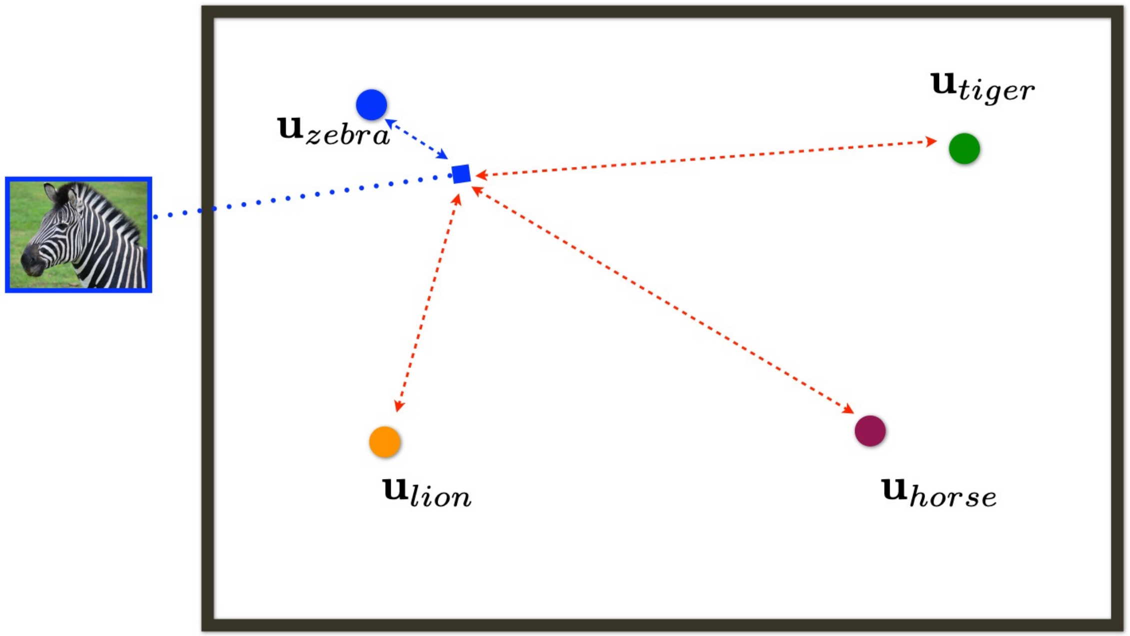
Image Embedding

$$\Psi(I_i) = \mathbf{W} \cdot CNN(I_i; \mathbf{\Theta}) \colon \mathbb{R}^D \rightarrow \mathbb{R}^d$$

Label Embedding

$$\Psi_L(word_i) = \mathbf{u}_i : \{1, ..., L\} \rightarrow \mathbb{R}^d$$

$\mathbb{R}^d$



Similarity in Embedding Space

$$D(\mathbf{u}, \mathbf{u}') = \frac{\mathbf{u}}{||\mathbf{u}||} \cdot \frac{\mathbf{u}'}{||\mathbf{u}'||}$$

Objective Function:

$$\min_{\mathbf{W},\mathbf{U}} \sum_{i}^{N} \mathcal{L}_C(\mathbf{W}, \mathbf{U}, I_i, y_i) + \lambda_1 ||\mathbf{W}||_F^2 + \lambda_2 ||\mathbf{U}||_F^2$$

[Bengio *et al*.,, NIPS'10]

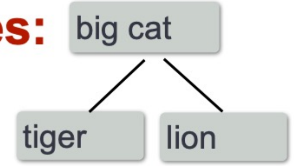
[Weinberger, Chapelle, NIPS'09]

Adapted from slide by: Leonid Sigal

Unified Semantic Embedding

"A Unified Semantic Embedding: Relating Taxonomies and Attributes" (Hwang and Sigal, NIPS 2014)

Taxonomies:



Adding regularization from **ontology / taxonomy** over labels

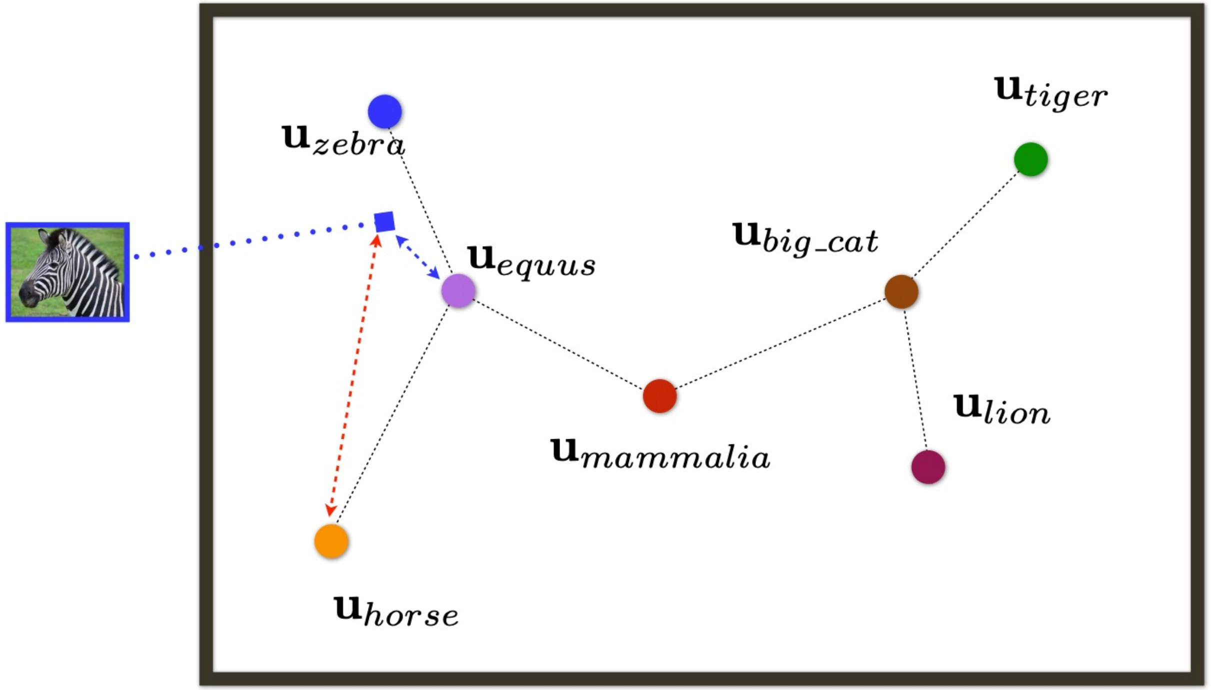
Image Embedding

$$\Psi_I(I_i) = \mathbf{W} \cdot CNN(I_i) : \mathbb{R}^D \rightarrow \mathbb{R}^d$$

Label Embedding

$$\Psi_L(word_i) = \mathbf{u}_i : \{1, ..., L\} \rightarrow \mathbb{R}^d$$

$$\mathcal{L}_S(\boldsymbol{W}, \boldsymbol{U}, \boldsymbol{x}_i, y_i) = \sum_{s \in \mathcal{P}_{y_i}} \sum_{c \in \mathcal{S}_s} [1 + \underline{\|\boldsymbol{W}\boldsymbol{x}_i - \boldsymbol{u}_s\|_2^2} - \underline{\|\boldsymbol{W}\boldsymbol{x}_i - \boldsymbol{u}_c\|_2^2}]_+$$



Similarity in Embedding Space

$$D(\mathbf{u}, \mathbf{u}') = ||\mathbf{u} - \mathbf{u}'||_2^2$$

Objective Function:

$$\min_{\mathbf{W},\mathbf{U},\mathbf{B}} \sum_i^N \mathcal{L}_C(\mathbf{W}, \mathbf{U}, I_i, y_i) + \underline{\mathcal{L}_S(\mathbf{W}, \mathbf{U}, I_i, y_i)} + \mathcal{L}_A(\mathbf{W}, \mathbf{U}, I_i, y_i) + \lambda_1 ||\mathbf{W}||_F^2 + \lambda_2 ||\mathbf{U}||_F^2$$

Slide credit: Leonid Sigal

Unified Semantic Embedding

Attributes : has(zebra, Stripes)

Attributes embedded as (basis) **vectors** in the semantic space

Image Embedding

$$\Psi_I(I_i) = \mathbf{W} \cdot CNN(I_i) : \mathbb{R}^D \rightarrow \mathbb{R}^d$$

Label Embedding

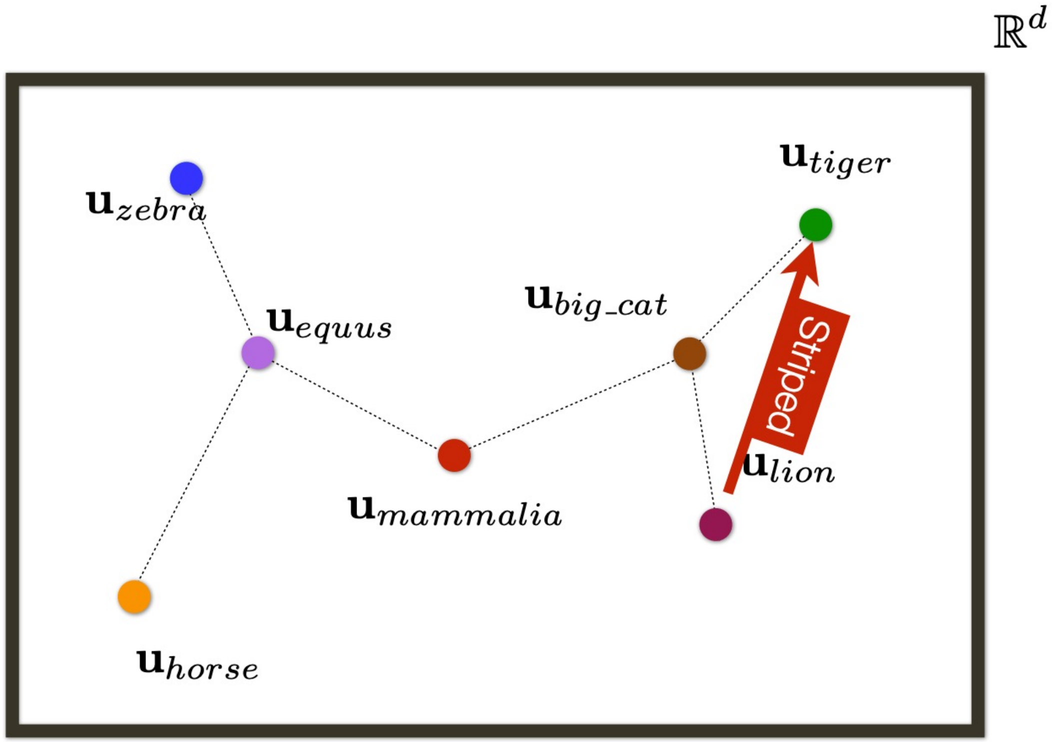
$$\Psi_L(word_i) = \mathbf{u}_i : \{1, ..., L\} \rightarrow \mathbb{R}^d$$

Attribute Embedding

$$\Psi_A(attr_i) = \mathbf{a}_i : \{1, ..., A\} \rightarrow \mathbb{R}^d, s.t. ||\mathbf{a}_i||^2 \leq 1$$

Similarity in Embedding Space

$$D(\mathbf{u}, \mathbf{u}') = ||\mathbf{u} - \mathbf{u}'||_2^2$$



Objective Function:

$$\min_{\mathbf{W},\mathbf{U},\mathbf{B}} \sum_i^N \mathcal{L}_C(\mathbf{W}, \mathbf{U}, I_i, y_i) + \mathcal{L}_S(\mathbf{W}, \mathbf{U}, I_i, y_i) + \underline{\mathcal{L}_A(\mathbf{W}, \mathbf{U}, I_i, y_i)} + \mathcal{R}(\mathbf{U}, \mathcal{B}) + \lambda_1||\mathbf{W}||_F^2 + \lambda_2||\mathbf{U}||_F^2$$

Slide credit: Leonid Sigal

Unified Semantic Embedding

[Hwang et al., 2014]

Image Embedding

$$\Psi_I(I_i) = \mathbf{W} \cdot CNN(I_i) : \mathbb{R}^D \rightarrow \mathbb{R}^d$$

Label Embedding

$$\Psi_L(word_i) = \mathbf{u}_i : \{1, ..., L\} \rightarrow \mathbb{R}^d$$

Attribute Embedding

$$\Psi_A(attr_i) = \mathbf{a}_i : \{1, ..., A\} \rightarrow \mathbb{R}^d, s.t. ||\mathbf{a}_i||^2 \leq 1$$

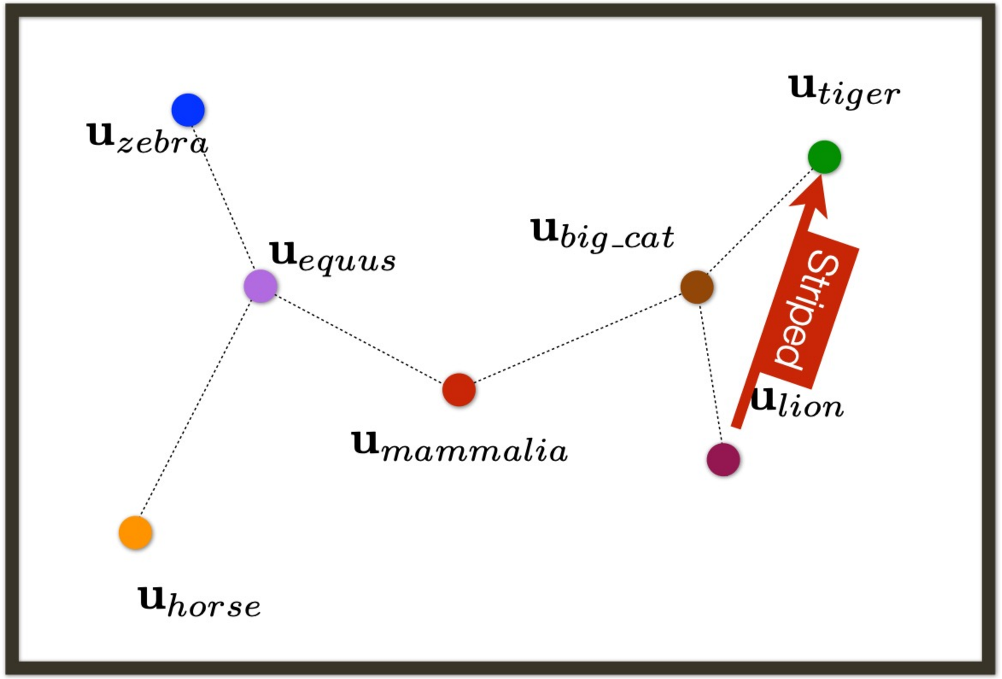
Similarity in Embedding Space

$$D(\mathbf{u}, \mathbf{u}') = ||\mathbf{u} - \mathbf{u}'||_2^2$$

$$\mathcal{R}(\boldsymbol{U}, \boldsymbol{B}) = \sum_c^{\mathrm{C}} \|\boldsymbol{u}_c - \boldsymbol{u}_p - \boldsymbol{U}^A \boldsymbol{\beta}_c\|_2^2 + \gamma_2 \|\boldsymbol{\beta}_c + \boldsymbol{\beta}_o\|_2^2.$$

each category is a parent + sparse subset of attribute bases

$\mathbb{R}^d$



Objective Function:

$$\min_{\mathbf{W},\mathbf{U},\mathbf{B}} \sum_i^N \mathcal{L}_C(\mathbf{W}, \mathbf{U}, I_i, y_i) + \mathcal{L}_S(\mathbf{W}, \mathbf{U}, I_i, y_i) + \mathcal{L}_A(\mathbf{W}, \mathbf{U}, I_i, y_i) + \underline{\mathcal{R}(\mathbf{U}, \mathcal{B})} + \lambda_1 ||\mathbf{W}||_F^2 + \lambda_2 ||\mathbf{U}||_F^2$$

Slide credit: Leonid Sigal

Animals with attributes

(we assume no association between classes and attributes)

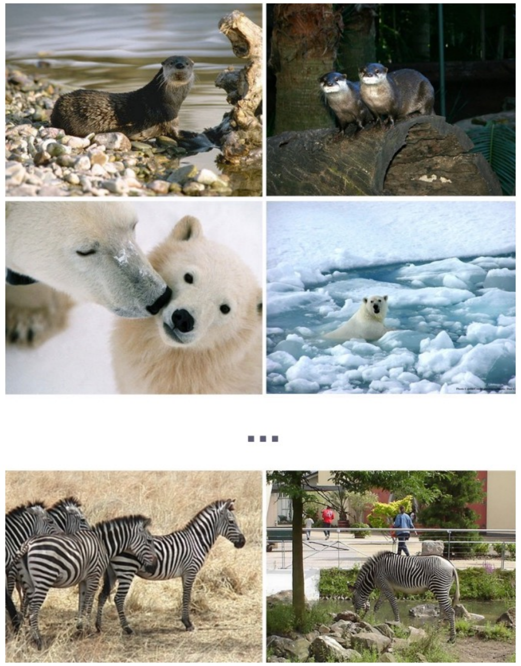
Labeled Images

Otter

Polar Bear

...

Zebra



30,475 Images

50 Animal Classes

Semantic Attributes

black
white
blue
brown
gray
orange
red
yellow
patches

...

paws
longlegs
longneck
tail
chew teeth
meat teeth
buck teeth
horns
claws
tusks

85 Attributes

Class Ontology

WordNet
A lexical database for English

50 Animal Classes are Leaves

[Lampert, Nickisch, Harmeling, CVPR'09]

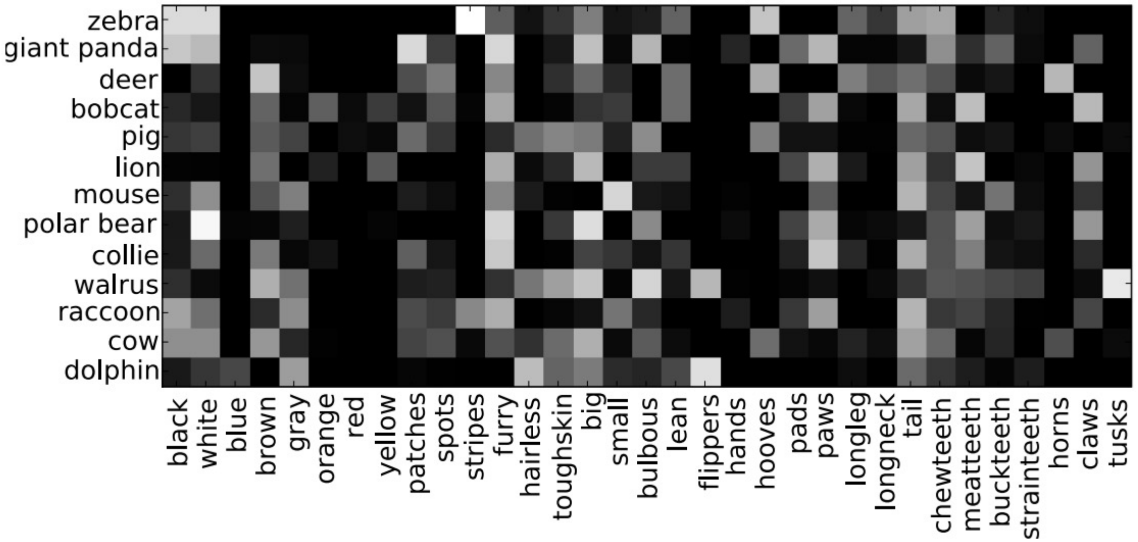
Slide credit: Leonid Sigal

Interpretable representations

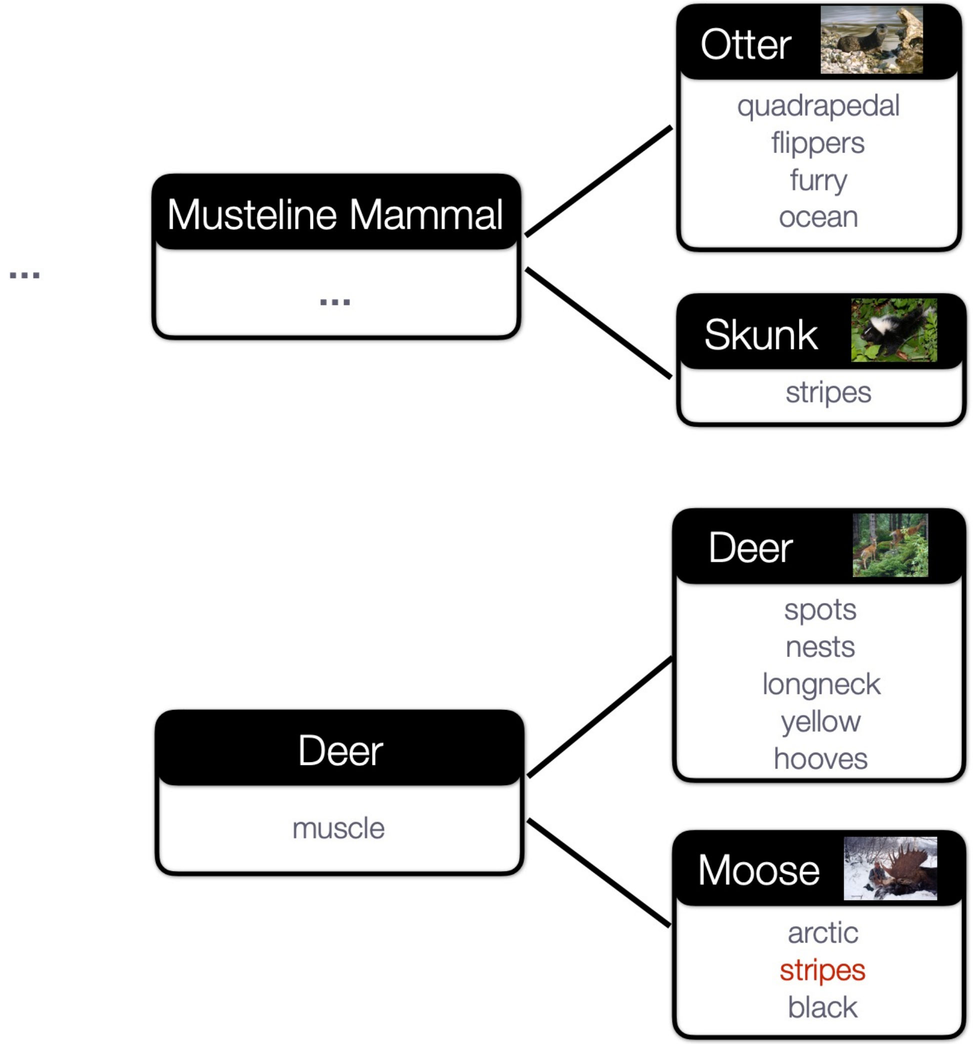
Results with AWA (with latent attributes)

Model **benefits:**

- highly interpretable
- efficient in learning



alternative attribute-based representations



Slide credit: Leonid Sigal

From words to sentences

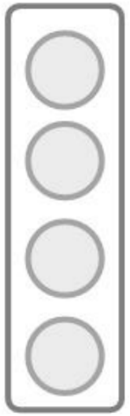
Label Embedding

$$\Psi_L(word_i) = \mathbf{u}_i : \{1, ..., L\} \to \mathbb{R}^d$$

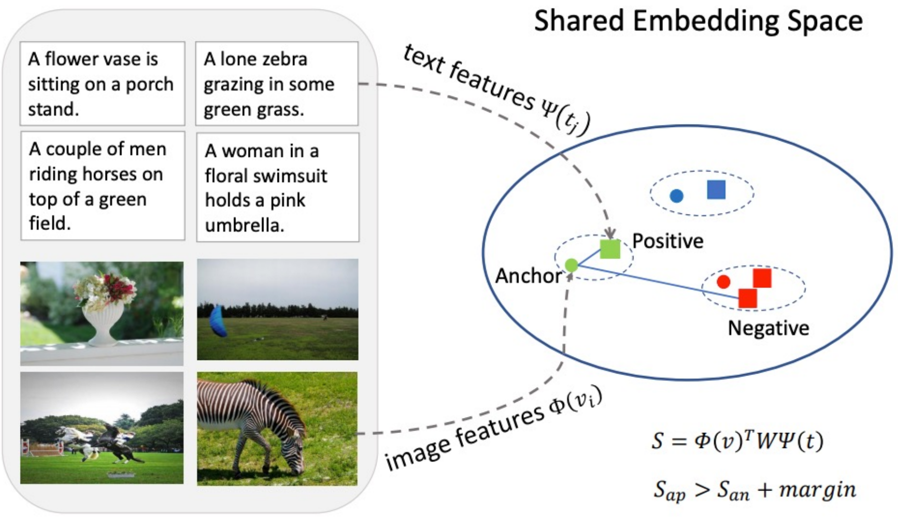
Sentence embedding

$$\Psi_L(w_1, \ldots, w_k)$$

Sequence of words (characters) → Composition Function →



Average
BoW (FCN)
RNN
CNN
Transformers
GraphNN



(i, c) : matching

$(\hat{\imath}, c), (i, \hat{c})$: not matching

Triplet based ranking loss:

$$\ell_{SH}(i,c) = \sum_{\hat{c}}[\alpha - s(i,c) + s(i,\hat{c})]_+ + \sum_{\hat{i}}[\alpha - s(i,c) + s(\hat{i},c)]_+$$

Applications

Retrieval

- Text to image/video retrieval
- Image/video to text retrieval

MS COCO

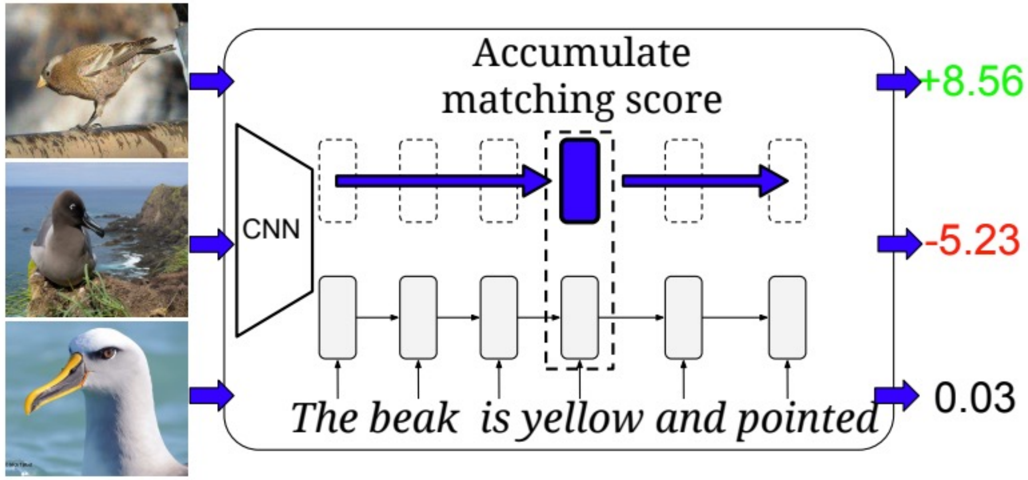
The man at bat readies to swing at the pitch while the umpire looks on.

A large bus sitting next to a very tall building.

Flicker 8k, Flicker 30k

- A biker in red rides in the countryside.
- A biker on a dirt path.
- A person rides a bike off the top of a hill and is airborne.
- A person riding a bmx bike on a dirt course.
- The person on the bicycle is wearing red.

Retrieval



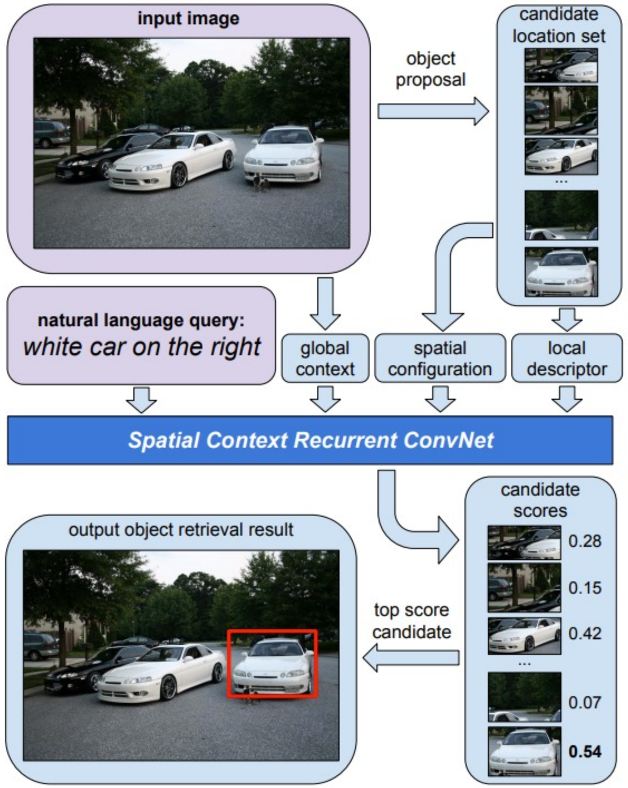
	Top-1 Acc (%)		AP@50 (%)	
Embedding	**DA-SJE**	**DS-SJE**	**DA-SJE**	**DS-SJE**
ATTRIBUTES	50.9	50.4	20.4	**50.0**
WORD2VEC	38.7	38.6	7.5	33.5
BAG-OF-WORDS	43.4	44.1	24.6	39.6
CHAR CNN	47.2	48.2	2.9	42.7
CHAR LSTM	22.6	21.6	11.6	22.3
CHAR CNN-RNN	54.0	54.0	6.9	45.6
WORD CNN	50.5	51.0	3.4	43.3
WORD LSTM	52.2	53.0	**36.8**	46.8
WORD CNN-RNN	**54.3**	**56.8**	4.8	48.7

CUB Birds

"Learning Deep Representations of Fine-Grained Visual Descriptions" (Reed et al, CVPR 2016)

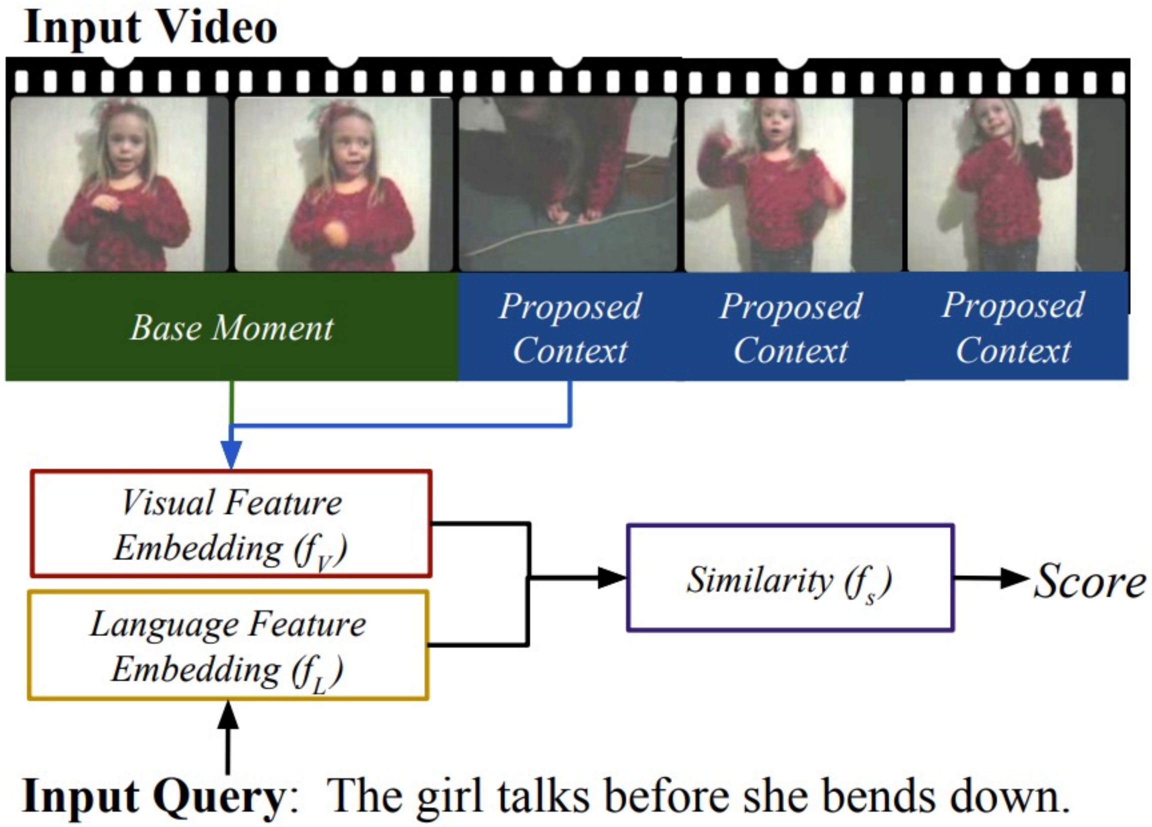
Retrieval

Match image region to language



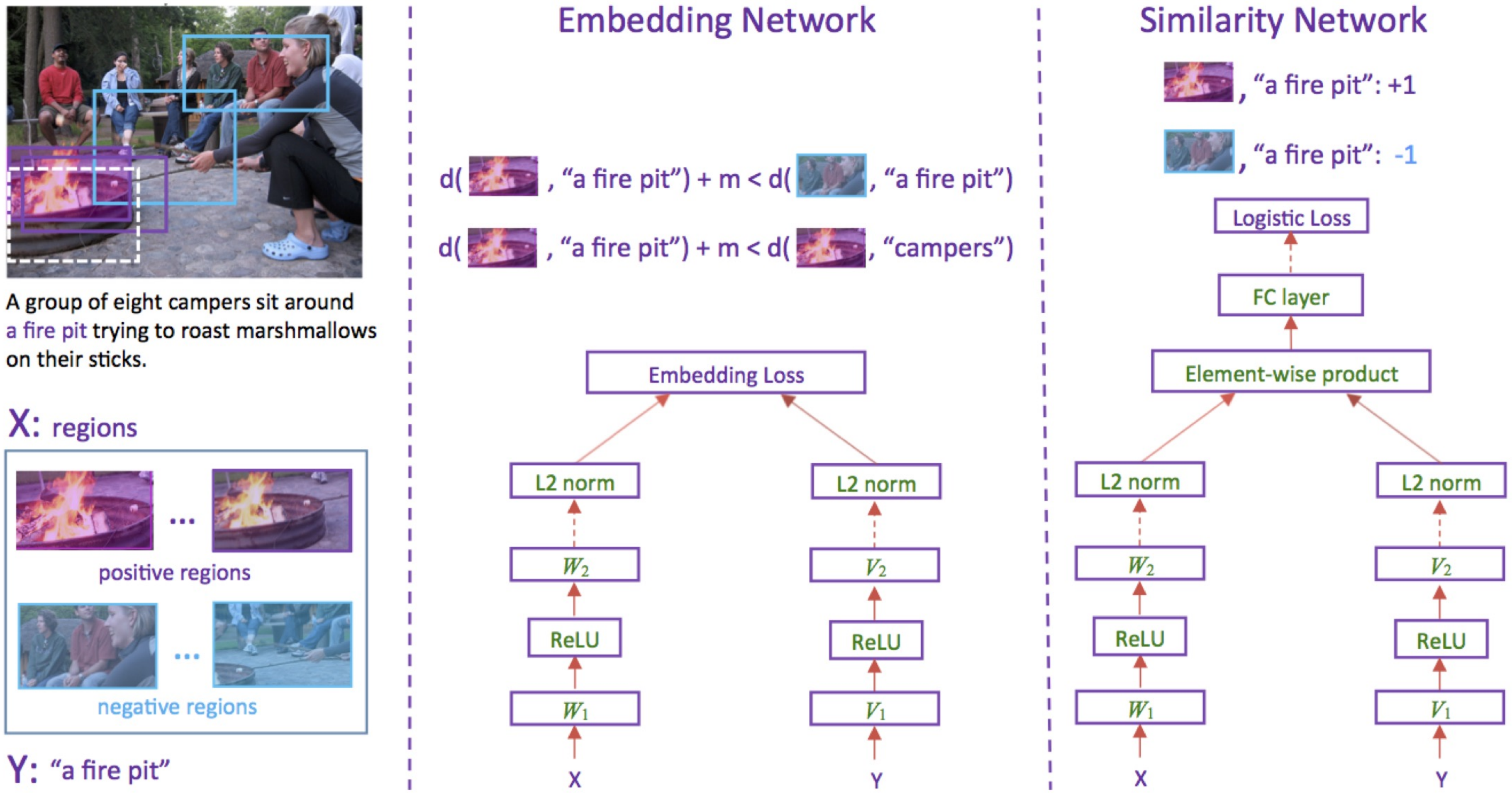
Natural Language Object Retrieval
(Hu et al, CVPR 2016)

Match video frames to language



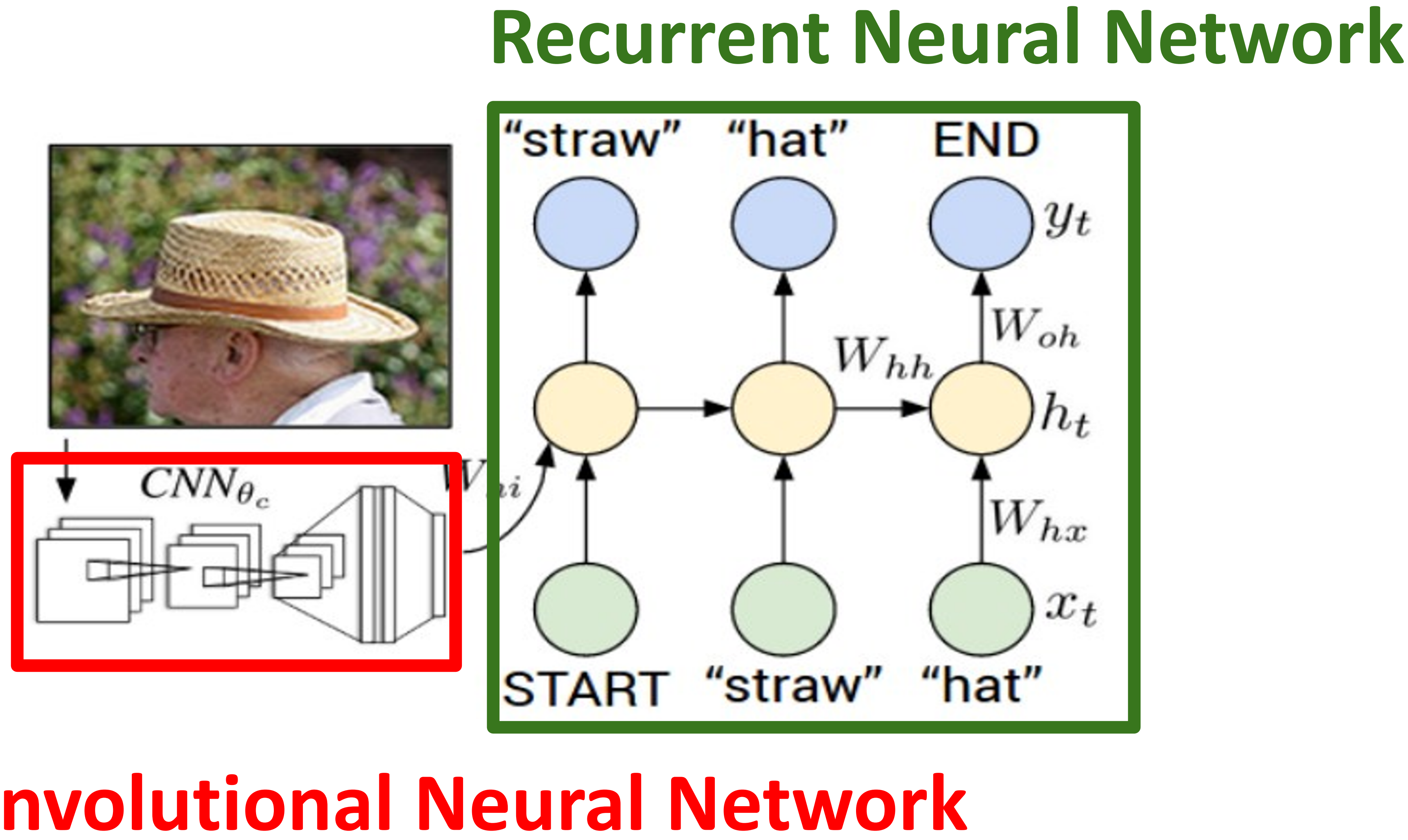
Localizing moments in video with temporal language
(Hendricks et al, EMNLP, 2018)

Retrieval: Phrase localization



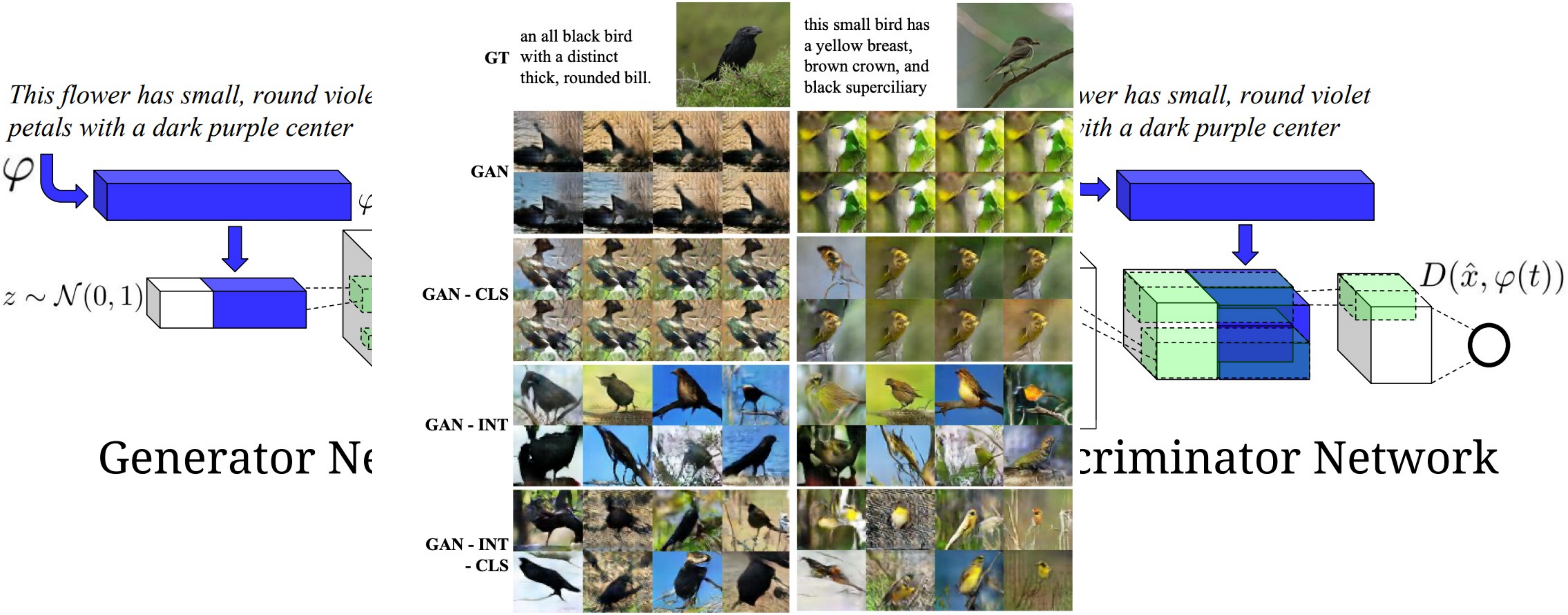
Learning Two-Branch Neural Networks for Image-Text Matching Tasks
(Wang et al, TPAMI 2018)

Translation (image to text)



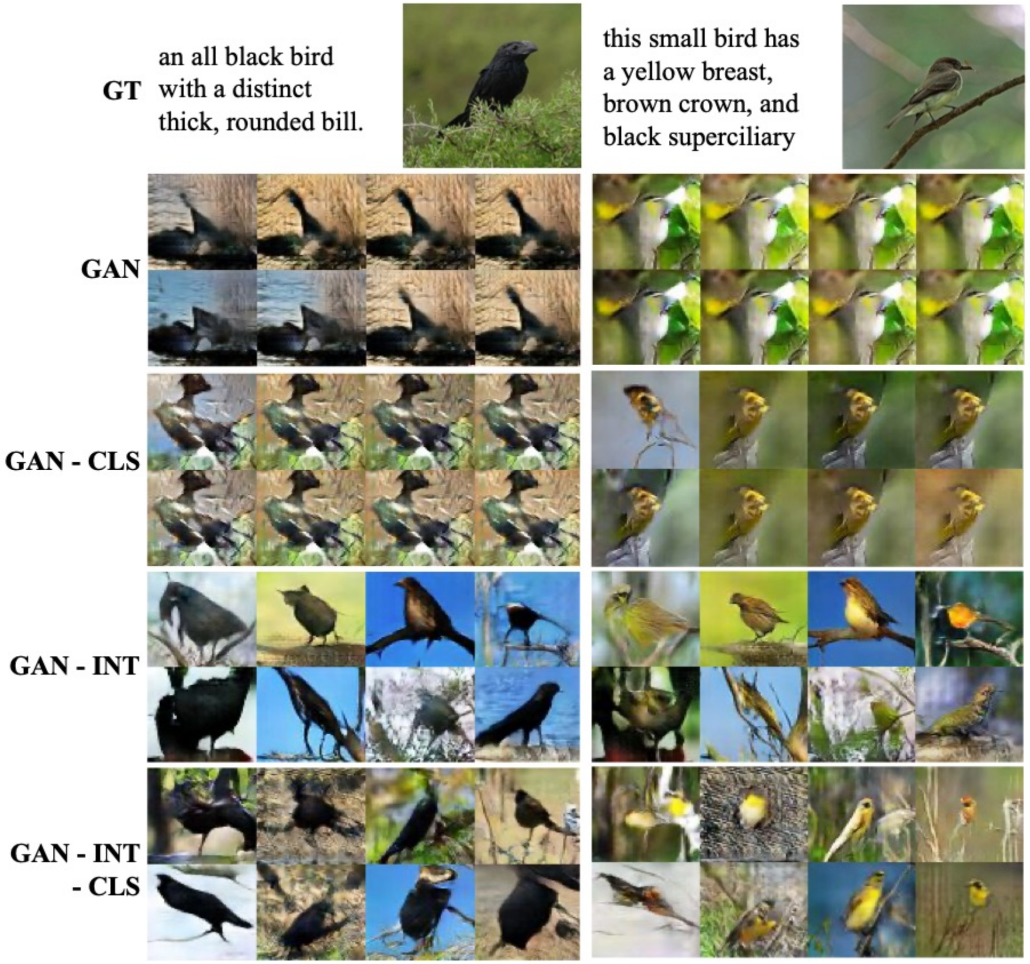
"Deep Visual-Semantic Alignments for Generating Image Descriptions" (Karpathy and Fei-Fei, CVPR 2015)

Translation (text to image)



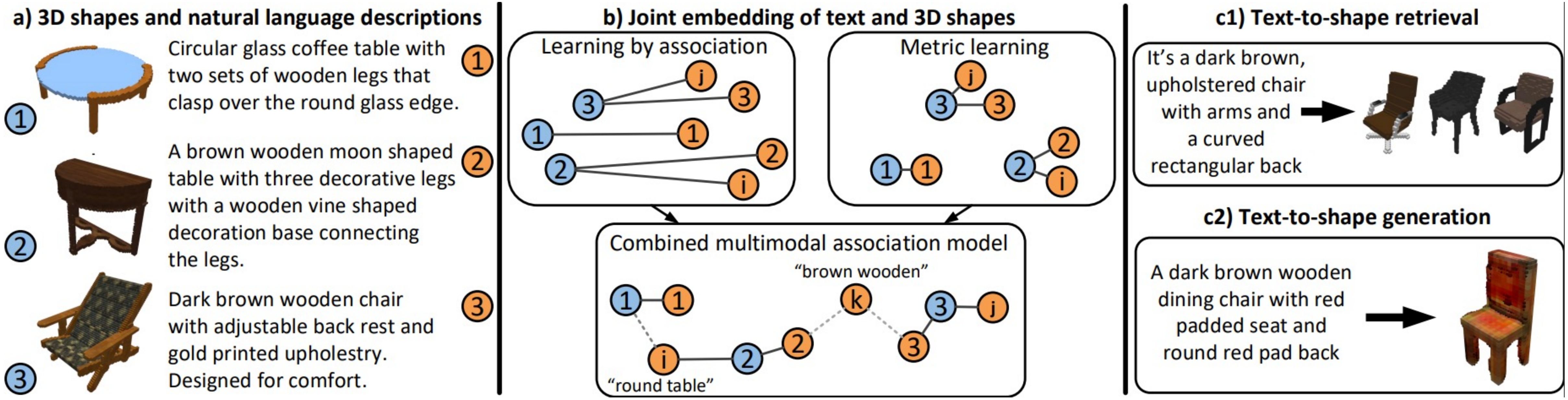
"Generative Adversarial Text to Image Synthesis" (Reed et al, ICML 2016)

Translation (text to image)



"Generative Adversarial Text to Image Synthesis" (Reed et al, ICML 2016)

Text and shape

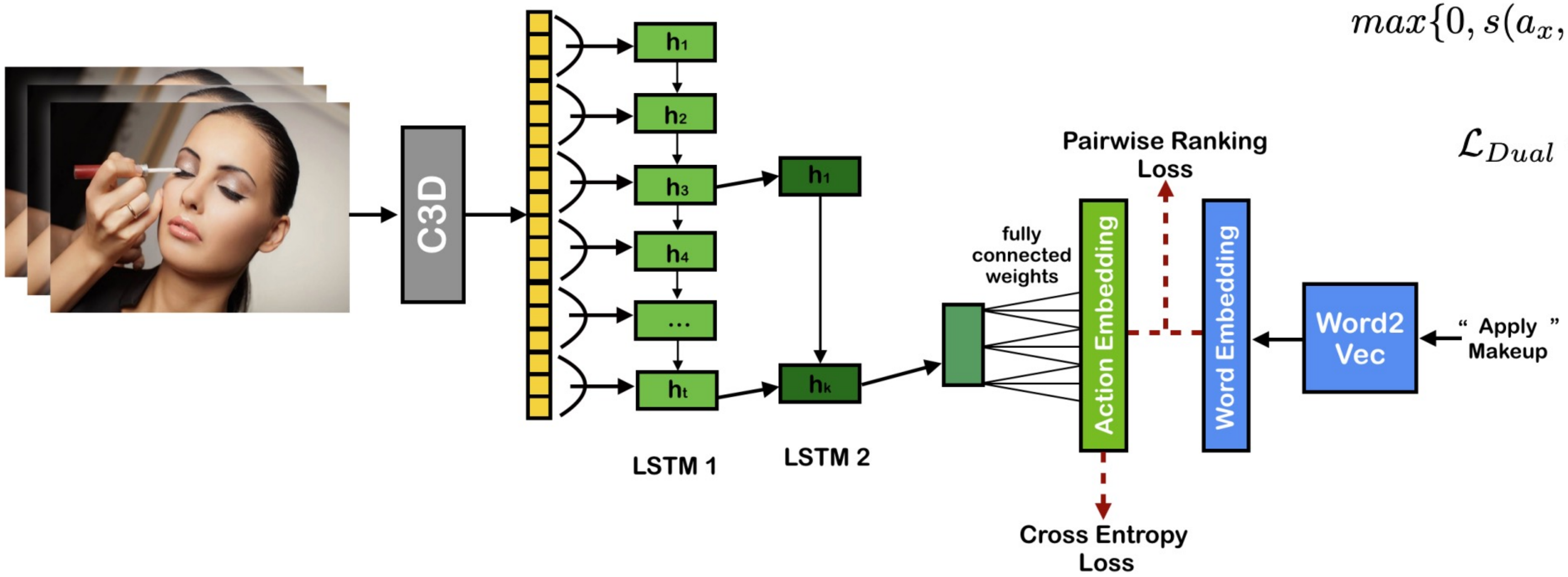


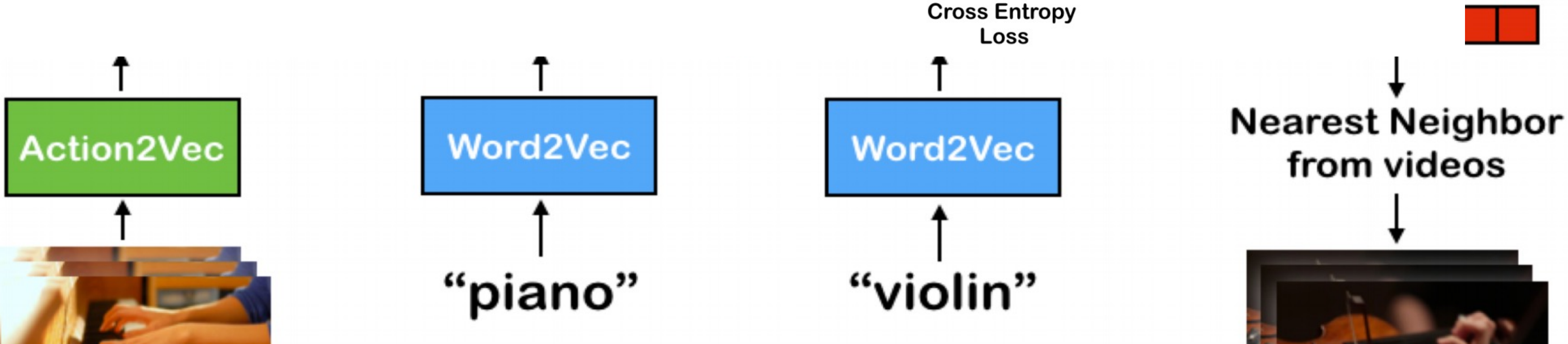
Text2shape: Generating shapes from natural language by learning joint embeddings
Chen et al, ACCV 2018

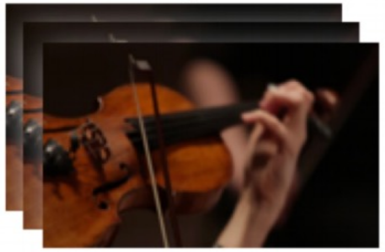
Words and actions

$$\mathcal{L}_{PR} = \min_{\theta} \sum_{i} \sum_{x} (1 - s(a_i, v_i)) + max\{0, s(a_x, v_i)\} + max\{0, s(a_i, v_x)\}$$

$$\mathcal{L}_{Dual} = \mathcal{L}_{PR} + \lambda * \mathcal{L}_{CE}$$







Action2Vec: A Crossmodal Embedding Approach to Action Learning (Hahn,et al, CVPR Deep Vision Workshop 2018)

Next time

- Paper presentations and discussion (Monday 1/25)
  - (Ke) DeVISE: A Deep Visual-Semantic Embedding Model
  - Deep Multimodal Embedding: Manipulating Novel Objects with Point-clouds, Language and Trajectories
- Paper critiques due by midnight Sunday 1/24