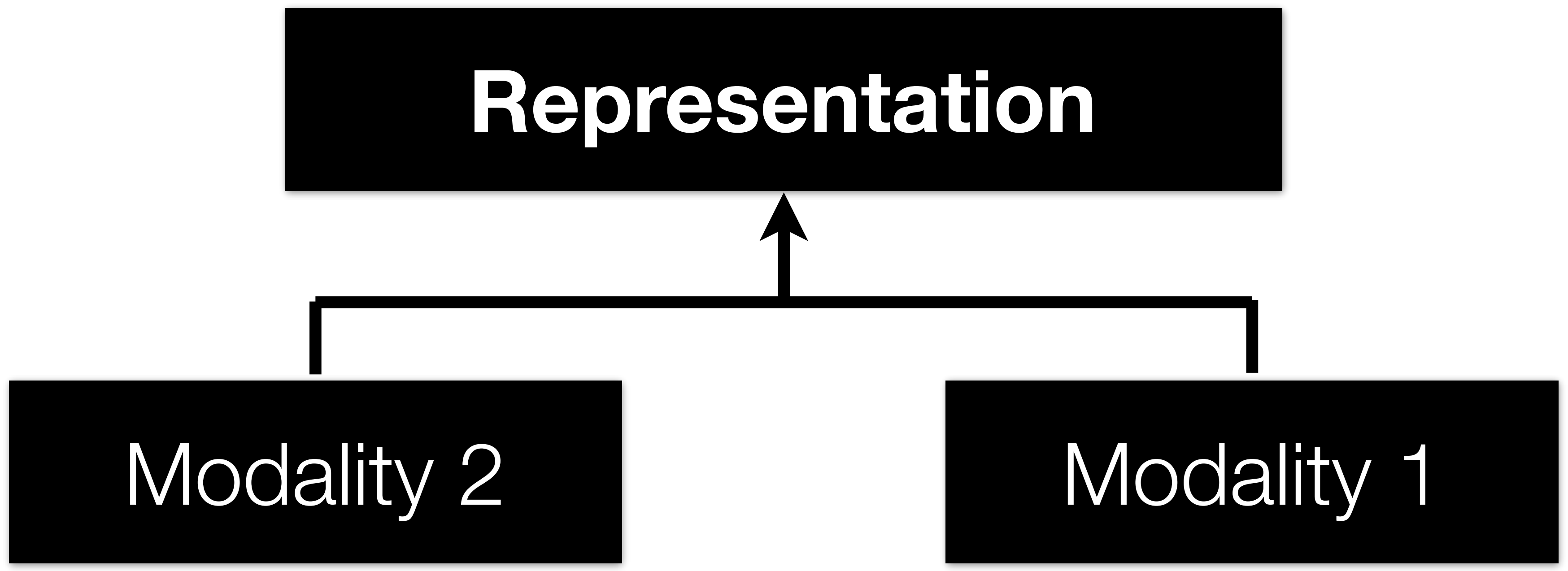
THE UNIVERSITY OF BRITISH COLUMBIA

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

Lecture 14: Coordinated Representations and Joint Embeddings [part 2]

Multimodal Representation Types

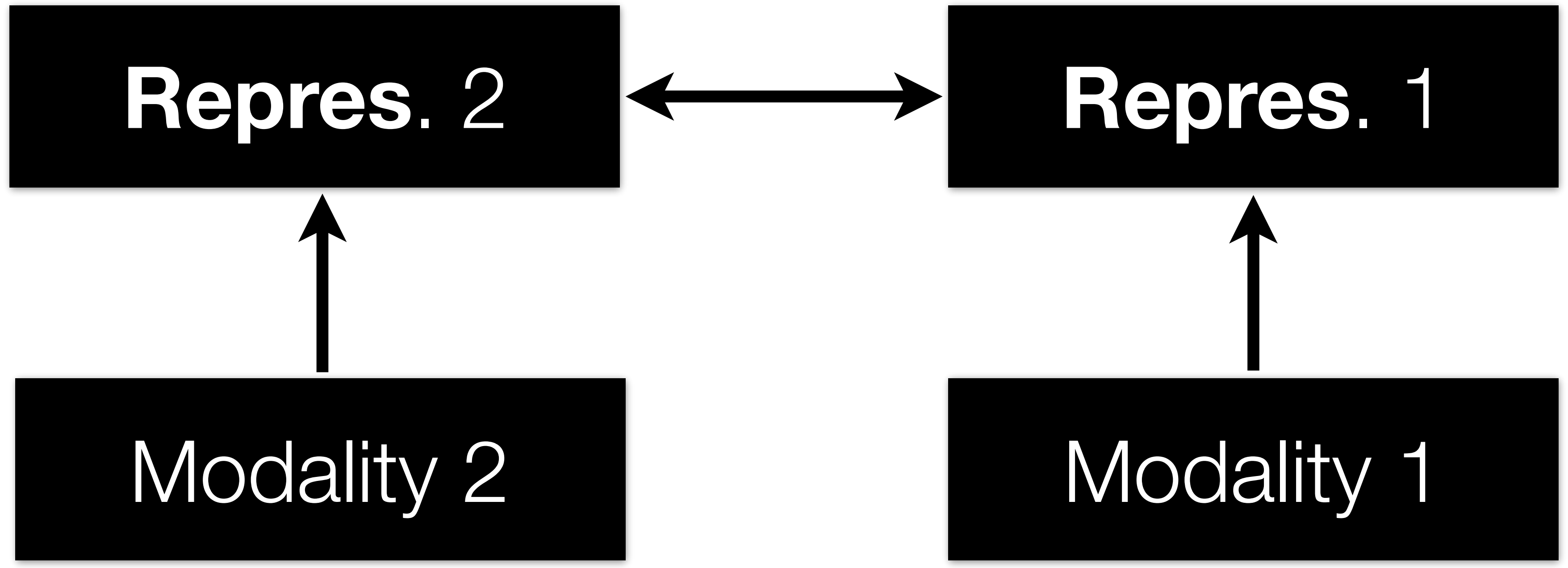
Joint representations:



— 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

Joint Representation: Deep Multimodal Autoencoders

[Ngiam et al., 2011]

Each **modality** can be pre-trained

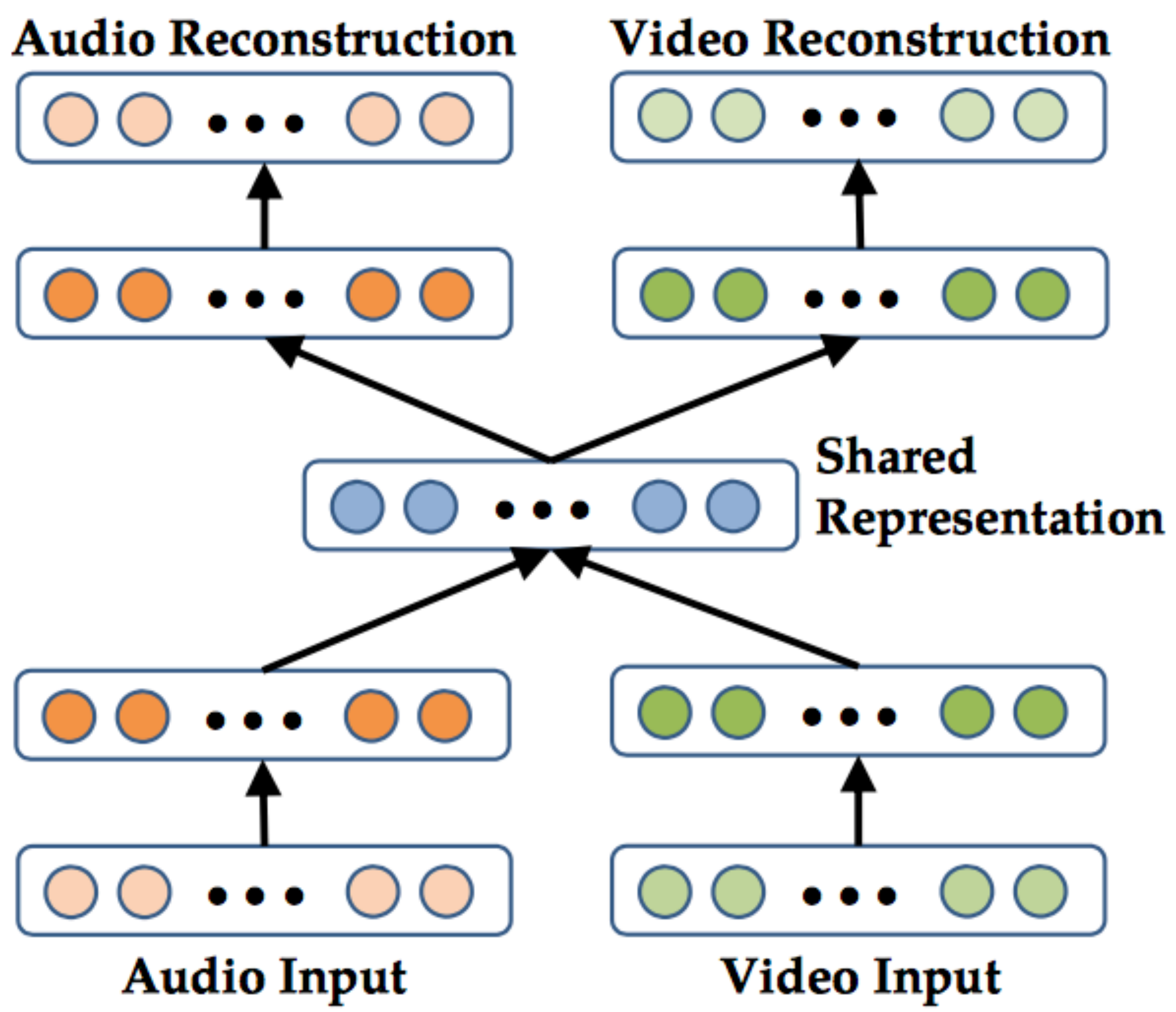
— using denoising autoencoder

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

— both Audio & Video

— just Audio

— just Video



*slide from Louis-Philippe Morency

Supervised Joint Representation

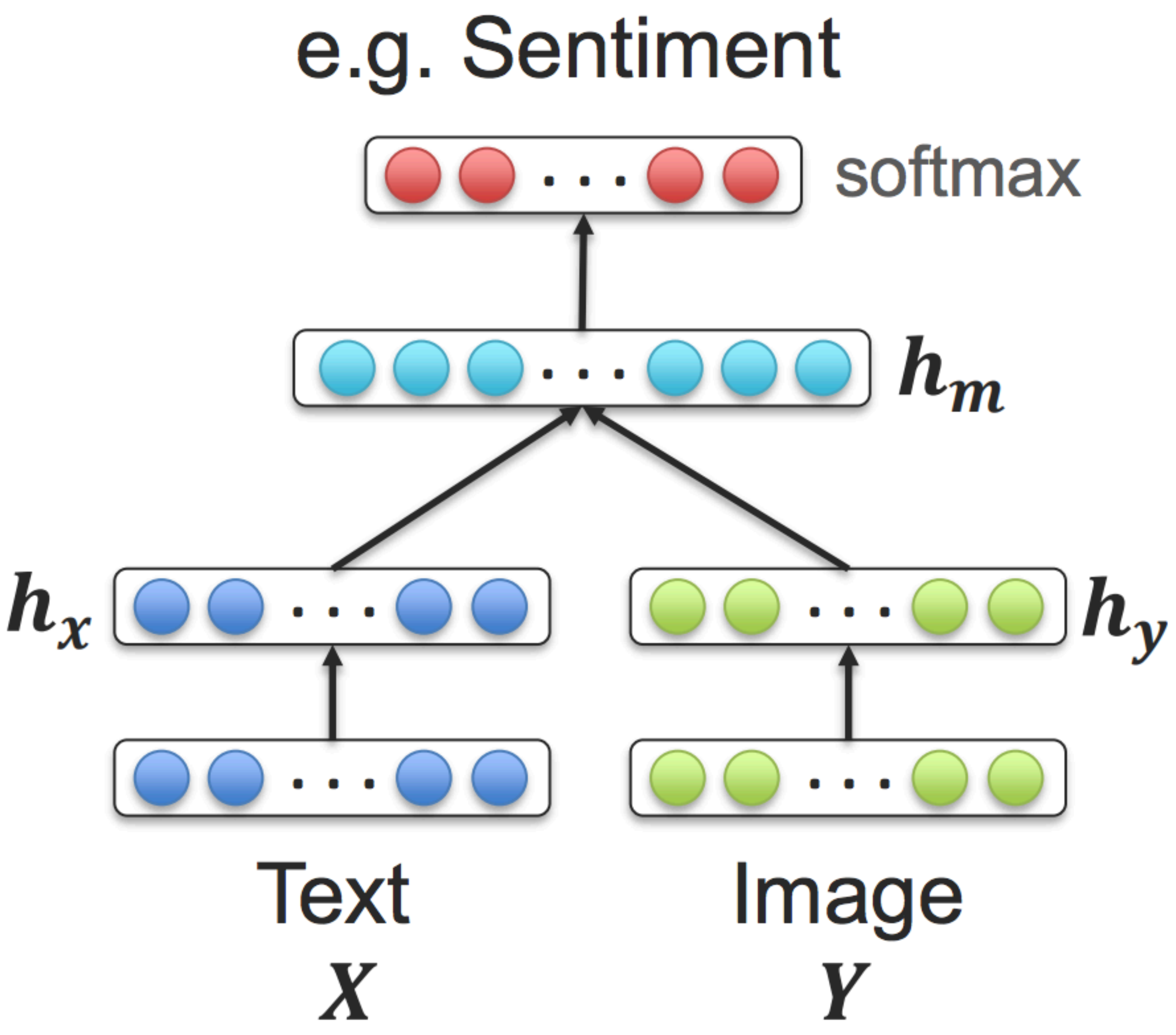
For supervised leaning tasks, we need to join unimodal representations

— Simple **concatenation**

— Element-wise **multiplicative** interactions

— many many others

Encoder-decoder Architectures



*slide from Louis-Philippe Morency

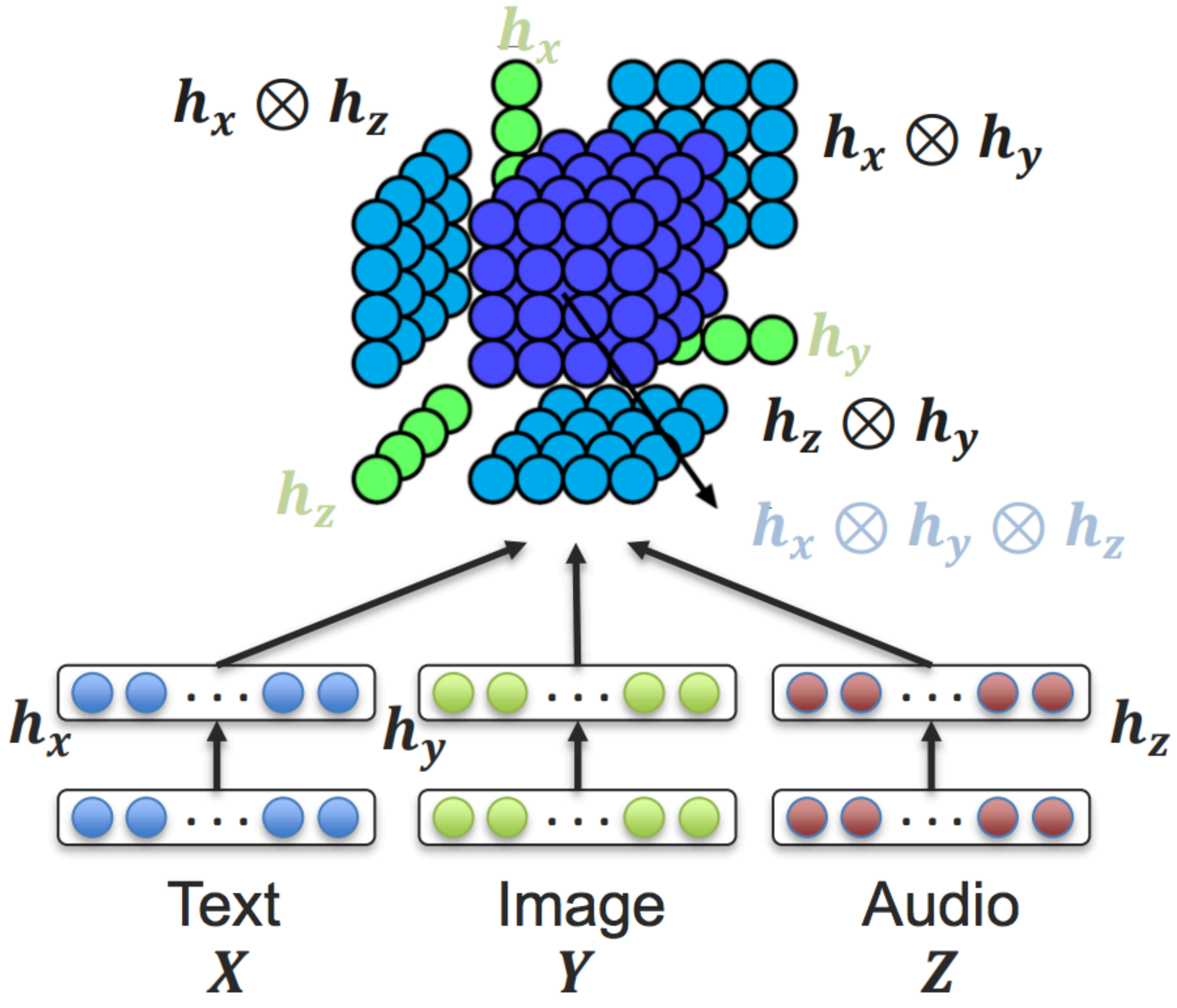
Multimodal Tensor Fusion Network (TFN)

For supervised leaning tasks, we need to join unimodal representations

— Simple **concatenation**

— Element-wise **multiplicative** interactions

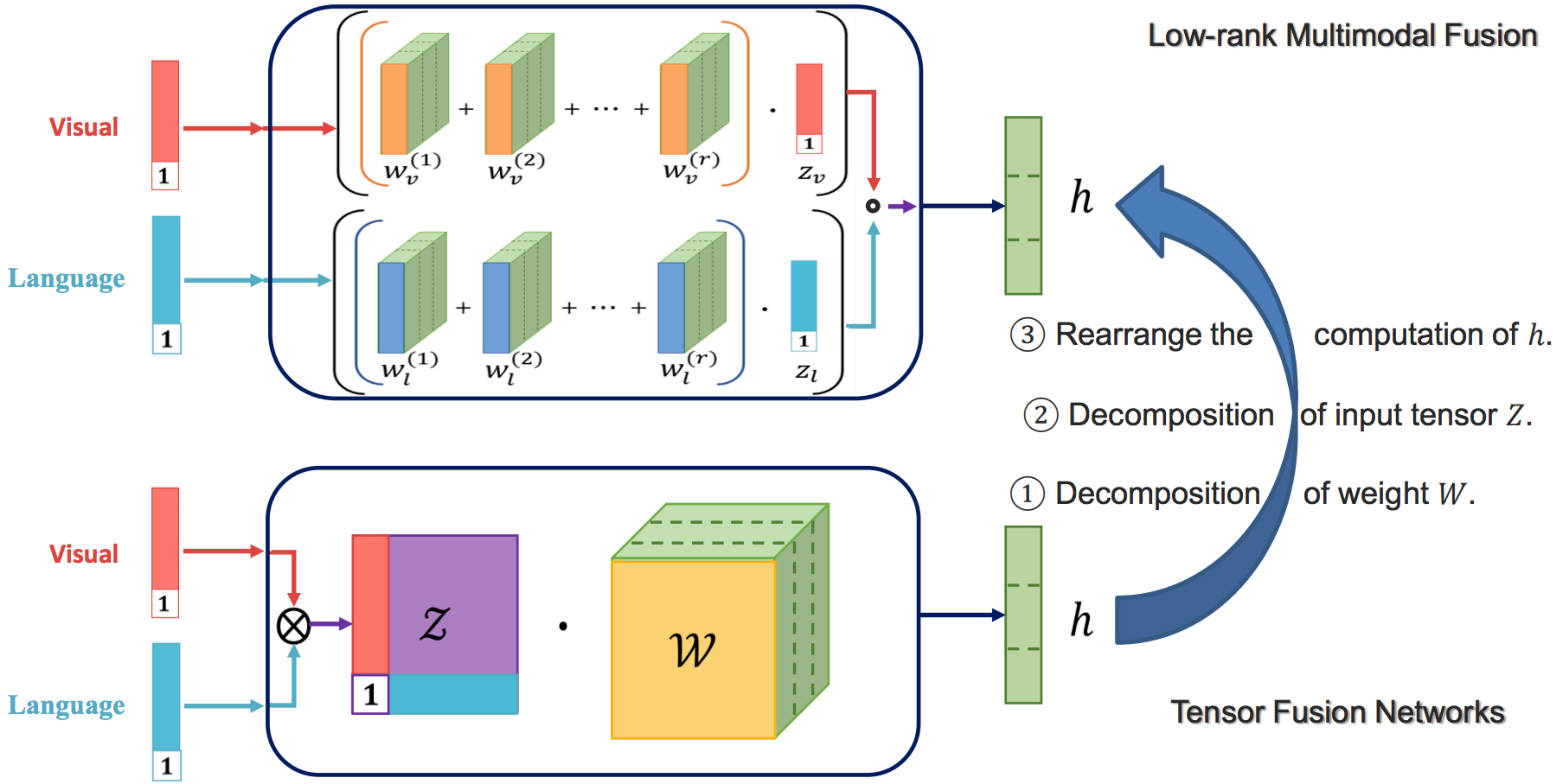
$$\mathbf{h}_m = \begin{bmatrix} \mathbf{h}_x \\ 1 \end{bmatrix} \otimes \begin{bmatrix} \mathbf{h}_y \\ 1 \end{bmatrix} \otimes \begin{bmatrix} \mathbf{h}_z \\ 1 \end{bmatrix}$$



[Zadeh, Jones and Morency, EMNLP 2017]

*slide from Louis-Philippe Morency

Low-rank Tensor Fusion



Tucker tensor decomposition leards to MUTAN fusion

[Ben-younes et al., ICCV 2017]

*slide from Louis-Philippe Morency

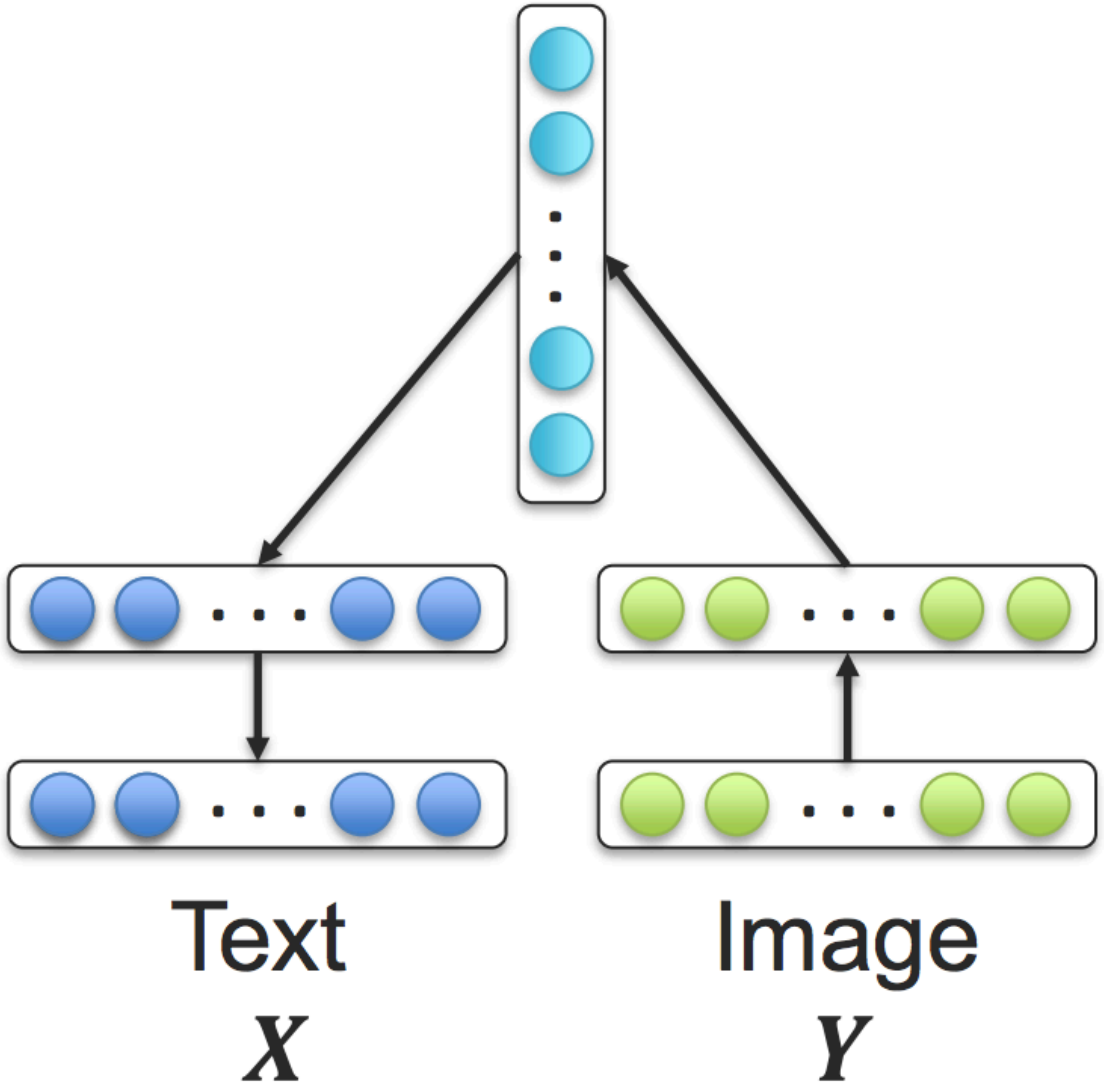
Supervised Joint Representation

For supervised leaning tasks, we need to join unimodal representations

— Simple **concatenation**

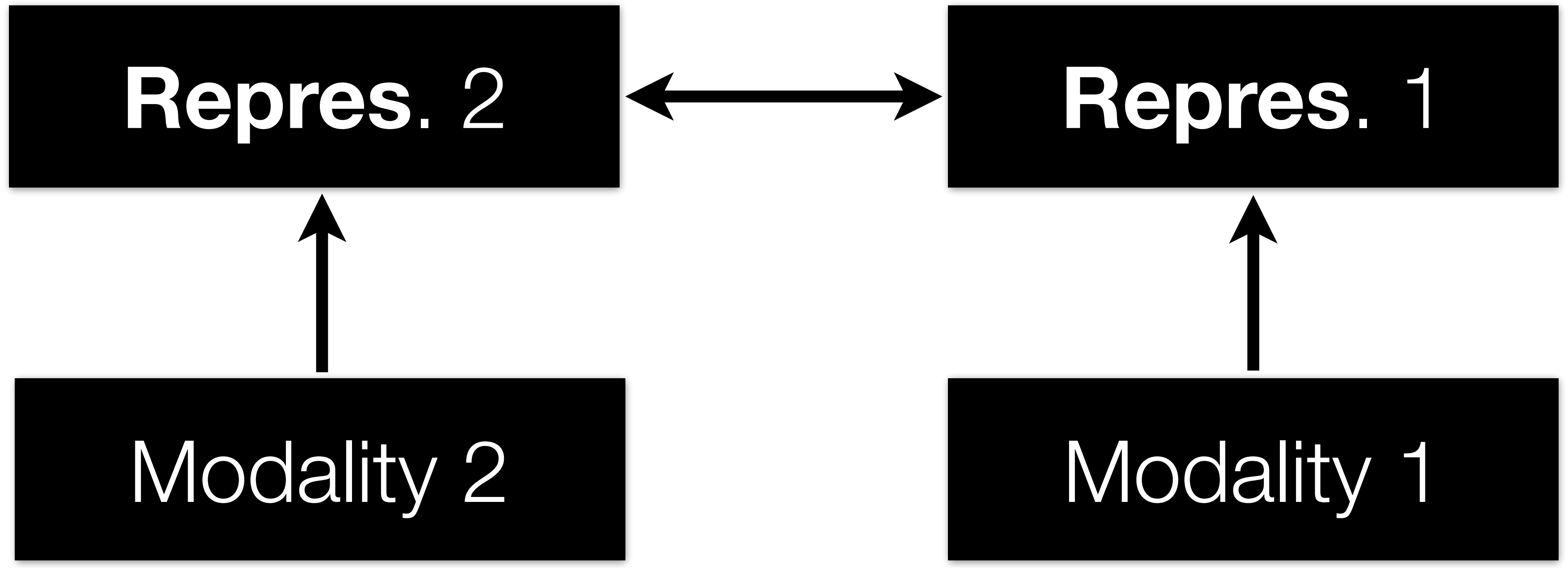
— Element-wise **multiplicative** interactions

Encoder-decoder Architectures



Multimodal Representation Types

Coordinated representations:



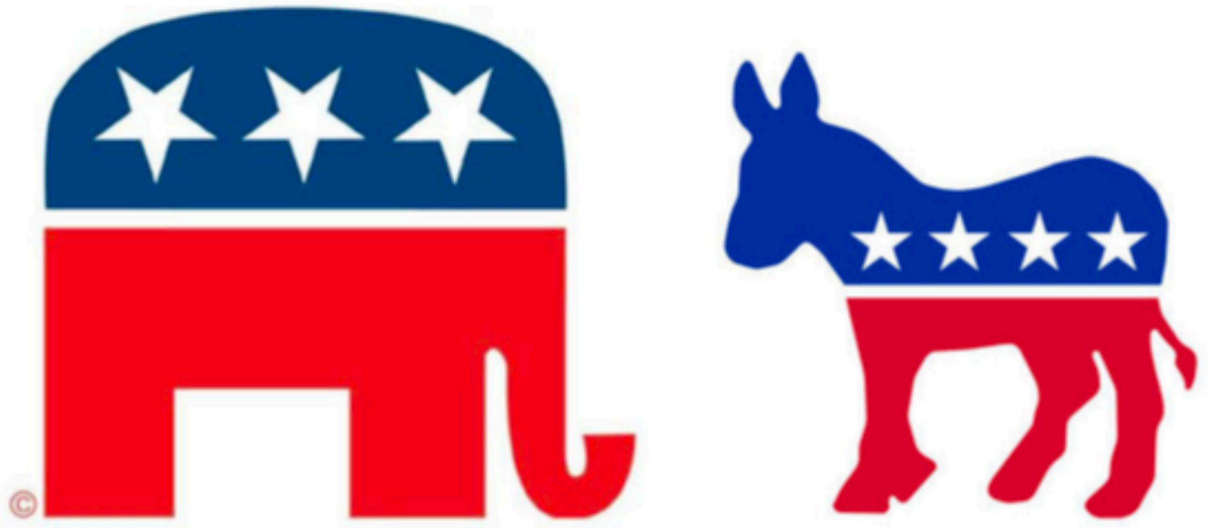
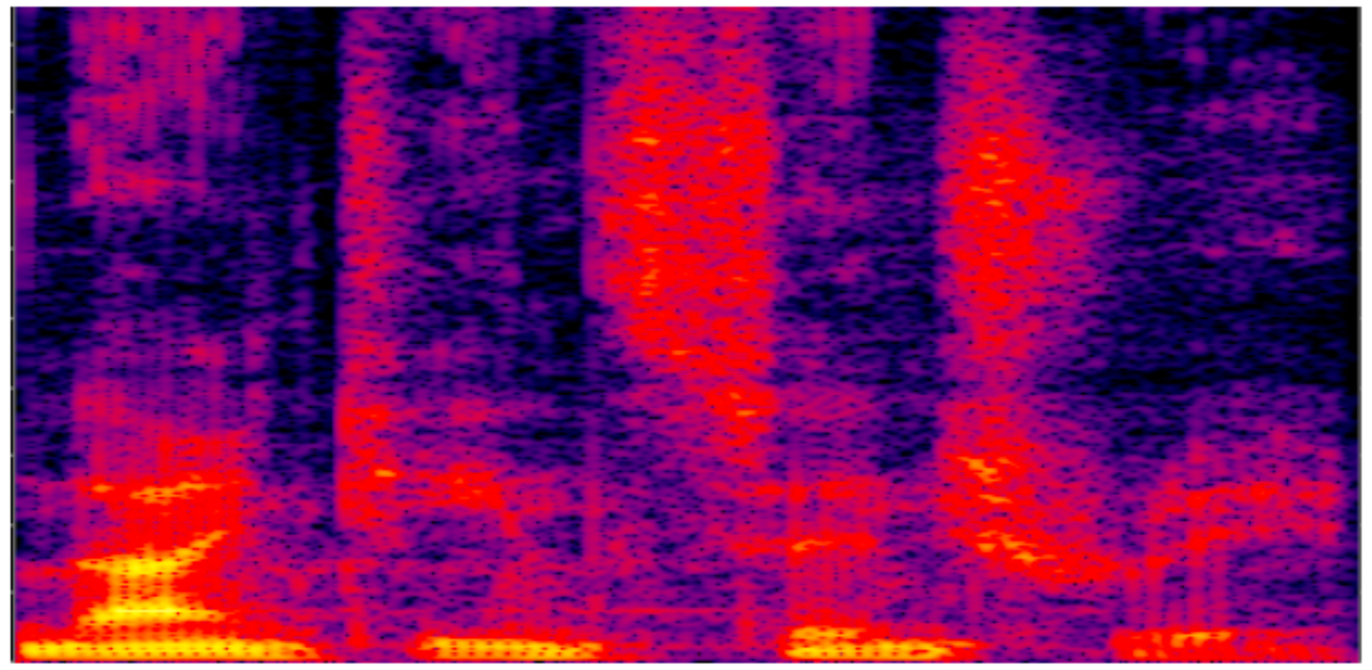
— **Similarity-based** methods (e.g., cosine distance)

— **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.,

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

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**:

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

*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**:

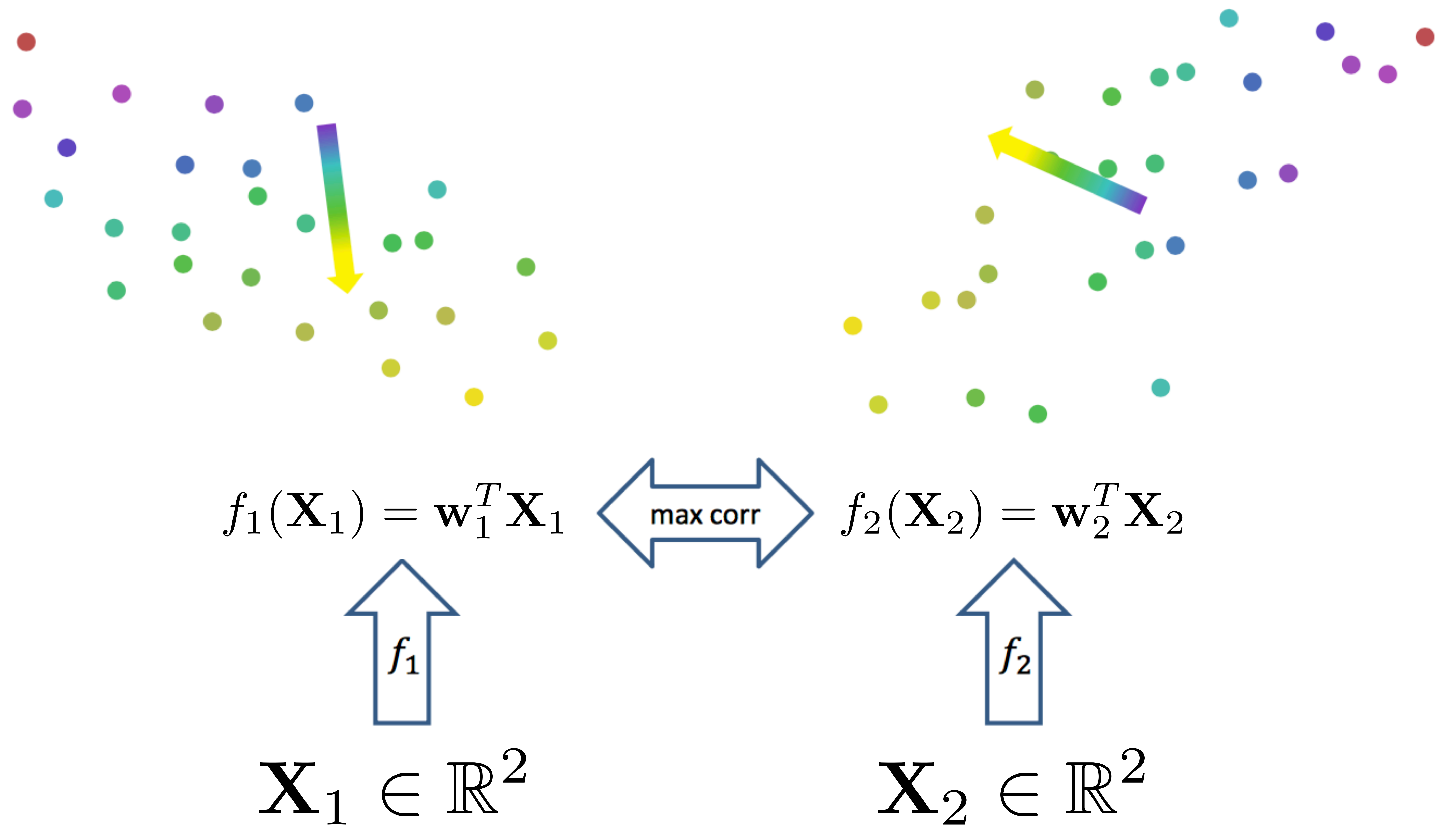
$$(\mathbf{w}_{1,:1}, \mathbf{w}_{2,:1}) = \arg\max \mathbf{corr}(\mathbf{w}_{1,:1}^T \mathbf{X}_1, \mathbf{w}_{2,:1}^T \mathbf{X}_2)$$

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:

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

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

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

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 $\mathbf{T} = \mathbf{U}\mathbf{D}\mathbf{V}^T$

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 $\mathbf{T} = \mathbf{UDV}^T$

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:

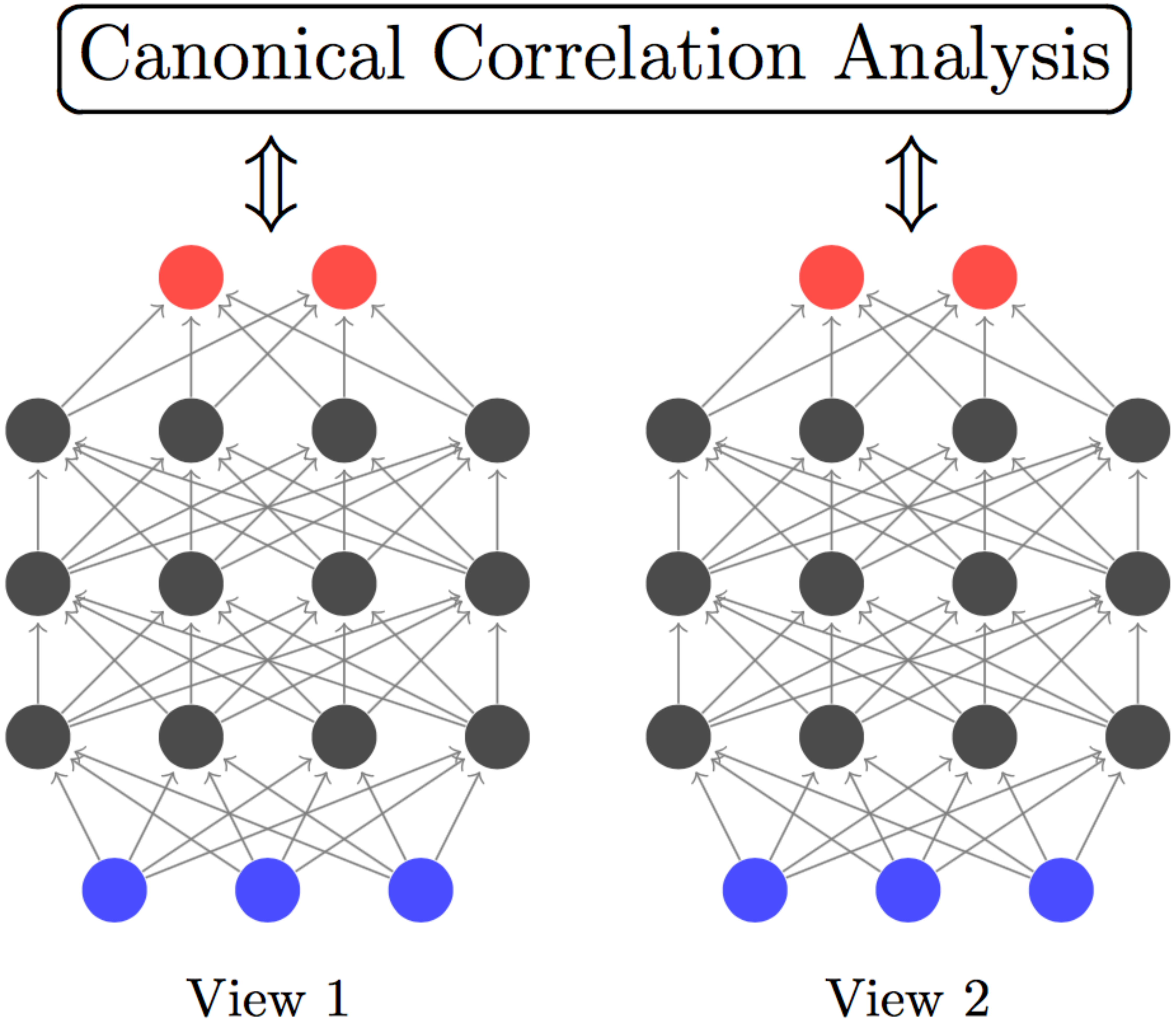
— KCCA is slower to train

— 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

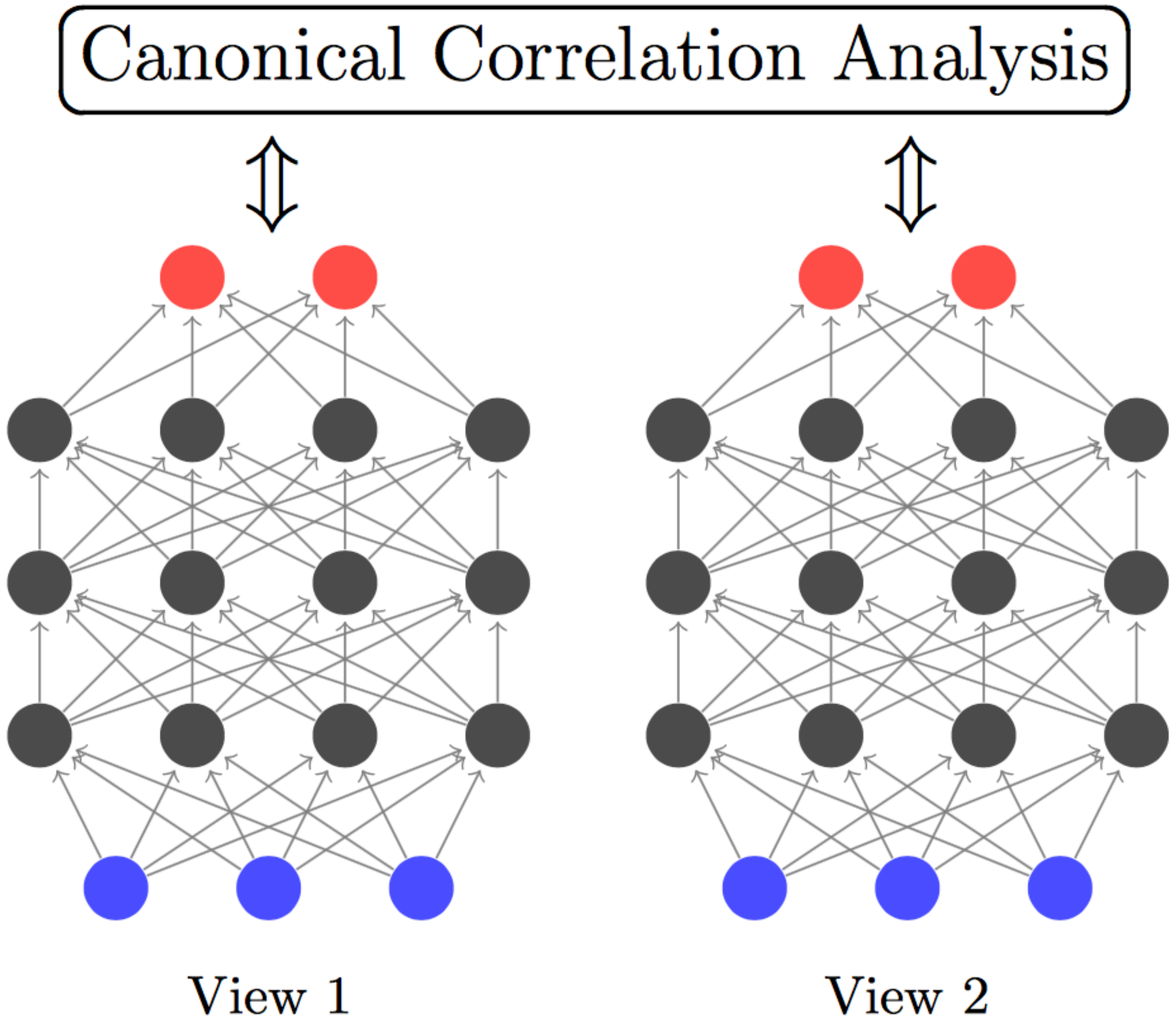
*slide from Andrew, Arora, Bilmes, Livescu

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.



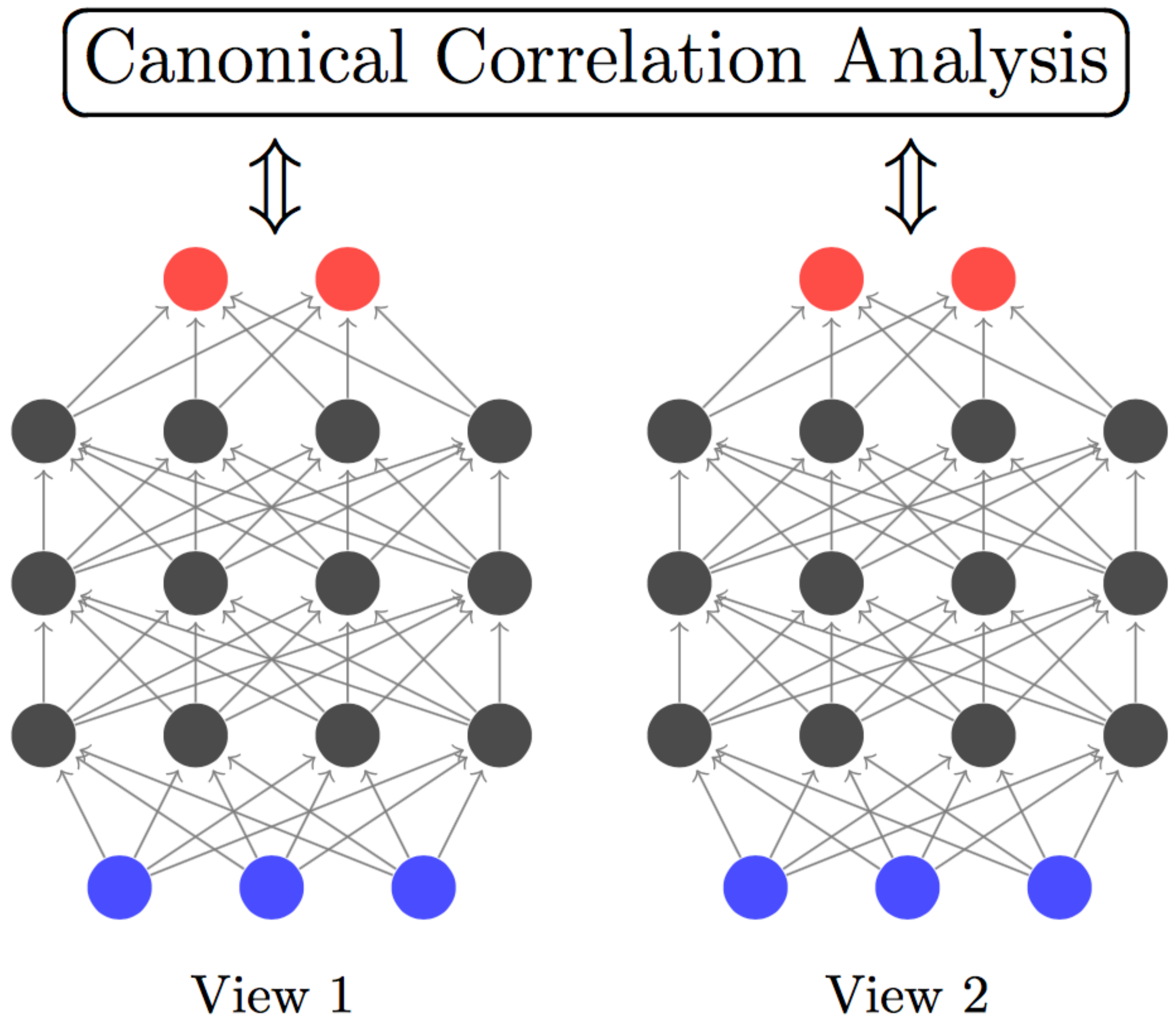
*slide from Andrew, Arora, Bilmes, Livescu

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.



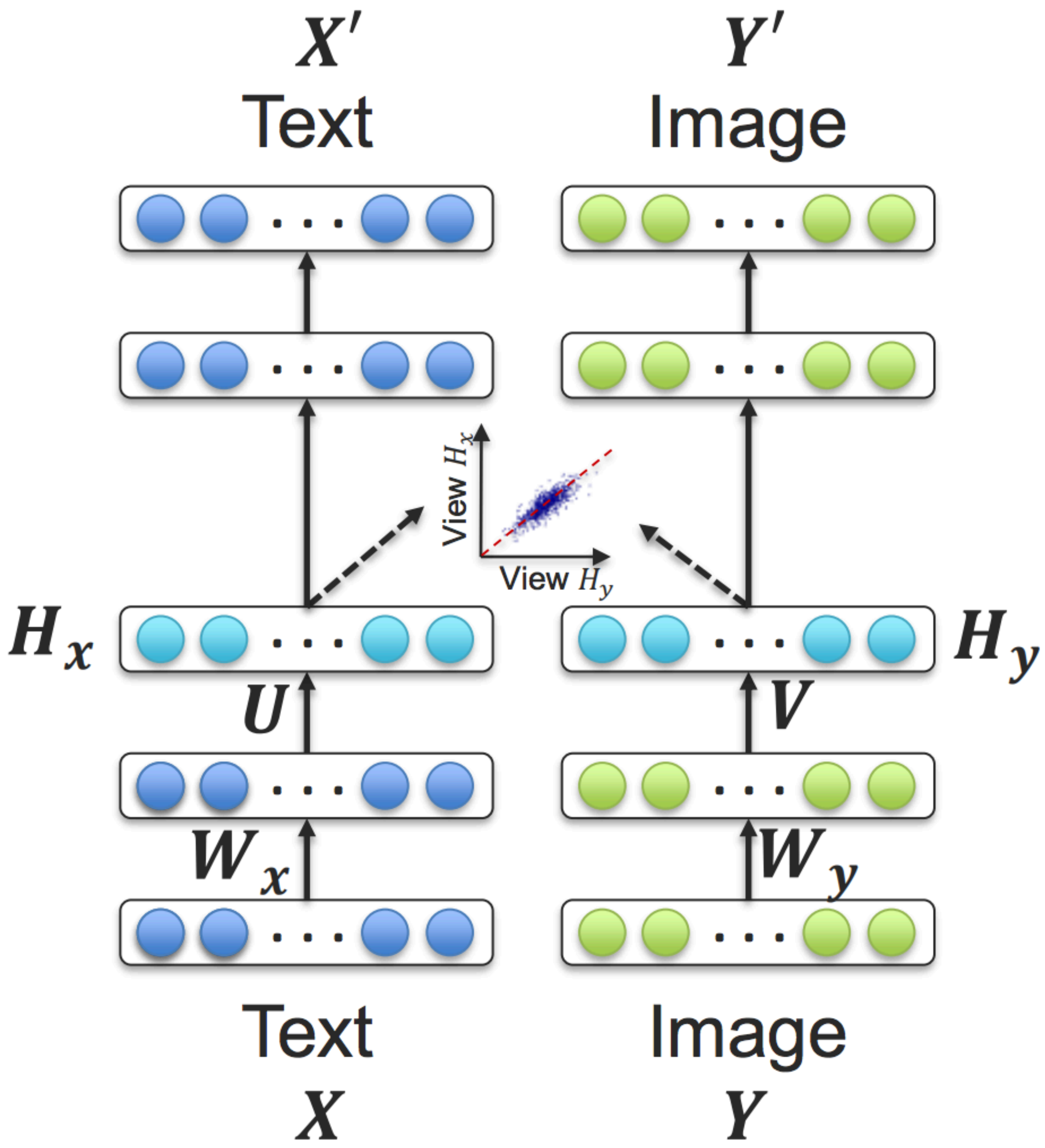
Correlation is a population objective, so instead of one instance (or minibatch) training, requires L-BFGS second-order method (with full-batch)

*slide from Andrew, Arora, Bilmes, Livescu

Deep Canonically Correlated Autoencoders (DCCAE)

Jointly optimize for DCCA and auto encoders loss functions

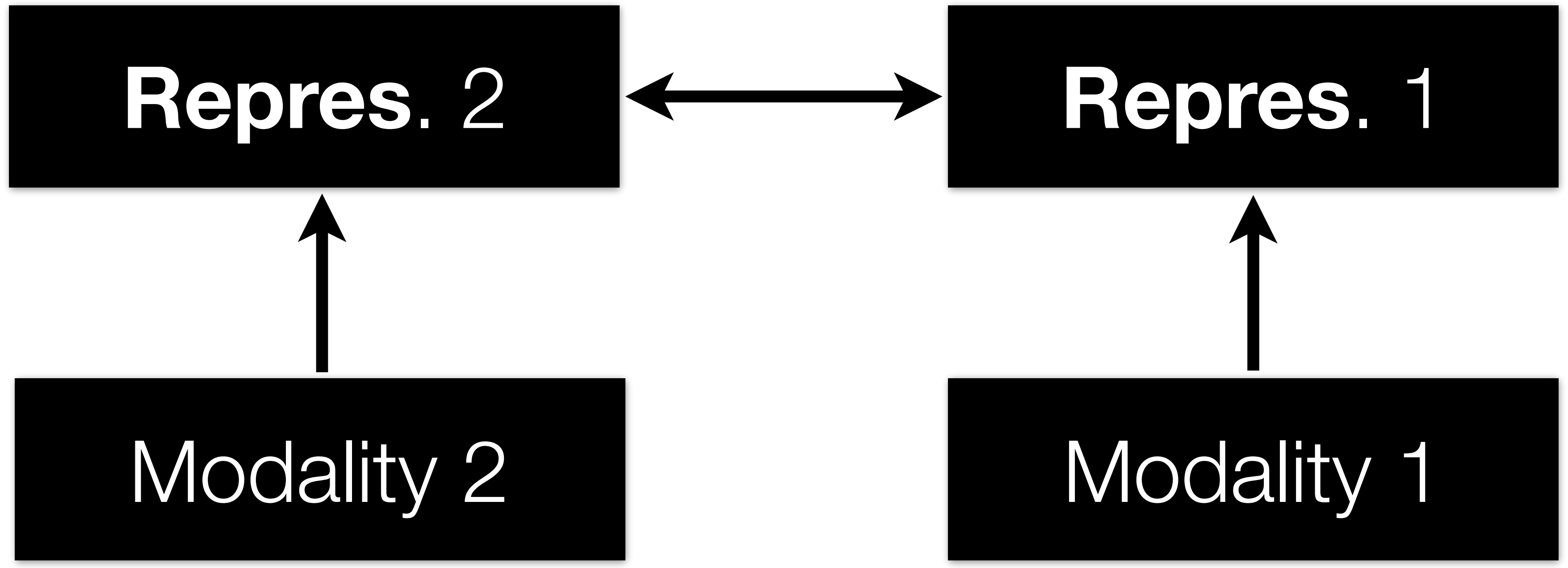
— A trade-off between multi-view correlation and reconstruction error from individual views



[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**

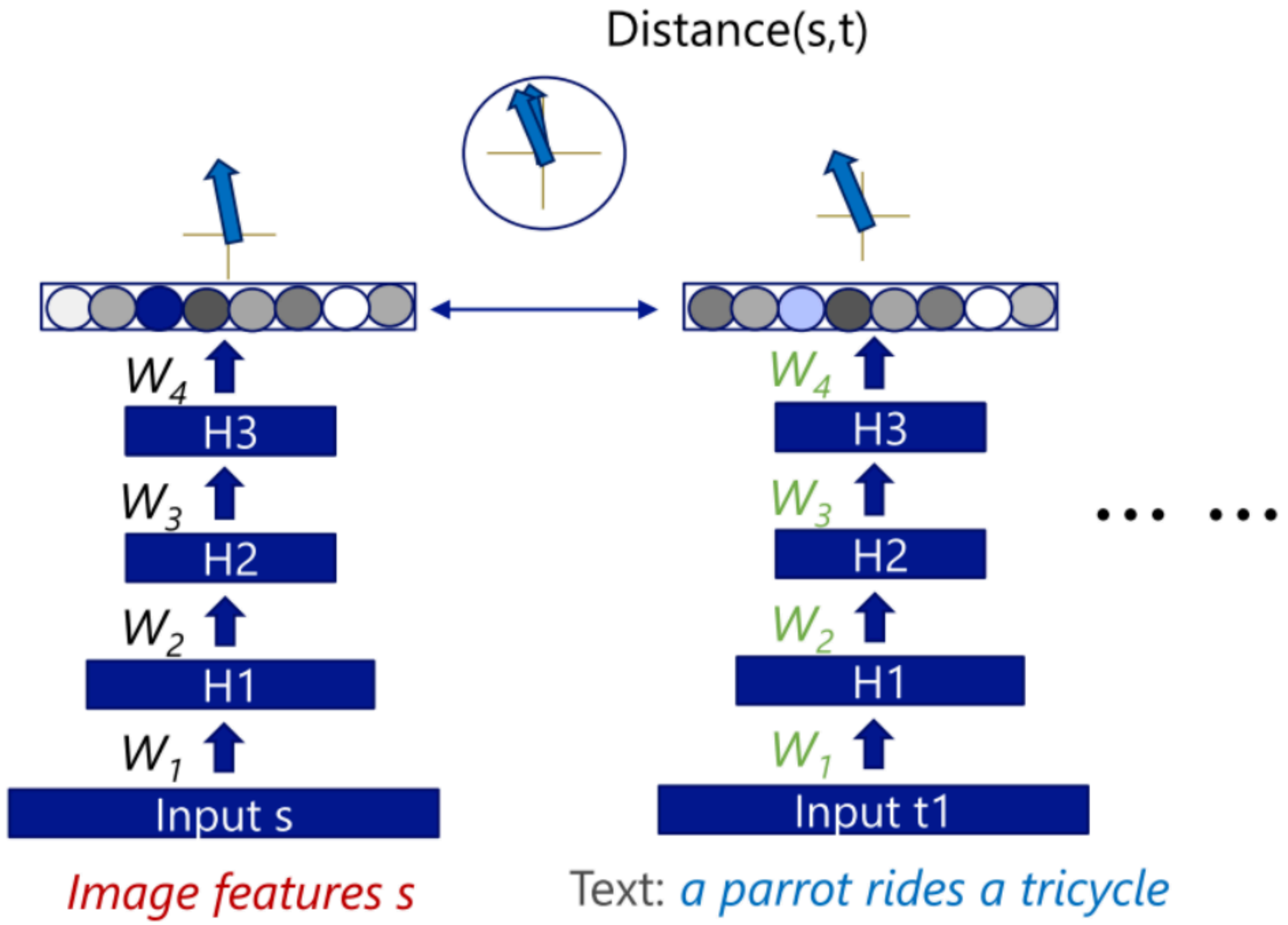
Correlated Representations: Find representations $f_1(\mathbf{x}_1), f_2(\mathbf{x}_2)$ for each view that maximize correlation:

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

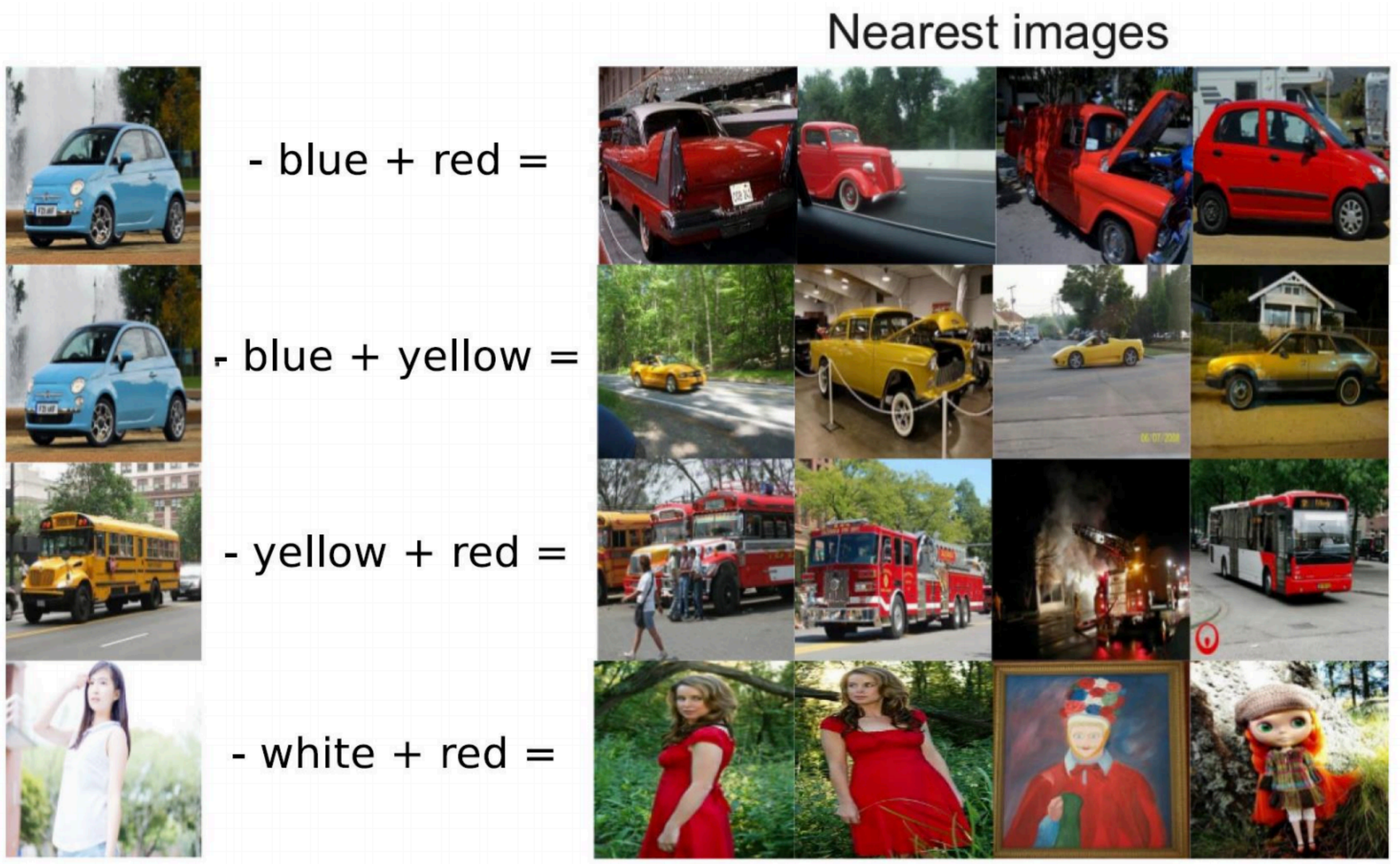
Joint Embeddings: Models that minimize distance between ground truth pairs of samples:

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

Joint **Embeddings**

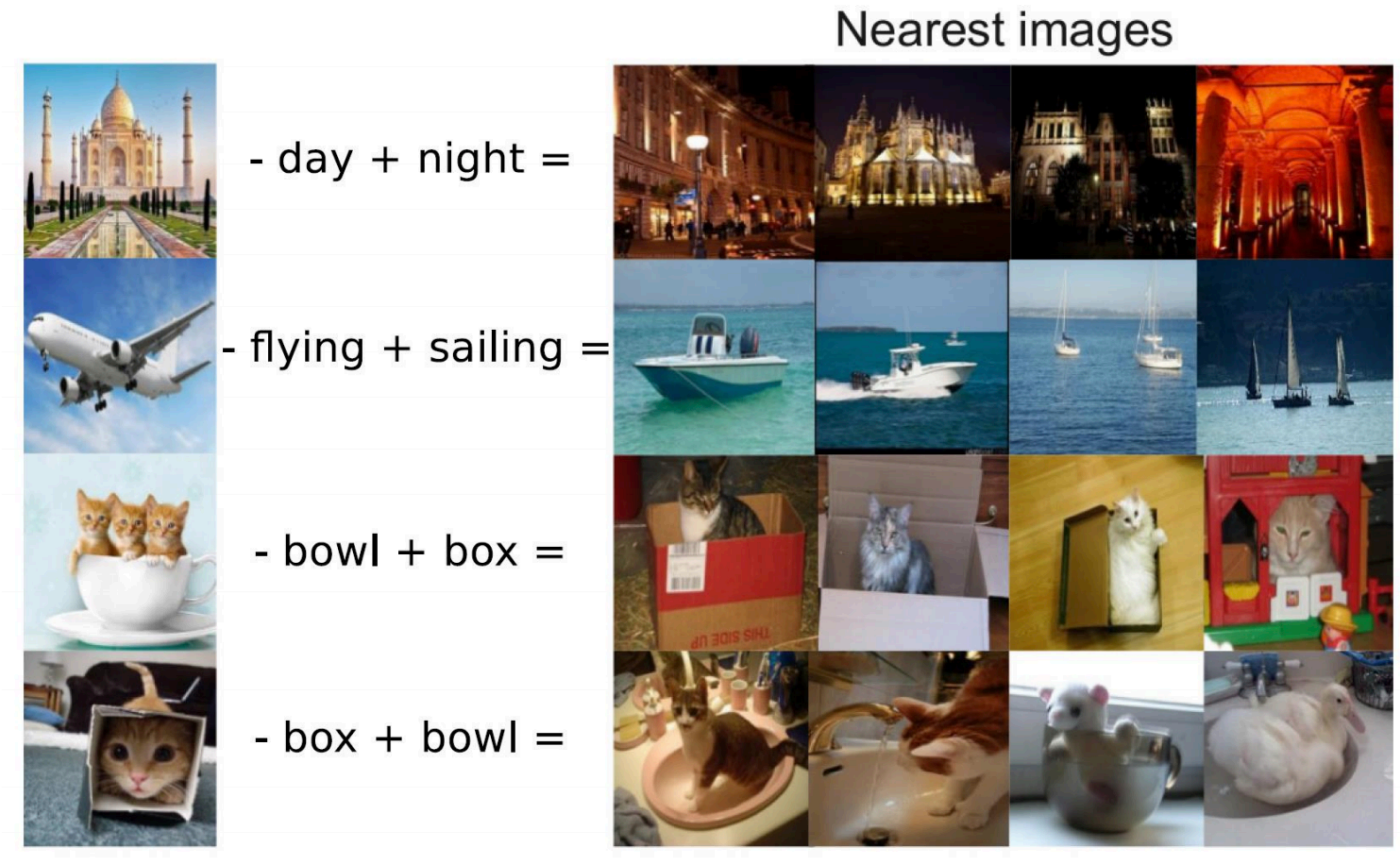


Joint **Embeddings**



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

Joint **Embeddings**



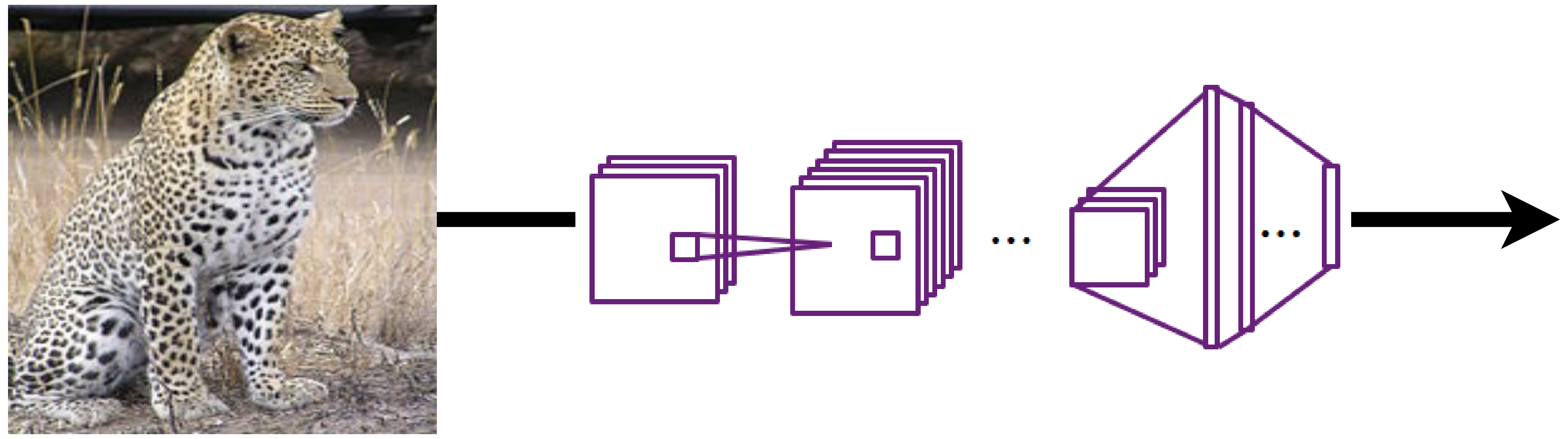
[Kiros et al., Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models, 2014]

Object **Classification**

Category	**Prediction**
Dog	No
Cat	No
Couch	No
Flowers	No
Leopard	**Yes**
…	…

Problem: For each image predict which category it belongs to out of a fixed set

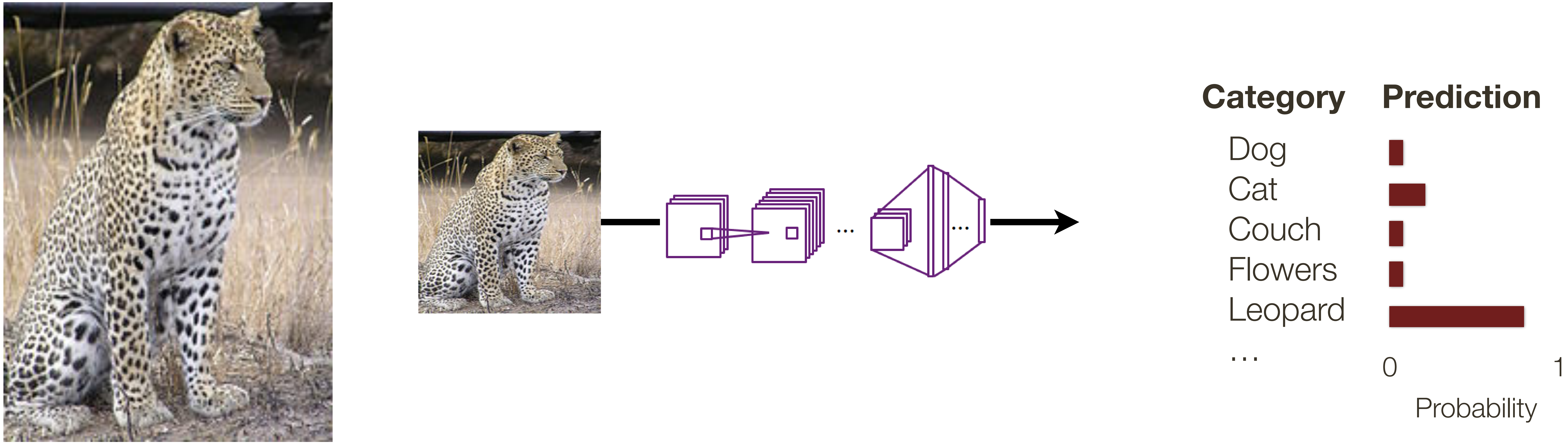
Object **Classification**



Category	Prediction
Dog	No
Cat	No
Couch	No
Flowers	No
Leopard	**Yes**
…	…

Problem: For each image predict which category it belongs to out of a fixed set

Object **Classification**



Problem: For each image predict which category it belongs to out of a fixed set

Discriminative Embeddings

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

$\mathbb{R}^d$

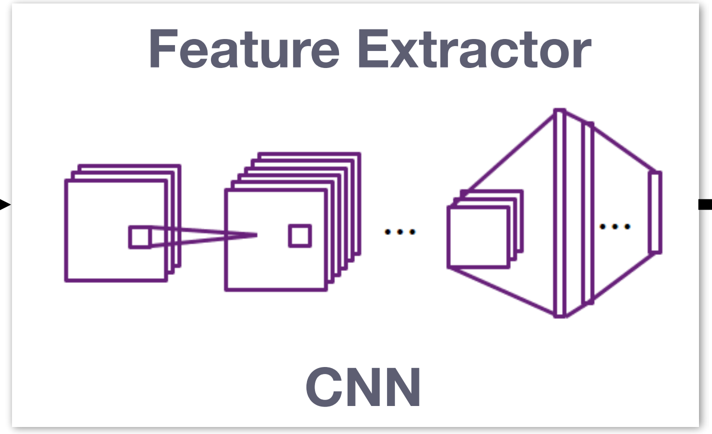
Discriminative Embeddings

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

Image Embedding

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

I_i



$CNN(I_i; \mathbf{\Theta})$

$\Psi_I(I_i)$

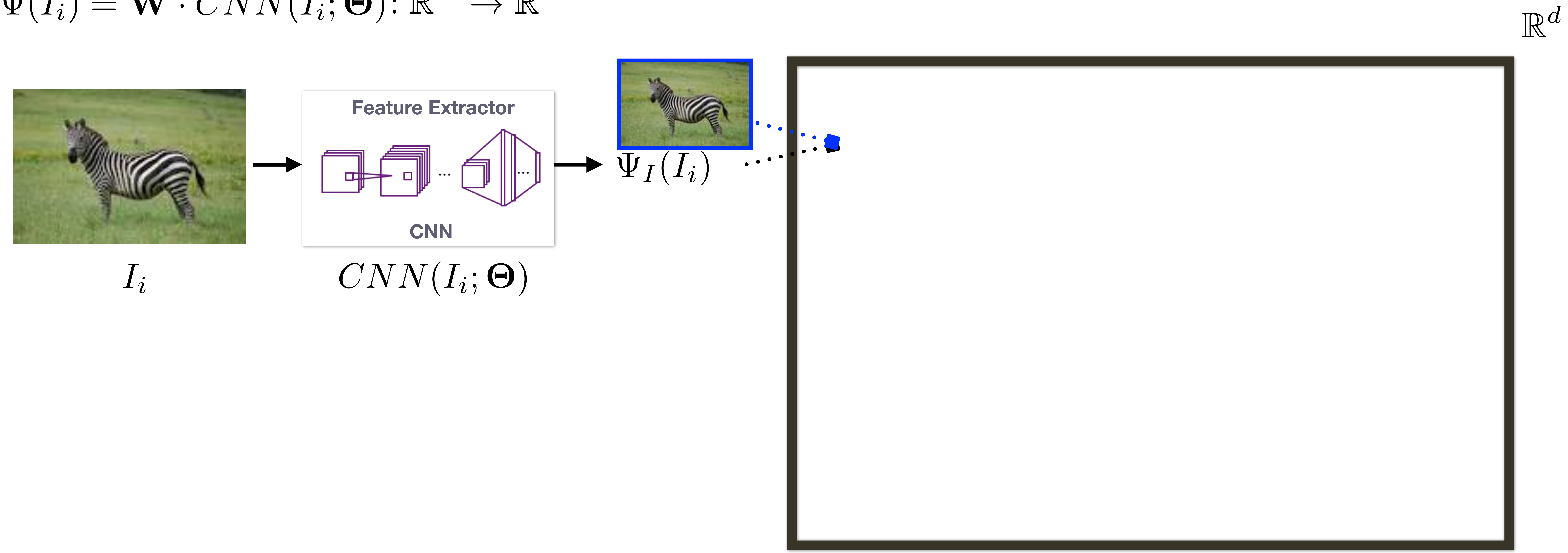
$\mathbb{R}^d$

Discriminative Embeddings

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

Image Embedding

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

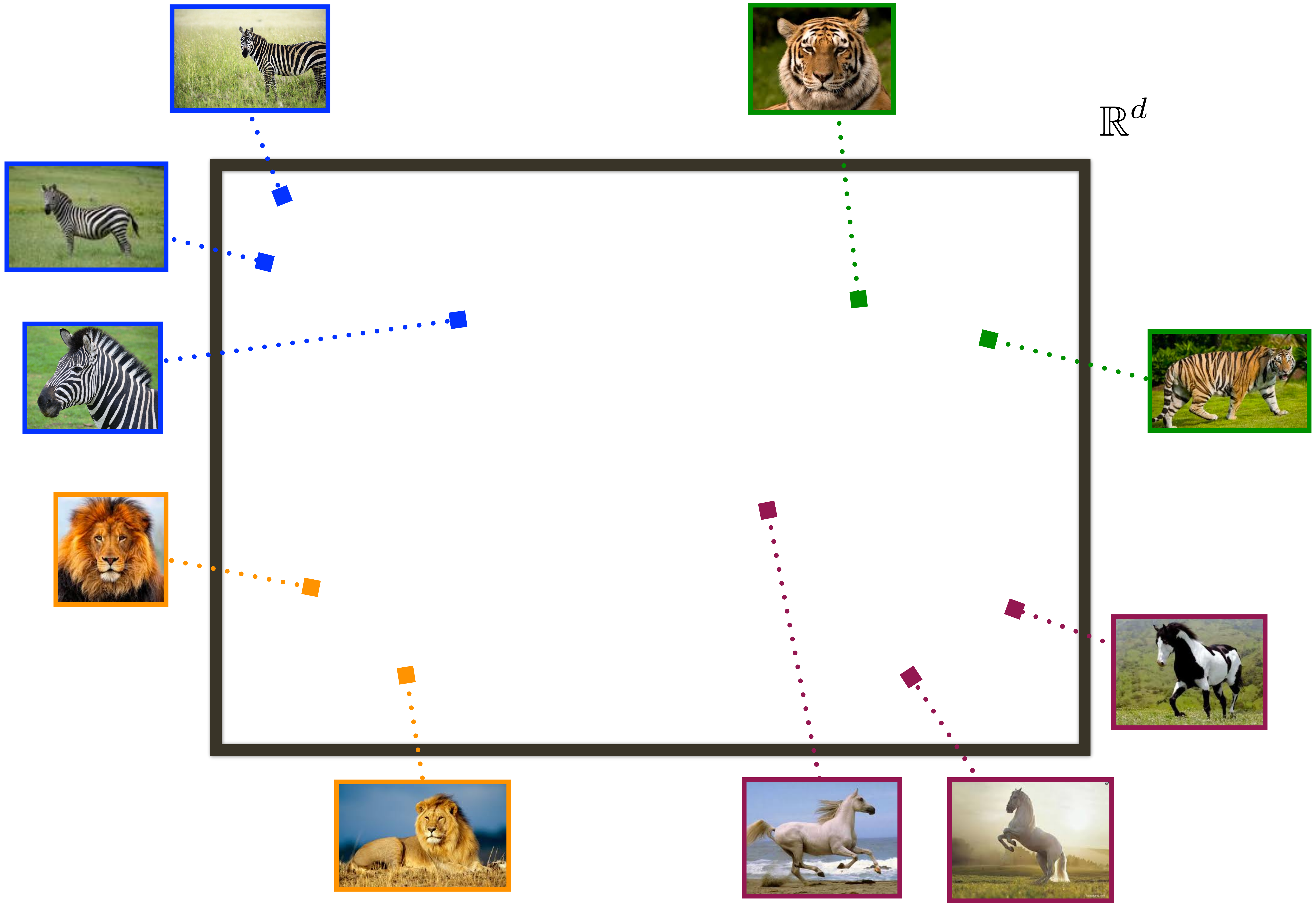


Discriminative Embeddings

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

Image Embedding

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



Discriminative Embeddings

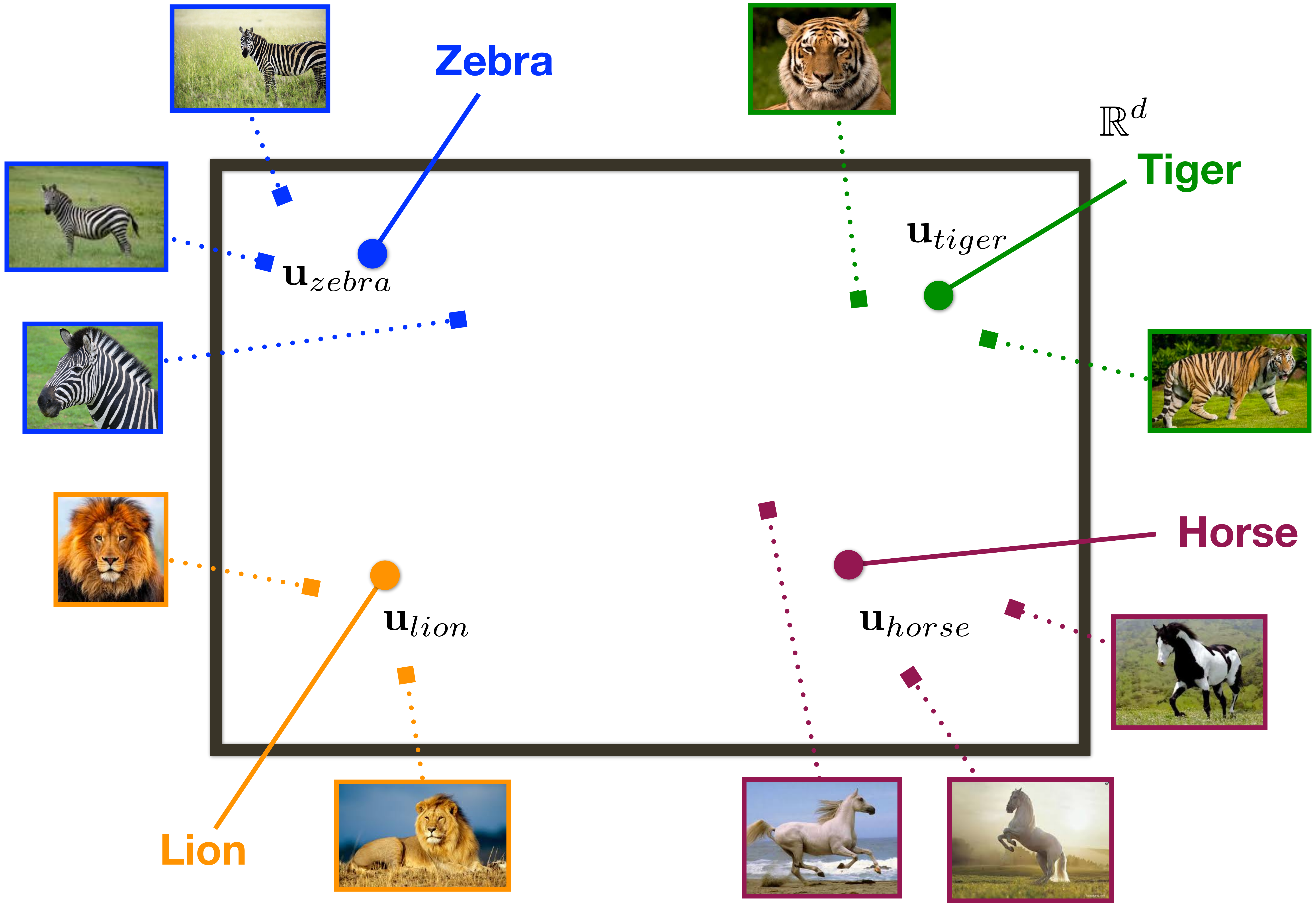
Images and **class labels** are embedded into the same space

Image Embedding

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

Label Embedding

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



Discriminative Embeddings

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

Image Embedding

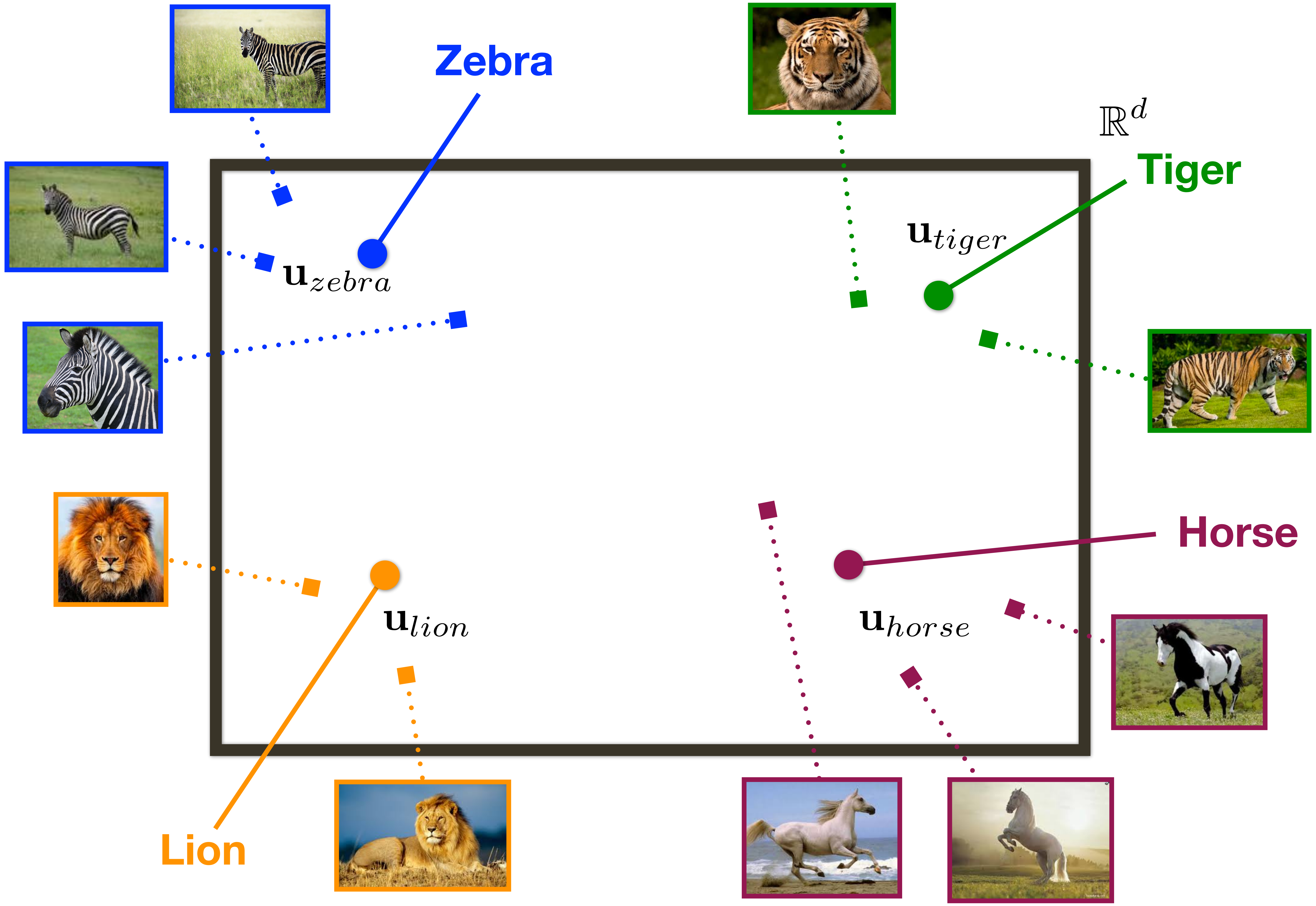
$$\Psi(I_i) = \mathbf{W} \cdot CNN(I_i; \mathbf{\Theta}) \colon \mathbb{R}^D \rightarrow \mathbb{R}^d$$

Label Embedding

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

Similarity in Embedding Space

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



Discriminative Embeddings

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

Image Embedding

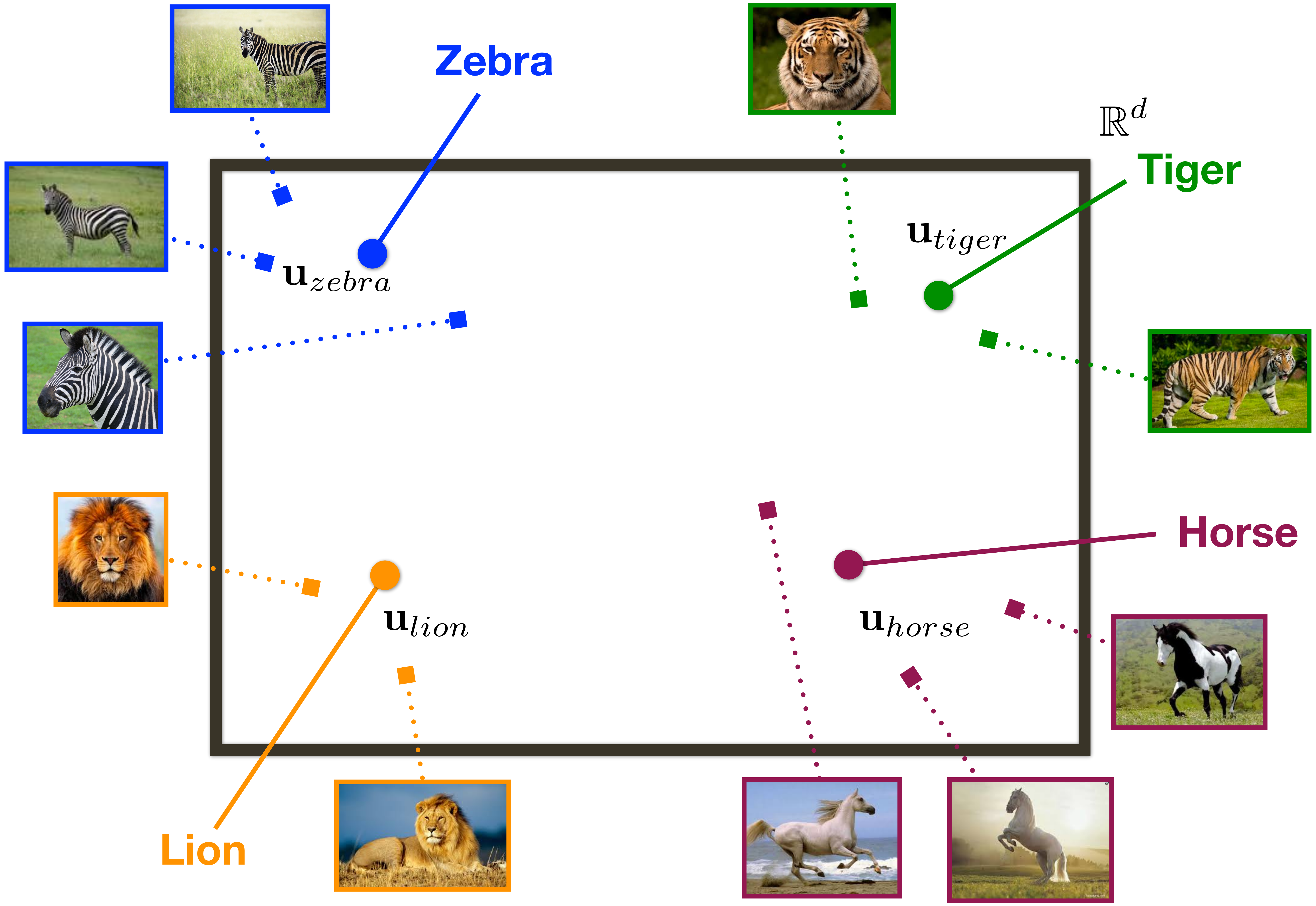
$$\Psi(I_i) = \mathbf{W} \cdot CNN(I_i; \mathbf{\Theta}) \colon \mathbb{R}^D \rightarrow \mathbb{R}^d$$

Label Embedding

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

Similarity in Embedding Space

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



Discriminative Embeddings

Image Categorization / Annotation

which object category does image belong to?

Image Embedding

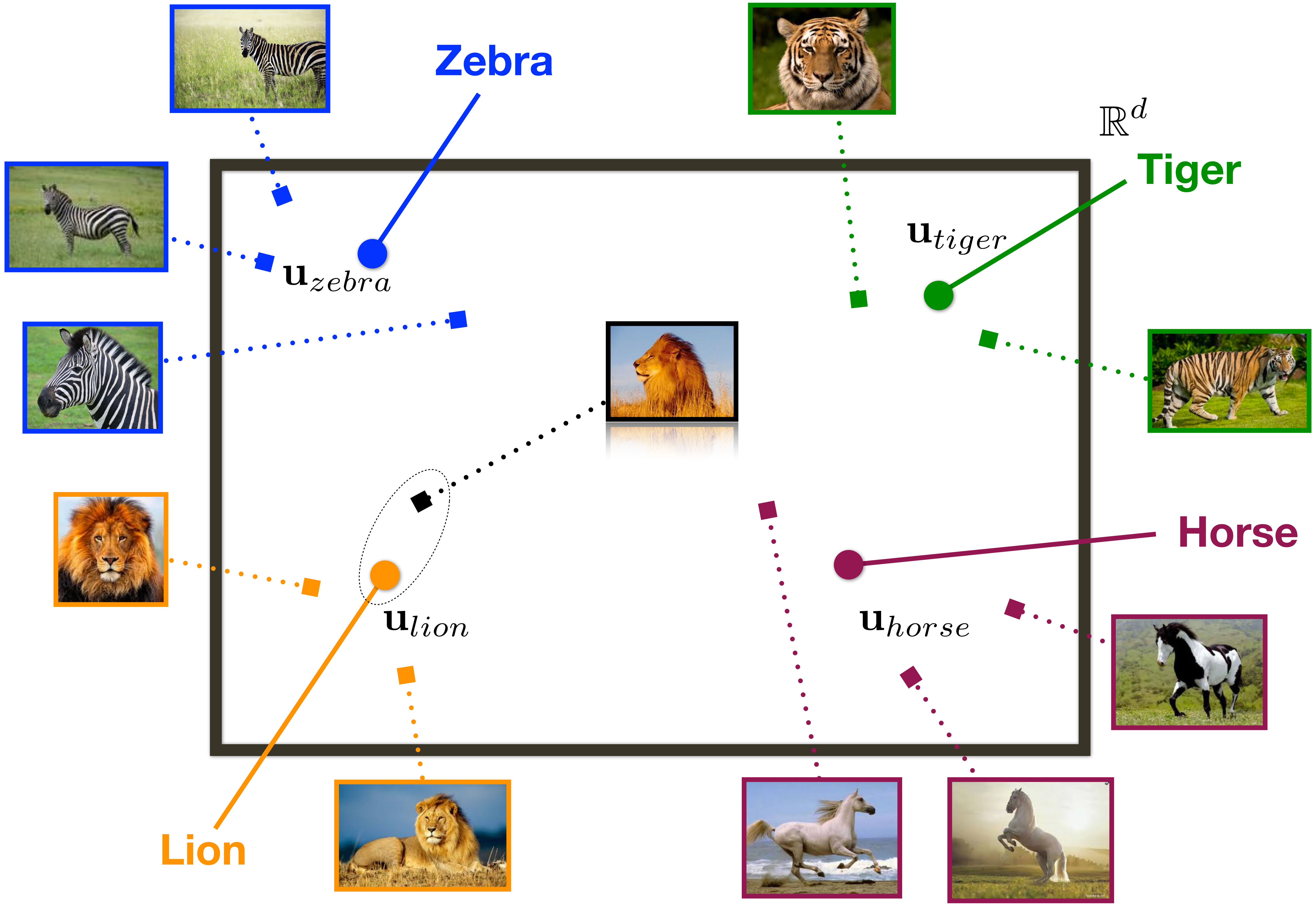
$$\Psi(I_i) = \mathbf{W} \cdot CNN(I_i; \mathbf{\Theta}) : \mathbb{R}^D \rightarrow \mathbb{R}^d$$

Label Embedding

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

Similarity in Embedding Space

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



Discriminative Embeddings

Image Embedding

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

Label Embedding

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

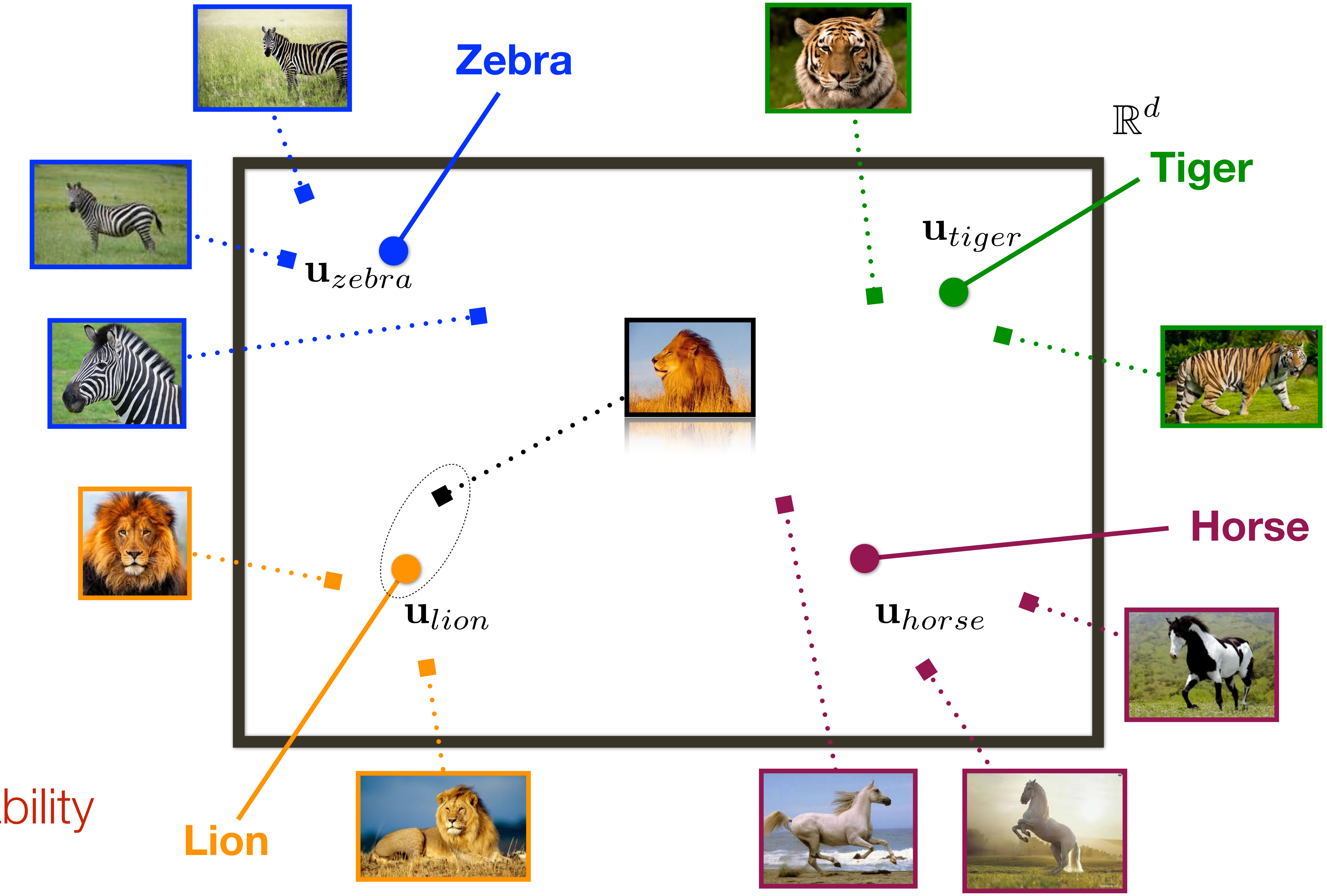
Similarity in Embedding Space

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

Distance can be interpreted as probability

Image Categorization / Annotation

which object category does image belong to?



Discriminative Embeddings

Search by Image

most similar image to a query?

Image Embedding

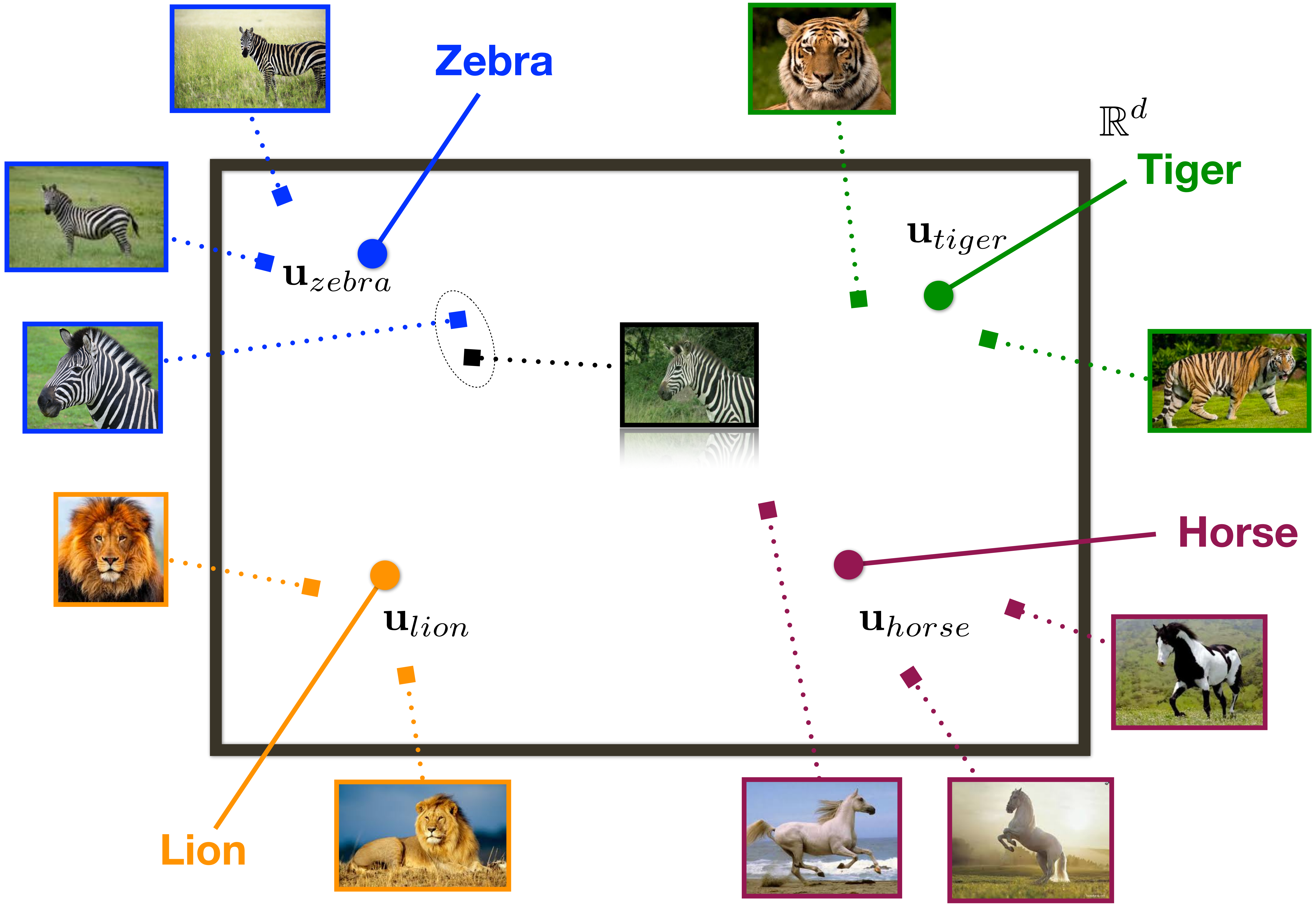
$$\Psi(I_i) = \mathbf{W} \cdot CNN(I_i; \mathbf{\Theta}) \colon \mathbb{R}^D \rightarrow \mathbb{R}^d$$

Label Embedding

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

Similarity in Embedding Space

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



Discriminative Embeddings

Image Embedding

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

Label Embedding

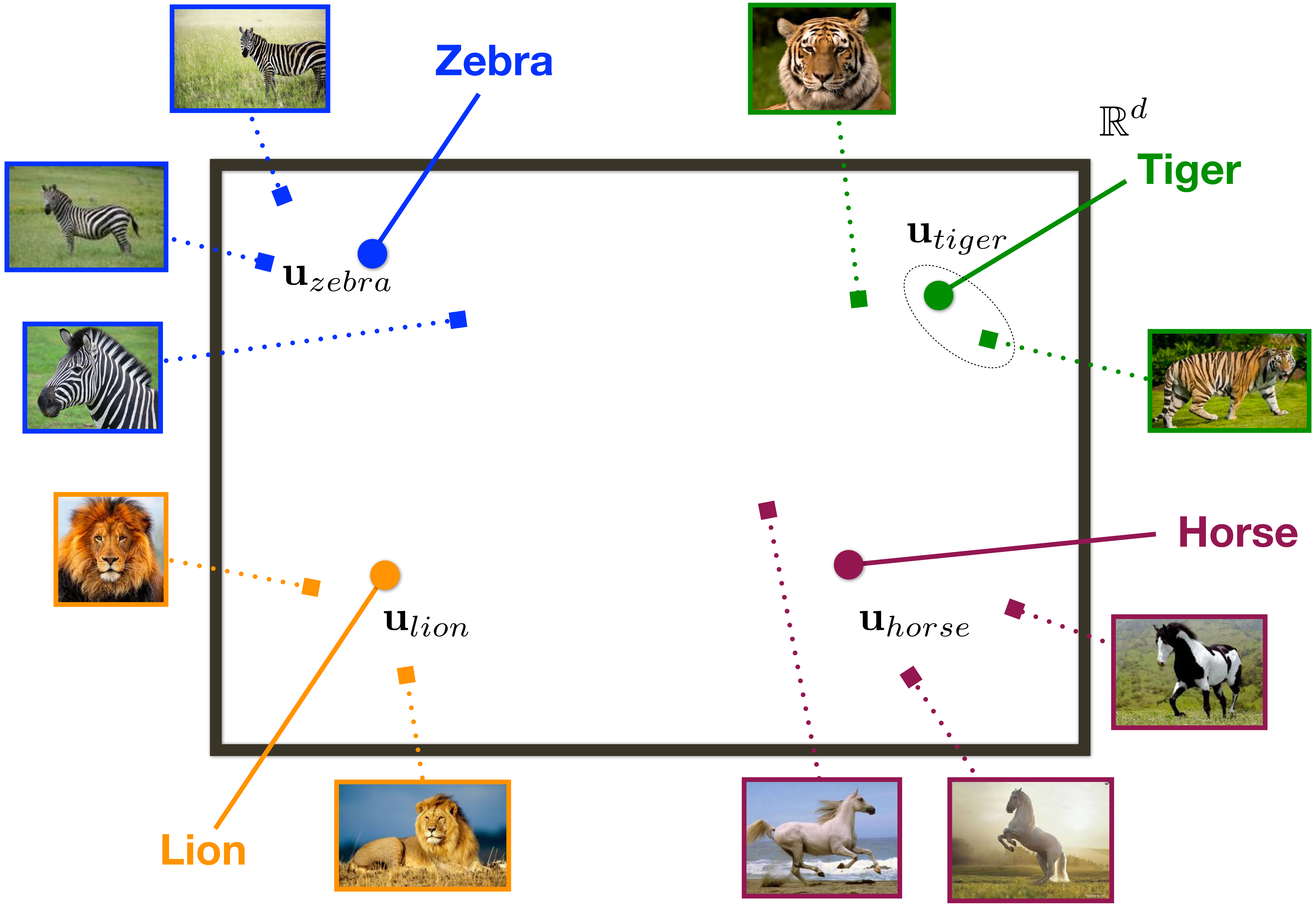
$$\Psi_L(word_i) = \mathbf{u}_i : \{1, ..., L\} \to \mathbb{R}^d$$

Similarity in Embedding Space

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

Search by Label

most representative image for a label?



Discriminative Embeddings

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

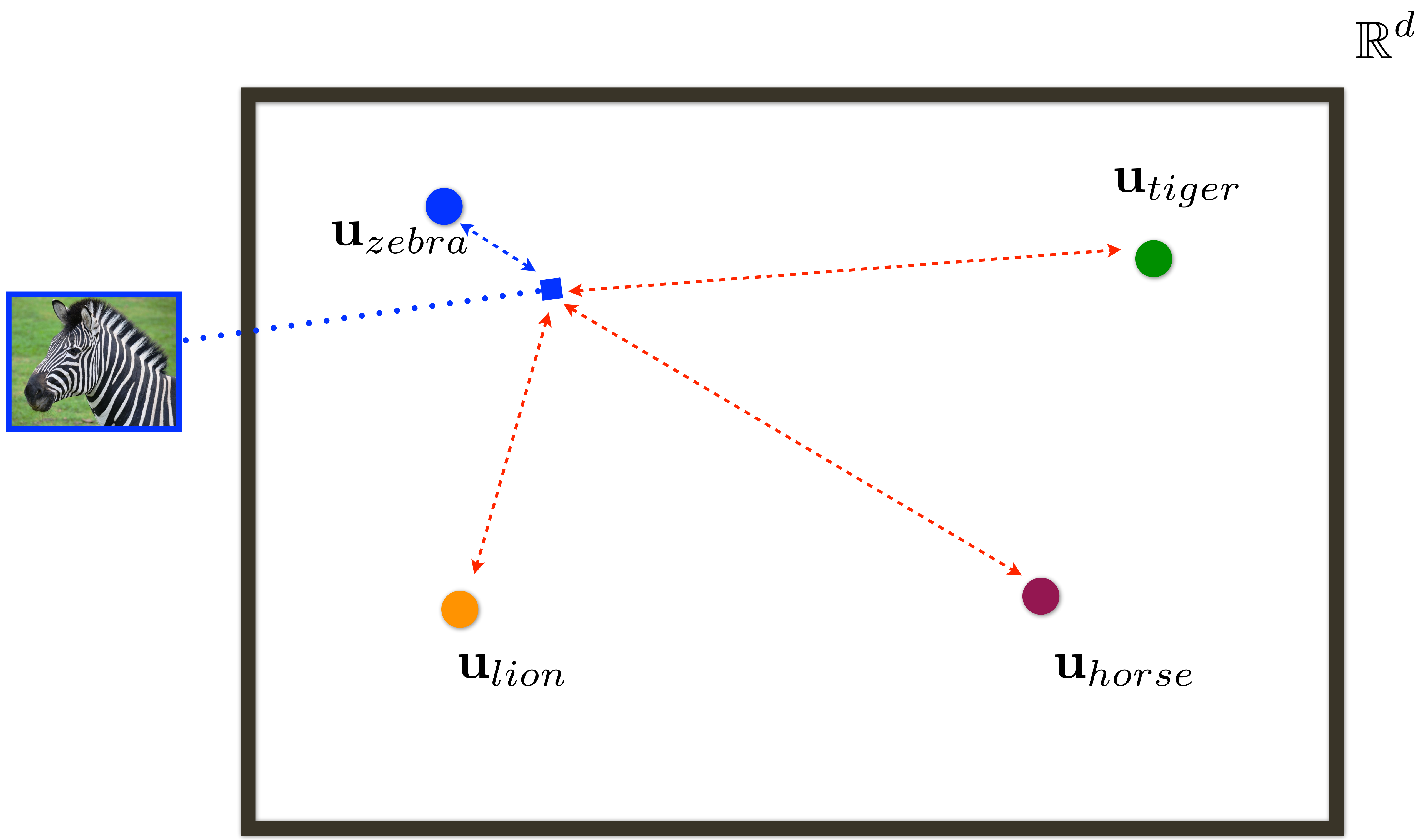
$$D(\mathbf{u}, \mathbf{u}') = ||\mathbf{u} - \mathbf{u}'||_2^2$$

Objective Function:

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

Why not minimize distance directly?

$$\mathcal{L}_C(\mathbf{W}, \mathbf{U}, I_i, y_i) = \sum [1 + D(\Psi(I_i), \mathbf{u}_{y_i}) - D(\Psi(I_i), \mathbf{u}_{y_c})]$$



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

[Weinberger, Chapelle, NIPS'09]

Discriminative Embeddings

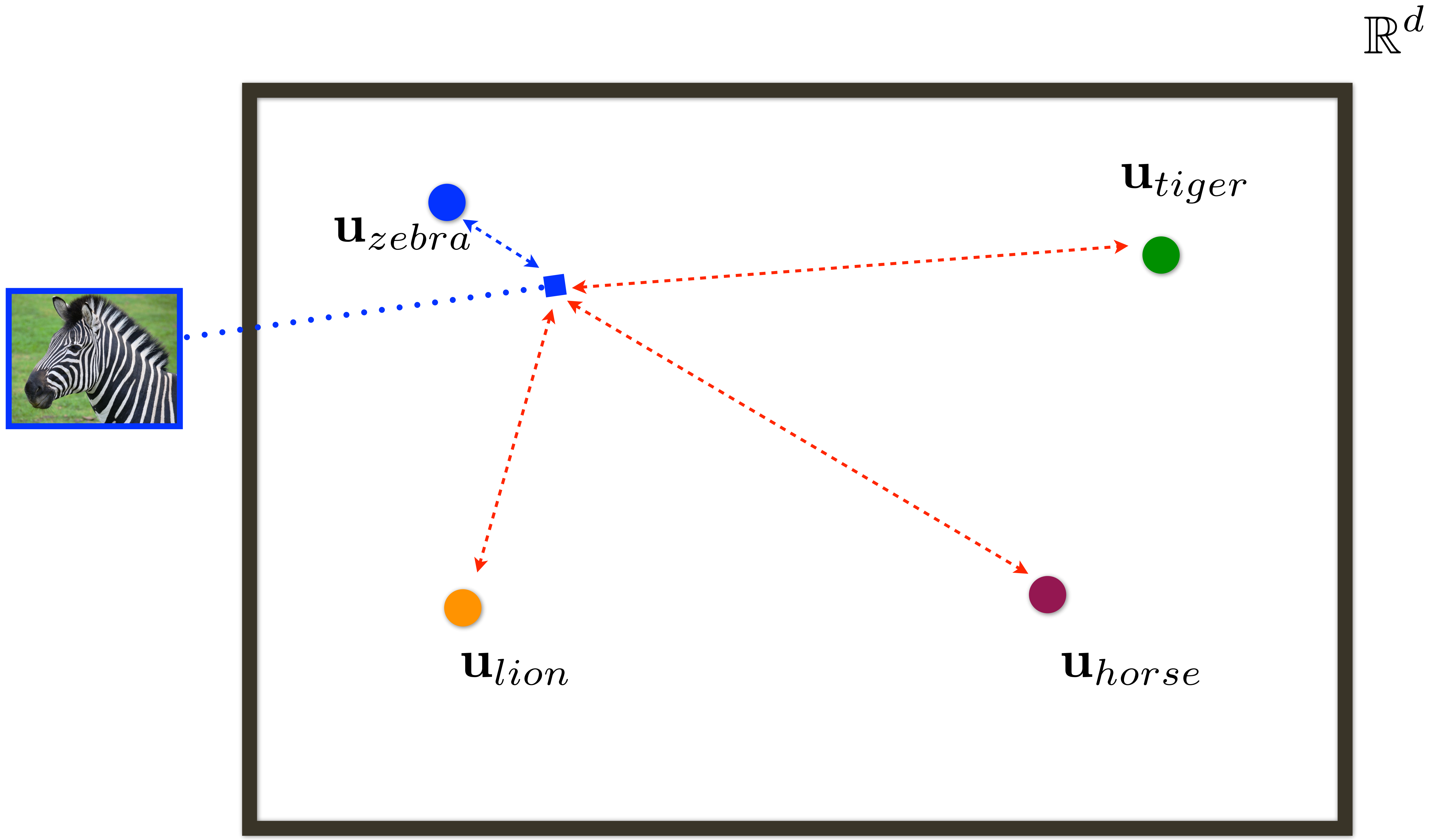
$$\mathcal{L}_C(\mathbf{W}, \mathbf{U}, I_i, y_i) = \sum max\{0, \alpha - \underline{D(\Psi(I_i), \mathbf{u}_{y_i})} + \underline{D(\Psi(I_i), \mathbf{u}_{y_c})}\}$$

Image Embedding

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

Label Embedding

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



Similarity in Embedding Space

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

Objective Function:

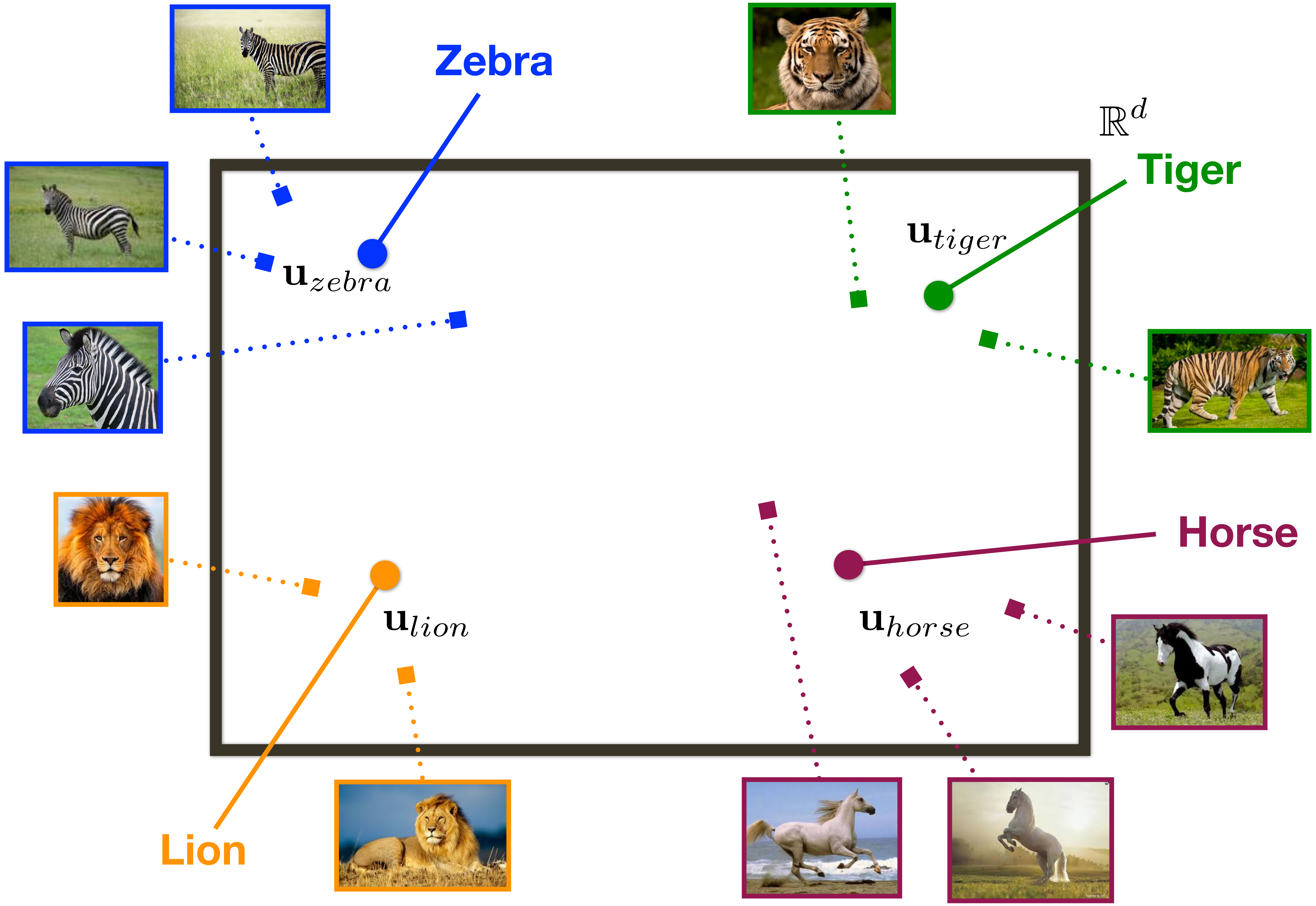
$$\min_{\mathbf{W}, \mathbf{U}} \sum_i^N \mathcal{L}_C(\mathbf{W}, \mathbf{U}, I_i, y_i) + \lambda_1 ||\mathbf{W}||_F^2 + \lambda_2 ||\mathbf{U}||_F^2$$

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

[Weinberger, Chapelle, NIPS'09]

Discriminative Embeddings

This is a very **convenient model**



Inducing semantics on the embedding space

Semantic Embeddings

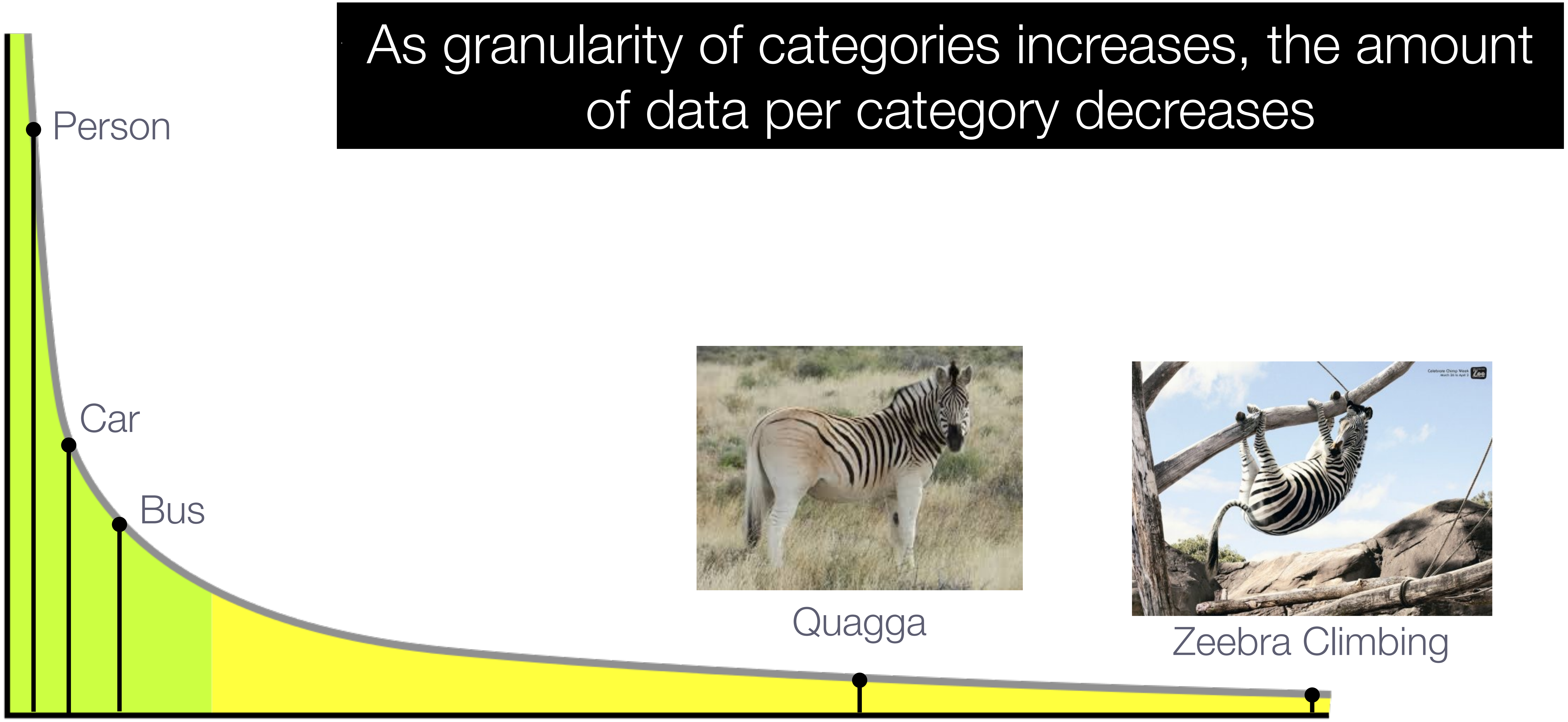
Why adding **semantics is useful**?

— Allows for transference of knowledge from classes that have a lot of data to those that have few (or no labeled instances)

— Can serve as additional regularization, so can be more efficient for learning.

Long Tail of Categories

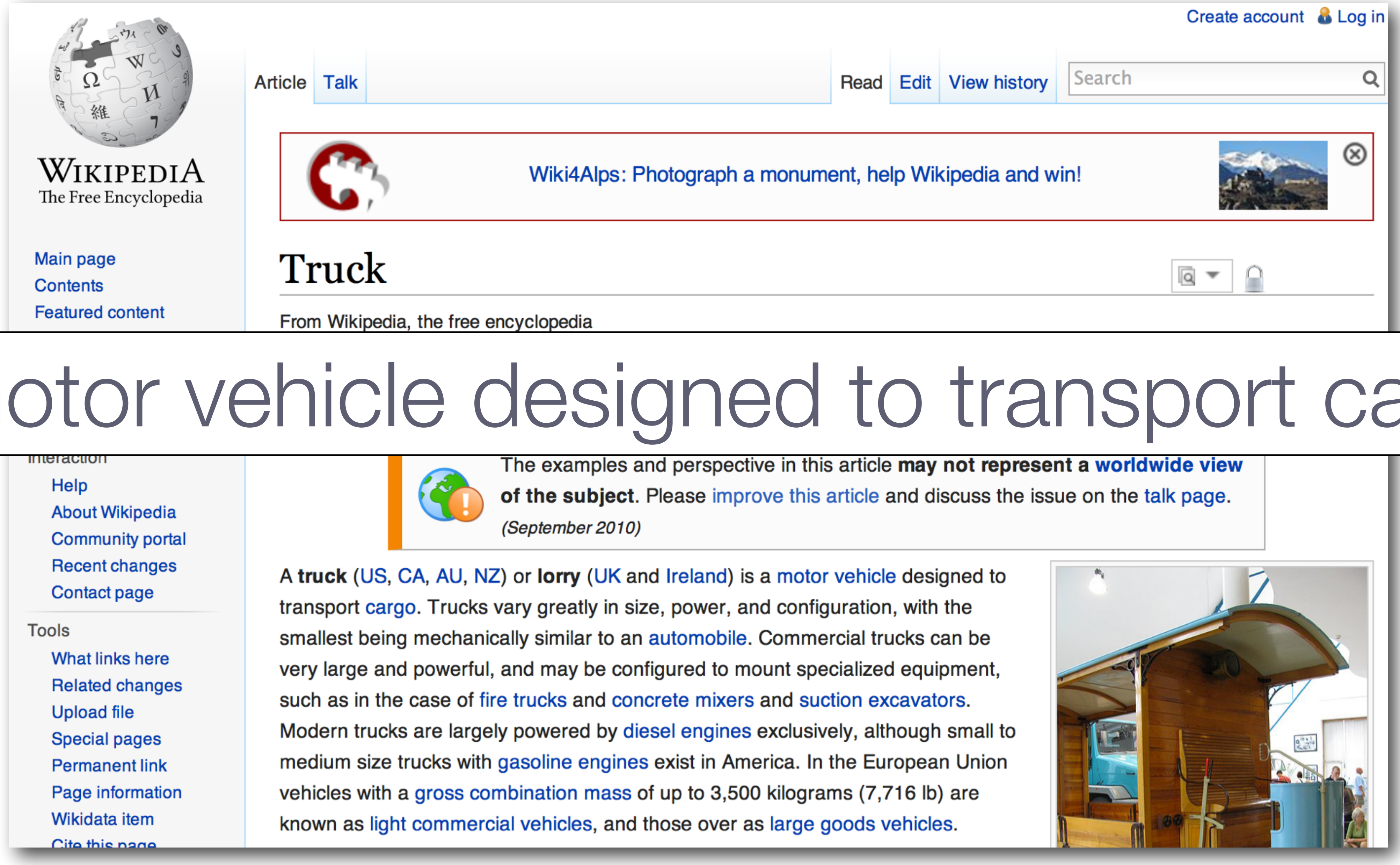
Few most frequent categories contain most of the samples, most of the categories contain few samples



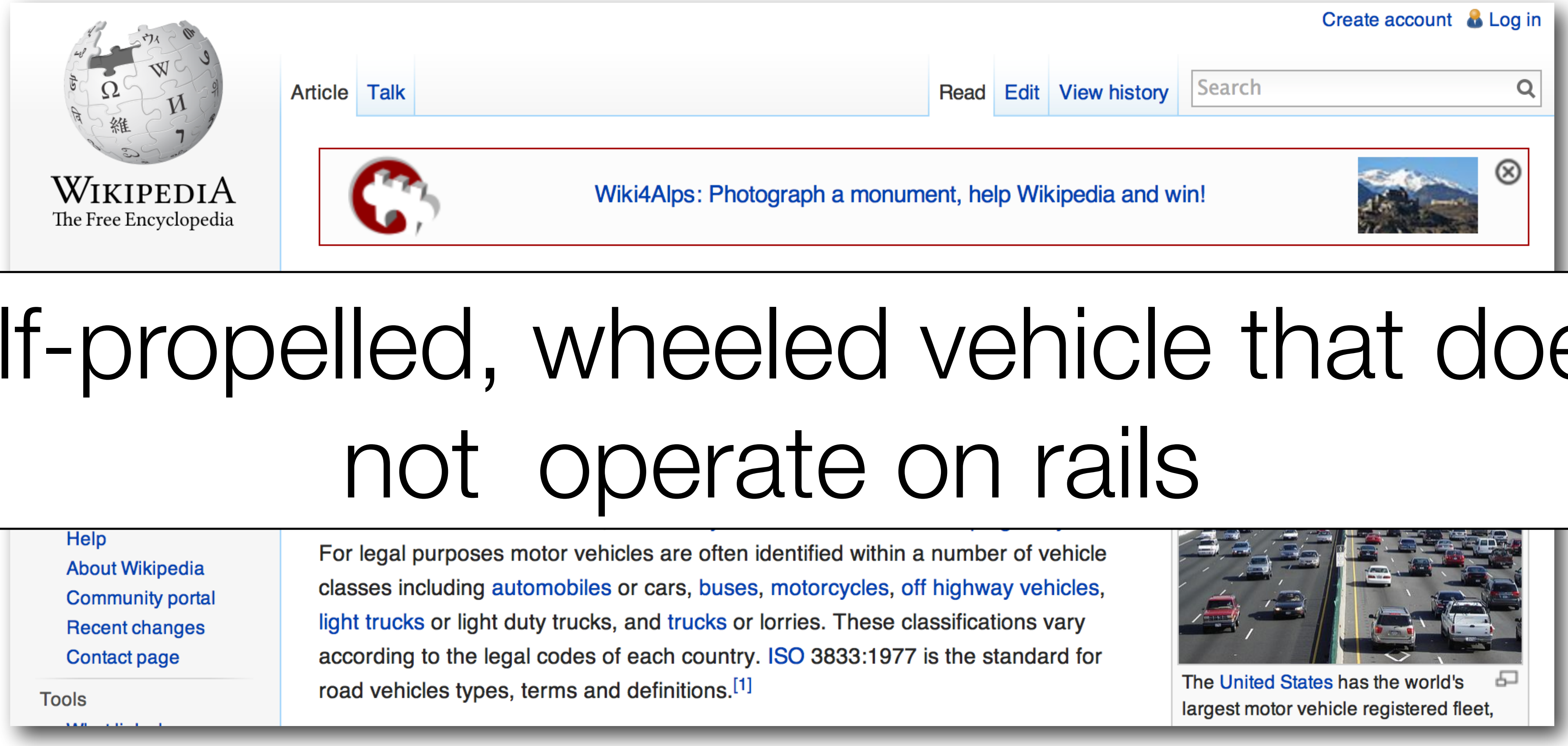
Inspiration from Human Structured Semantics

[Hwang et al., 2014]

Truck



motor vehicle designed to transport cargo



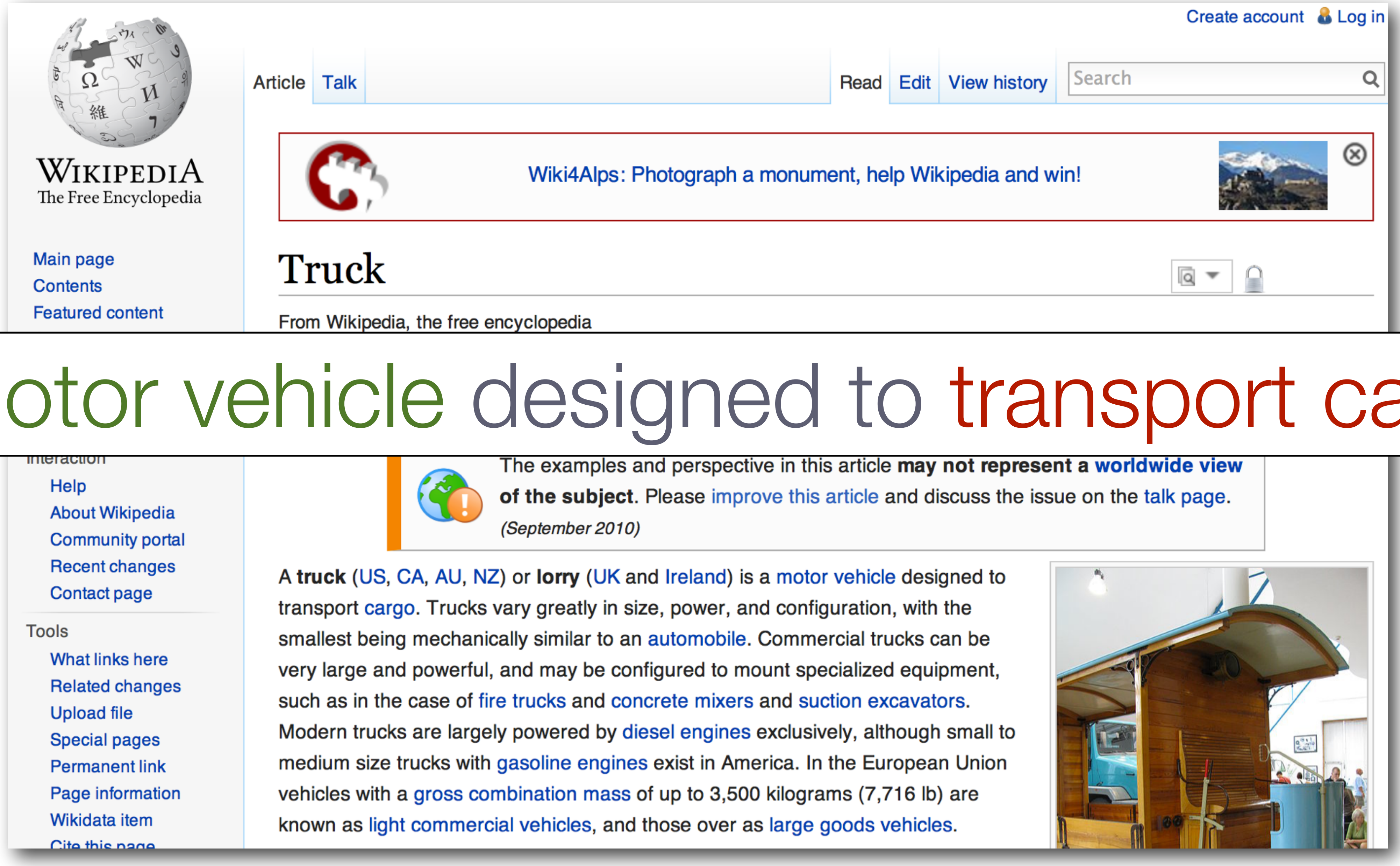
self-propelled, wheeled vehicle that does not operate on rails

Inspiration from Human Structured Semantics

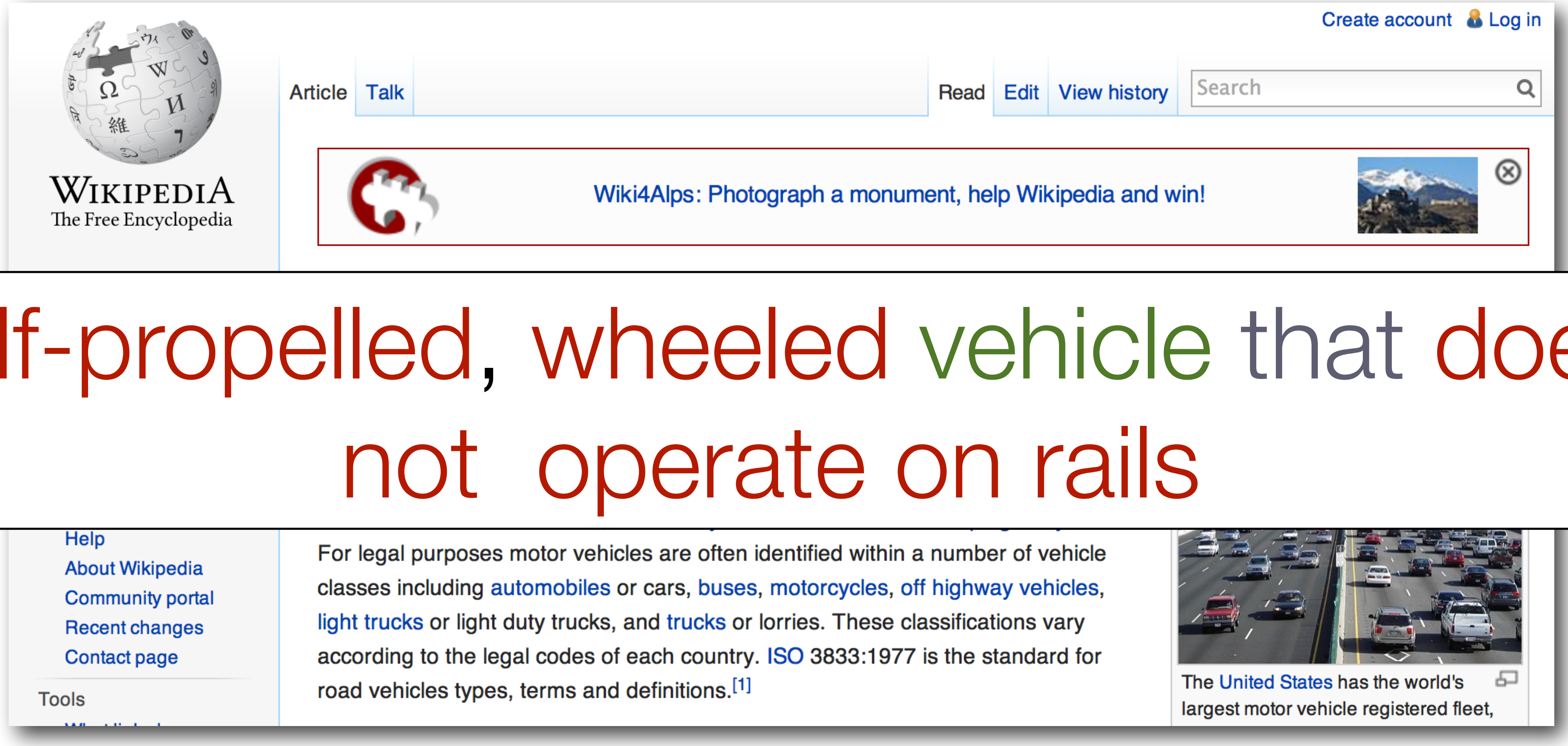
[Hwang et al., 2014]

Parent Category + Attributes

Truck



motor vehicle designed to transport cargo



self-propelled, wheeled vehicle that does not operate on rails

Unified Semantic Embedding

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

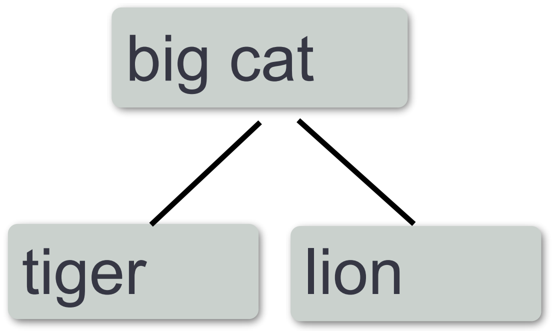
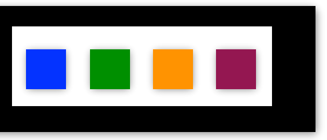
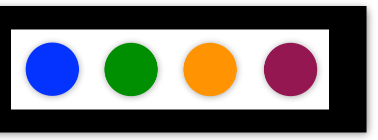


Image Embedding



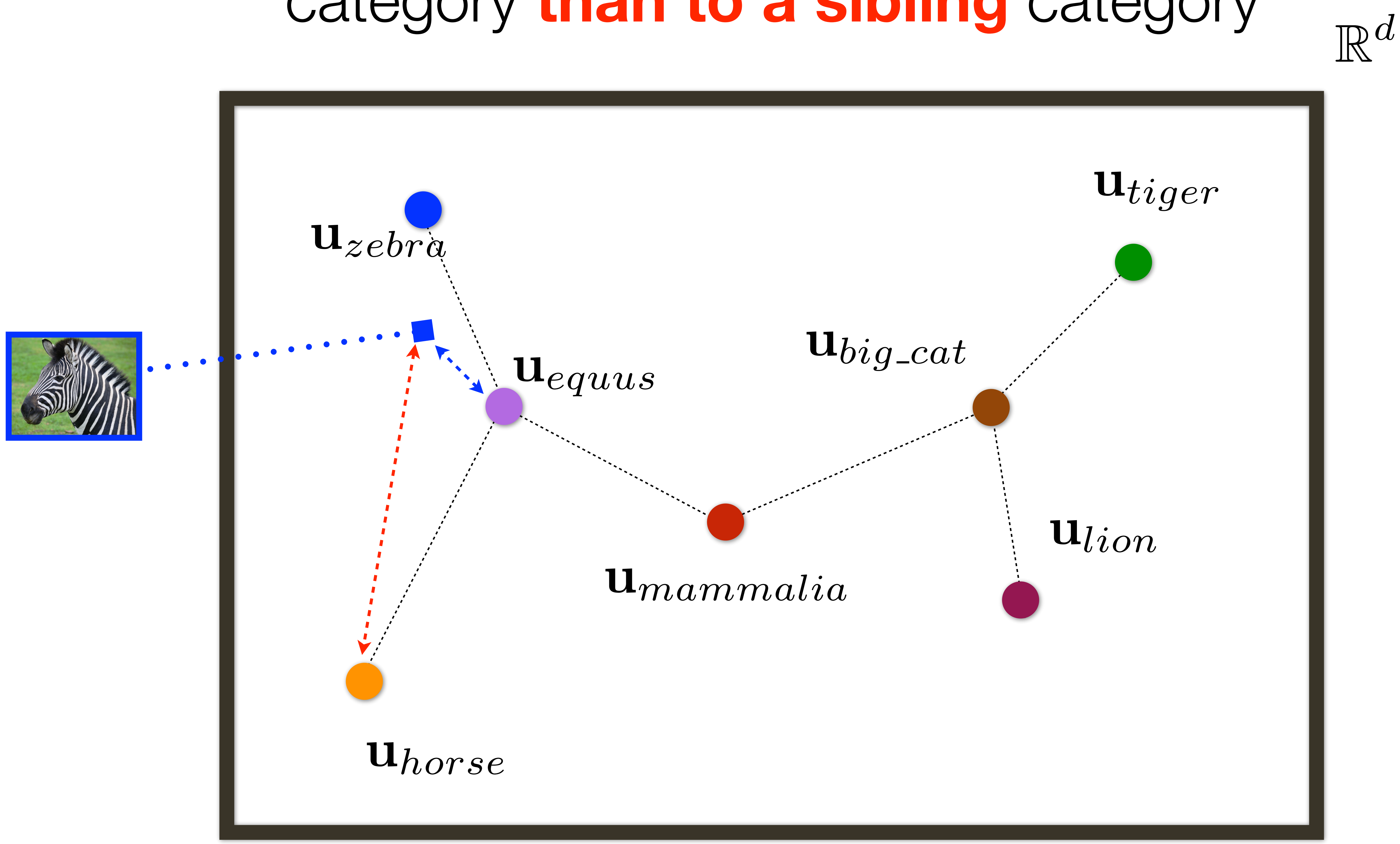
$$\Psi_I(I_i) = \mathbf{W} \cdot CNN(I_i) : \mathbb{R}^D \rightarrow \mathbb{R}^d$$

Label Embedding



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

Each sample is **closer to the parent** category **than to a sibling** category



Similarity in Embedding Space

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

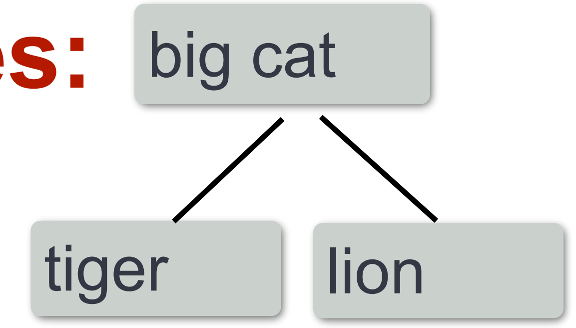
Objective Function:

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

$$||\mathbf{W}||_F^2 + \lambda||\mathbf{U}||_F^2$$

Unified Semantic Embedding

Taxonomies:



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

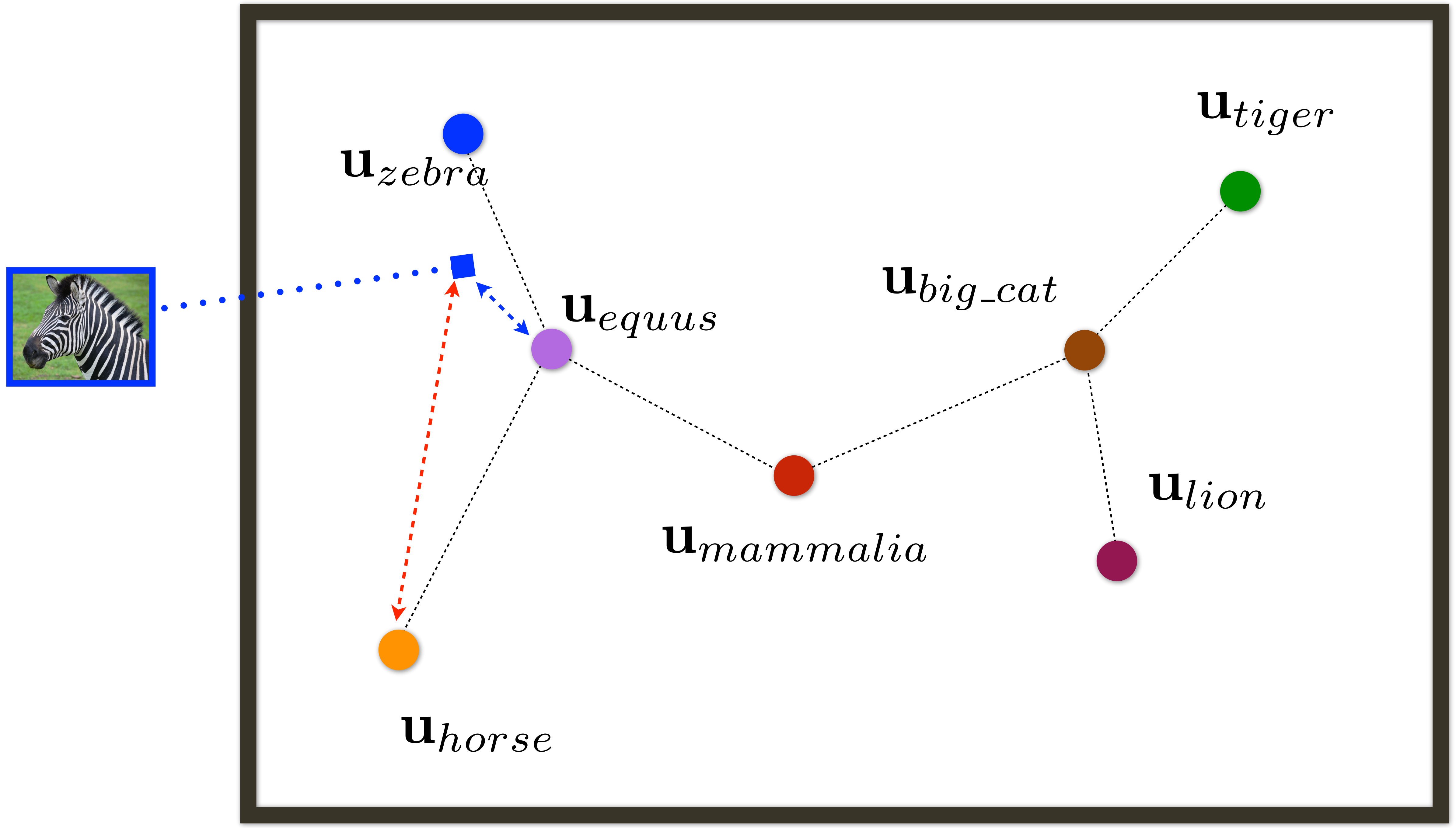
Image Embedding

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

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:

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

Unified Semantic Embedding

Attributes : has(zebra, Stripes)

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

Image Embedding

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

Label Embedding

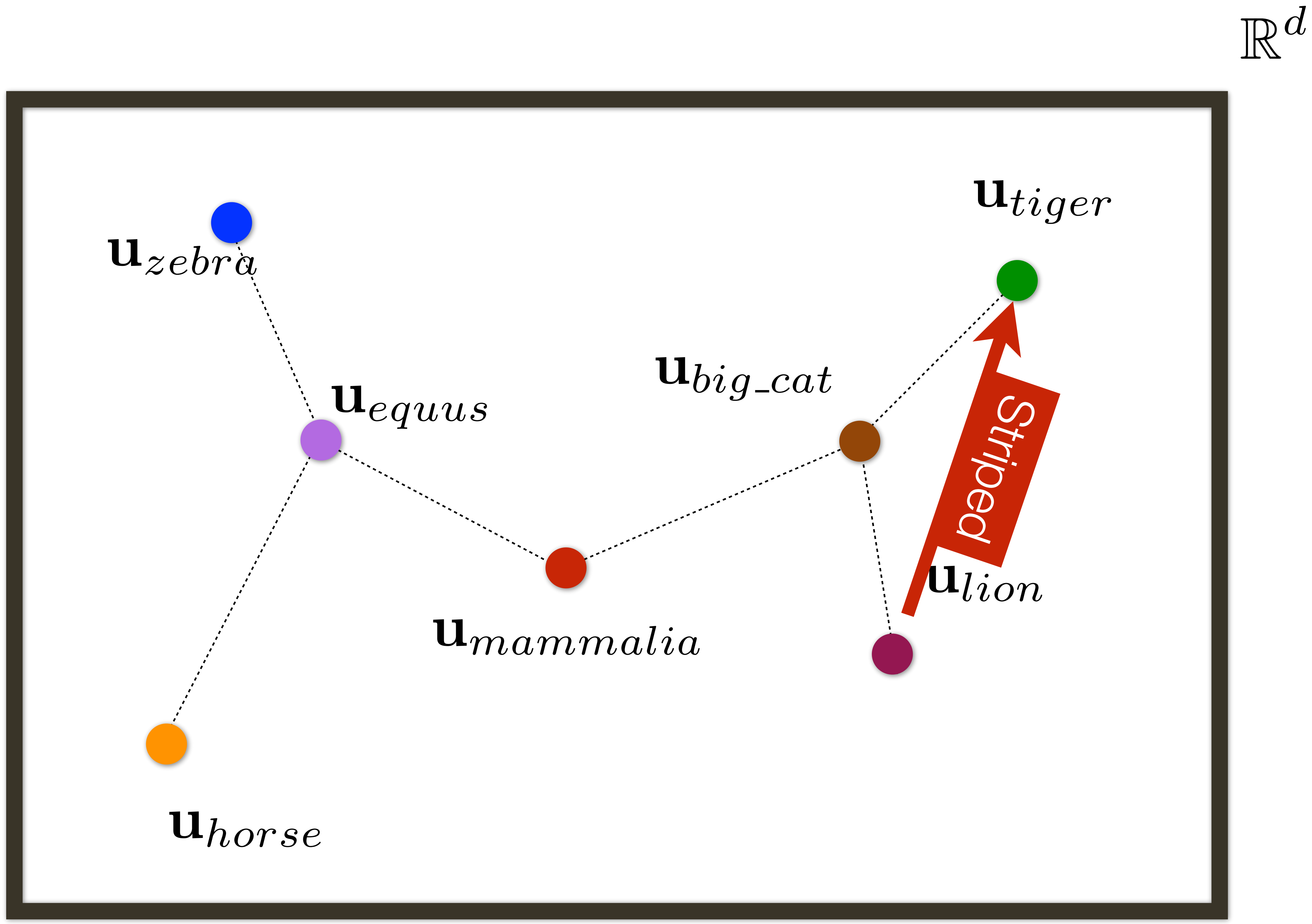
$$\Psi_L(word_i) = \mathbf{u}_i : \{1, ..., L\} \to \mathbb{R}^d$$

Attribute Embedding

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

Similarity in Embedding Space

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



Objective Function:

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

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

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

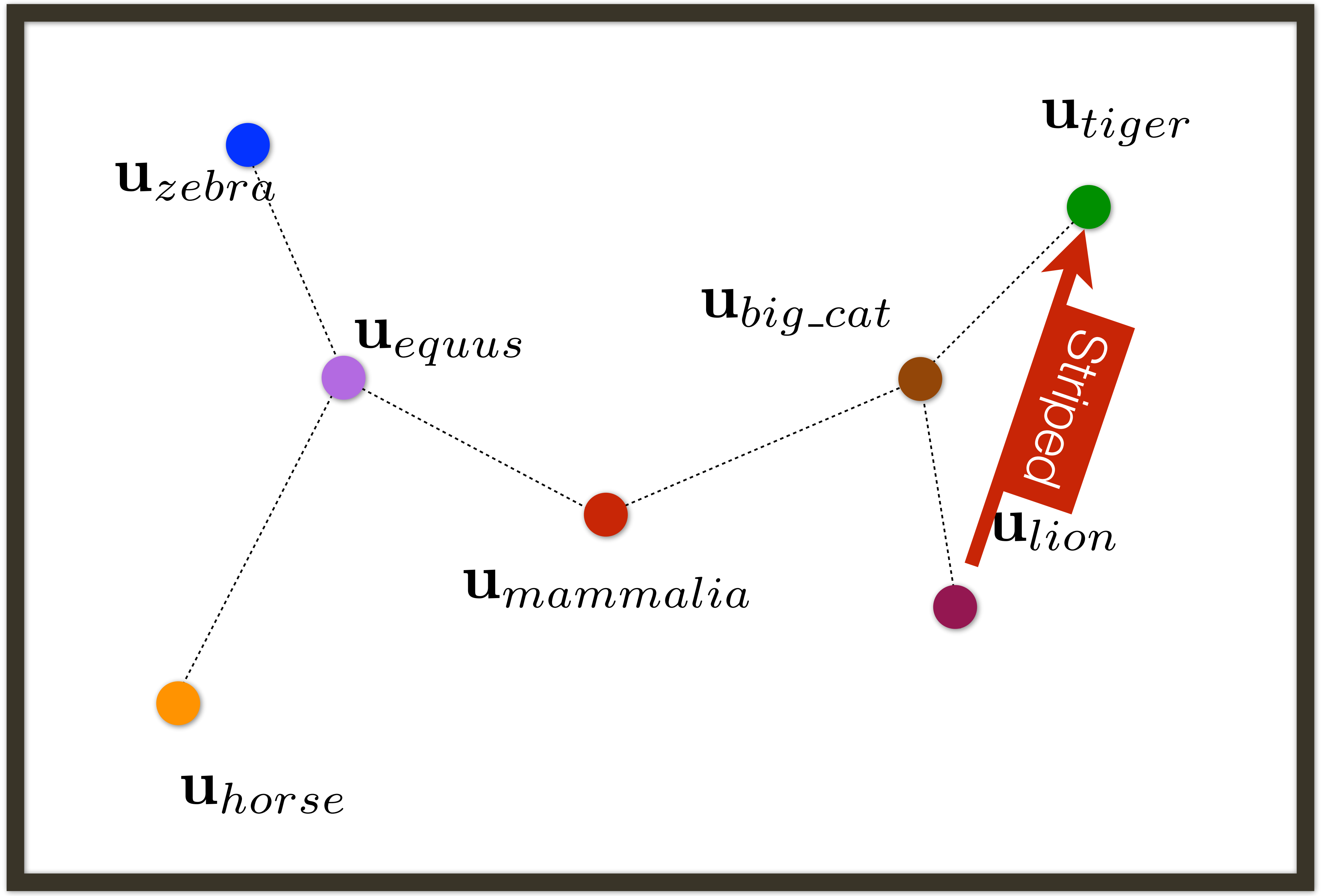
Similarity in Embedding Space

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

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

each category is a parent + sparse subset of attribute bases

$\mathbb{R}^d$



Objective Function:

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

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

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

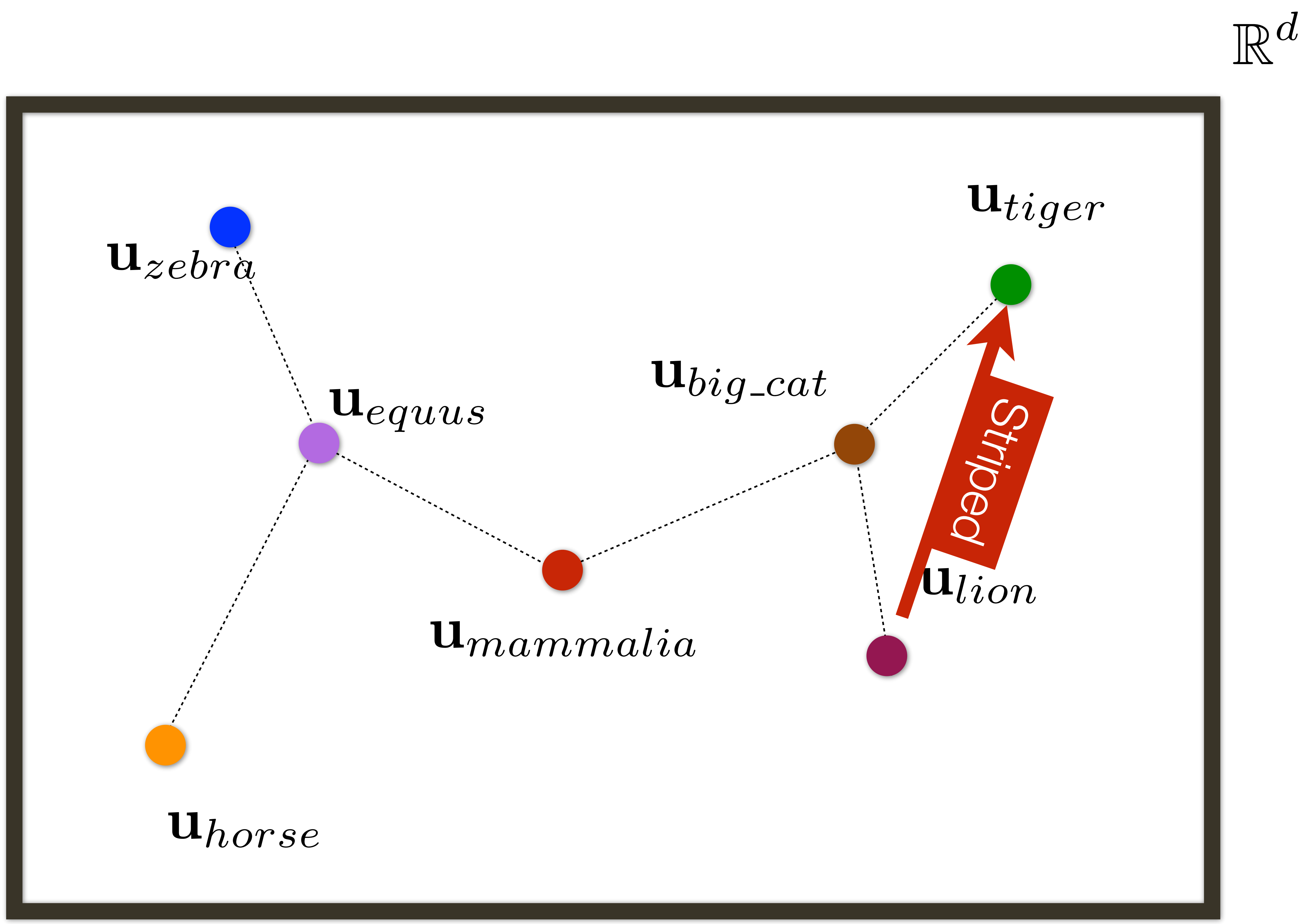
Similarity in Embedding Space

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

Objective Function:

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

Alternating optimization



Experiments: Animals with Attributes (AwA) Dataset

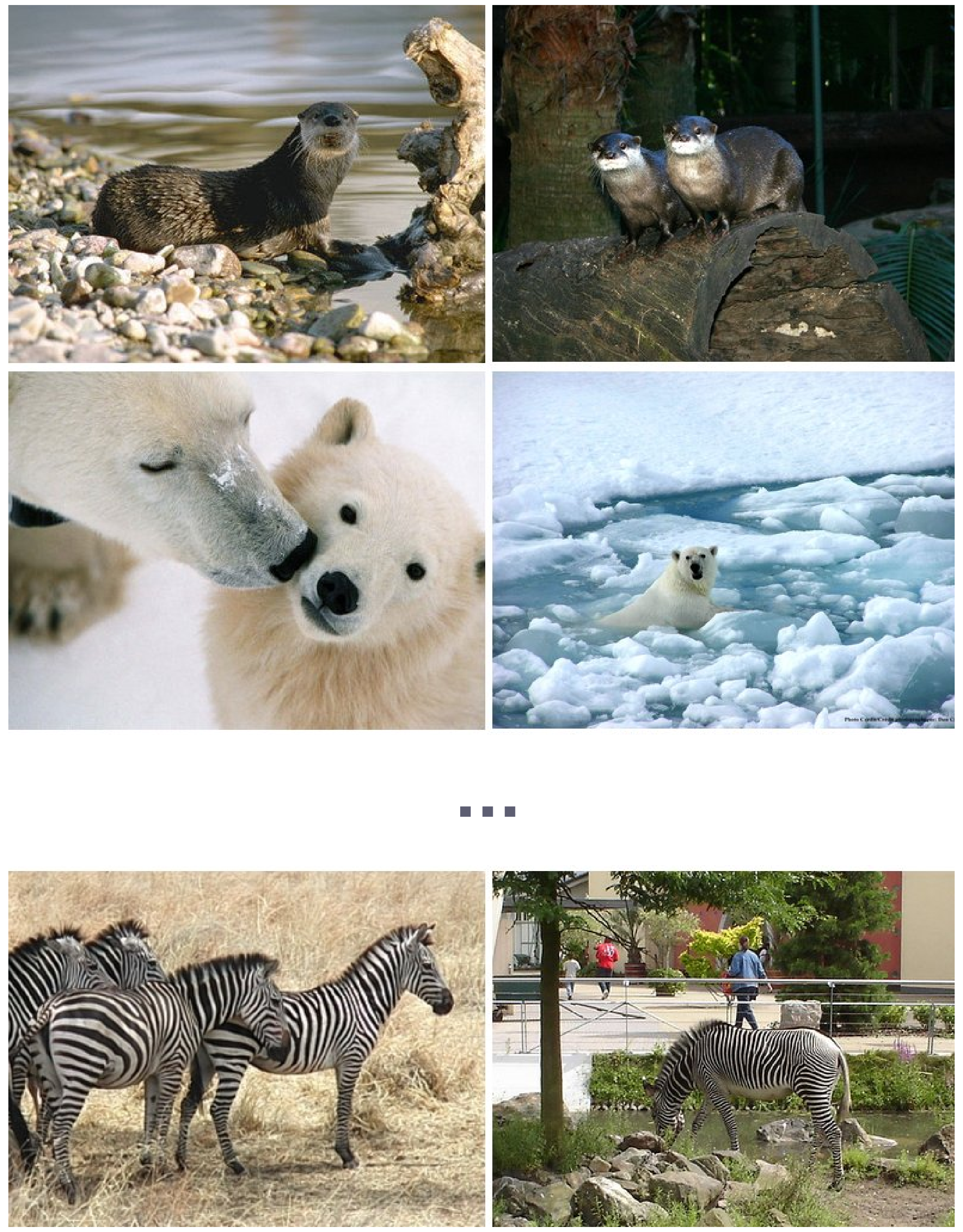
(we assume no association between classes and attributes)

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

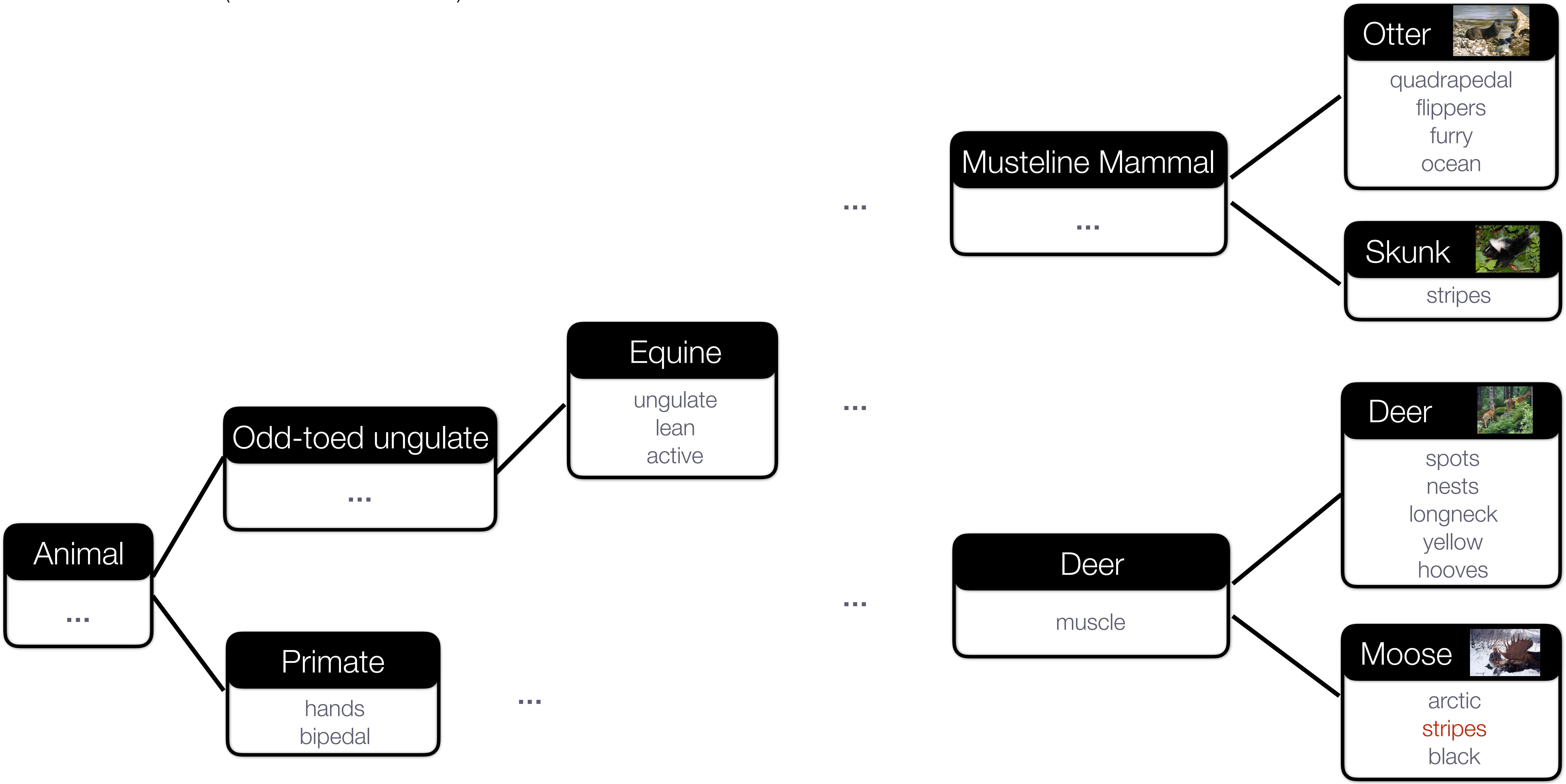
50 Animal Classes are Leaves

[Lampert, Nickisch, Harmeling, CVPR'09]

Experiments

Results with AWA (with latent attributes)

[Hwang et al., 2014]



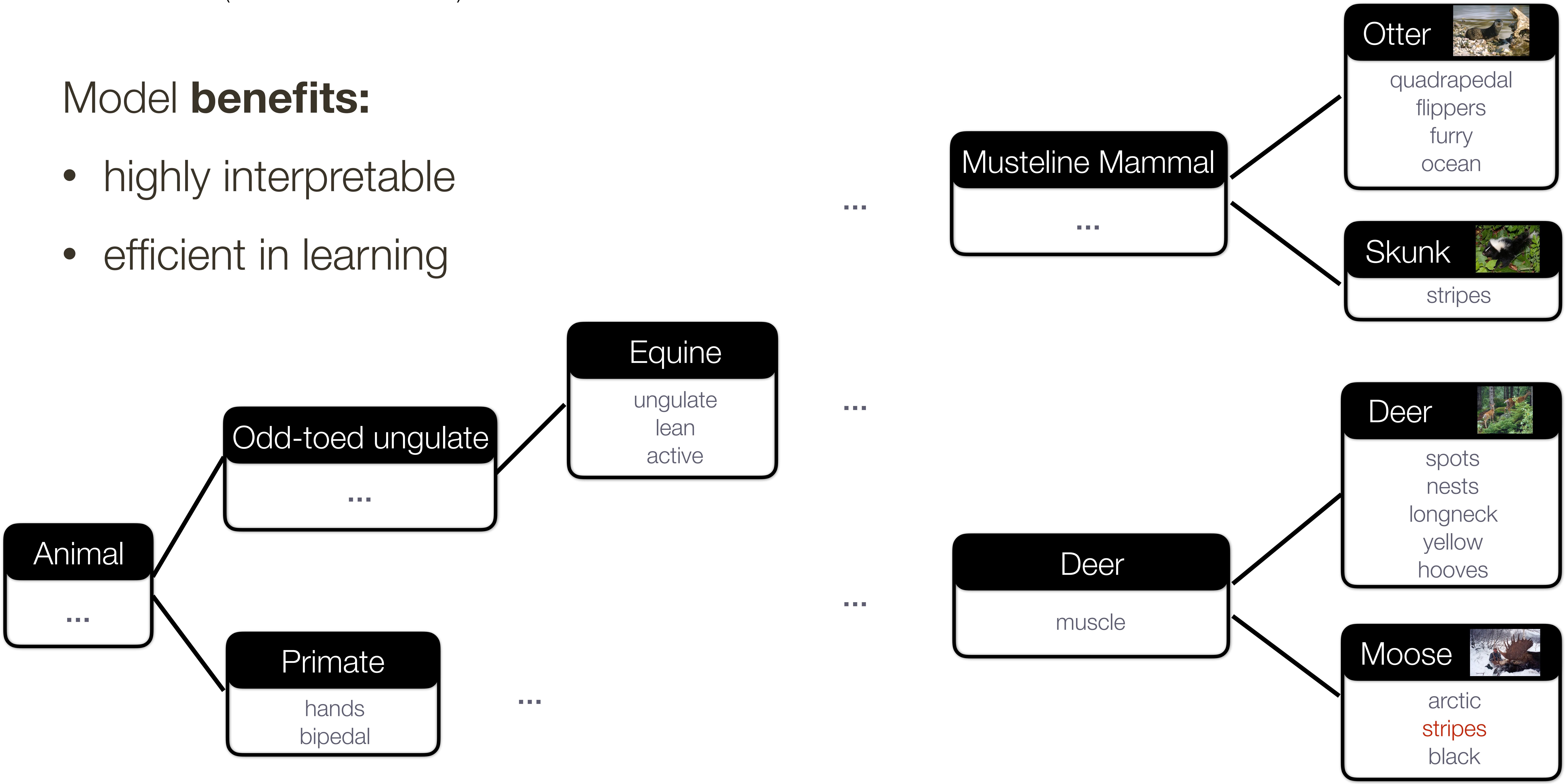
Experiments

[Hwang et al., 2014]

Results with AWA (with latent attributes)

Model **benefits:**

- highly interpretable
- efficient in learning



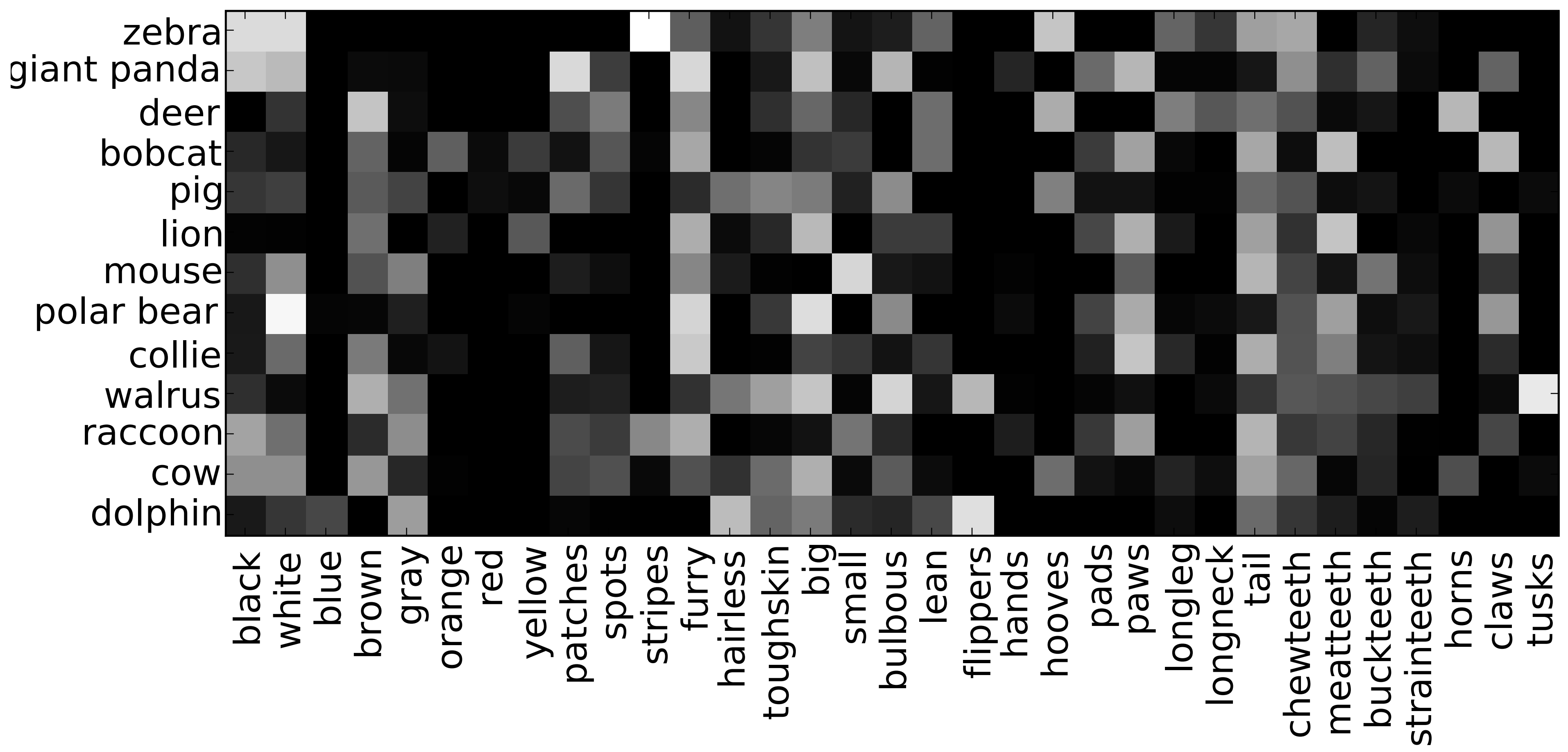
Experiments

Results with AWA (with latent attributes)

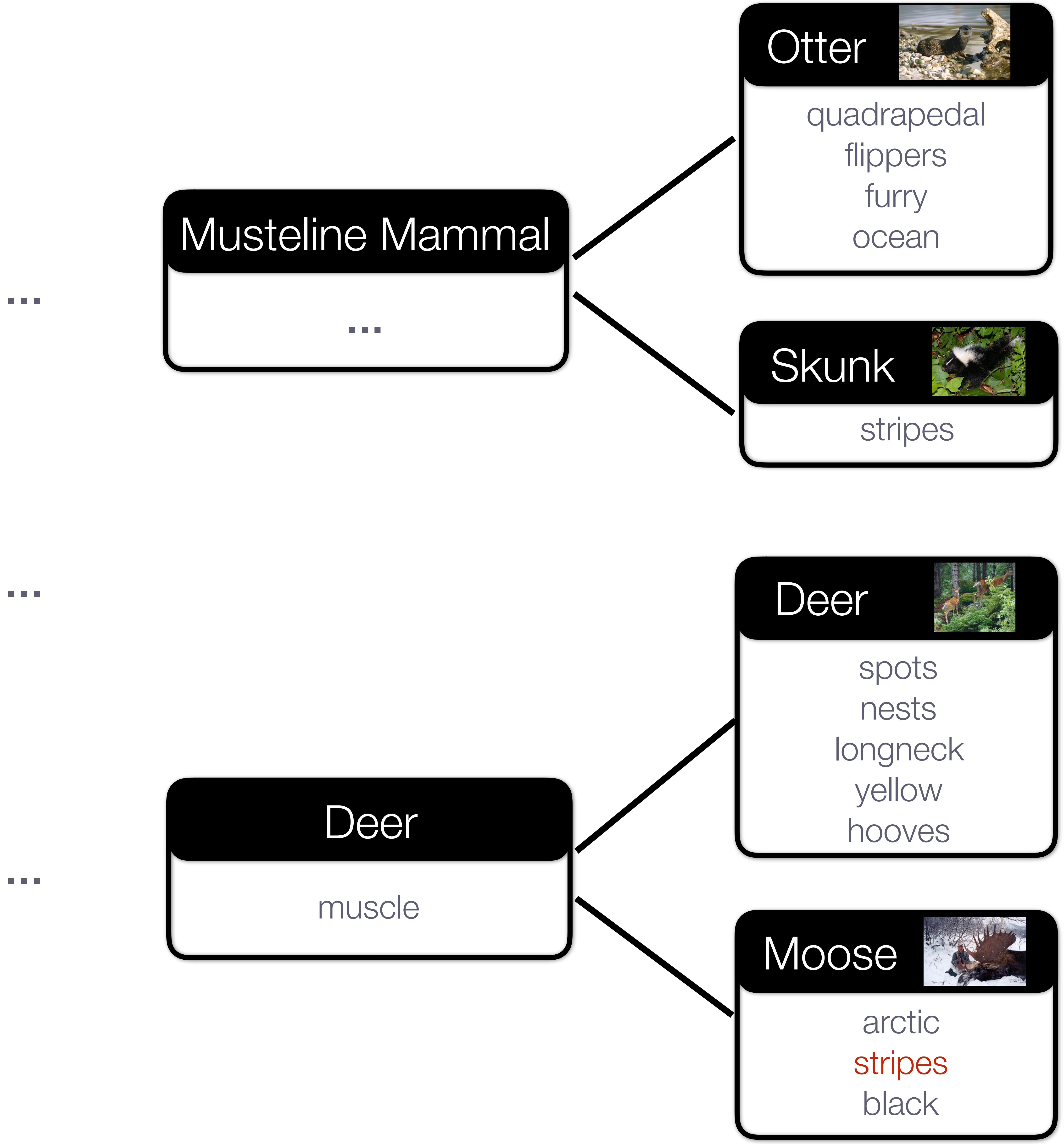
[Hwang et al., 2014]

Model **benefits:**

- highly interpretable
- efficient in learning



alternative attribute-based representations



Experiments

[Hwang et al., 2014]

Results with AWA (with latent attributes)

		Flat hit @ k (%)			Hierarchical precision @ k (%)	
	Method	1	2	5	2	5
No semantics	Ridge Regression	38.39 ± 1.48	48.61 ± 1.29	62.12 ± 1.20	38.51 ± 0.61	41.73 ± 0.54
	NCM **[1]**	43.49 ± 1.23	57.45 ± 0.91	75.48 ± 0.58	45.25 ± 0.52	50.32 ± 0.47
	LME	44.76 ± 1.77	58.08 ± 2.05	75.11 ± 1.48	44.84 ± 0.98	49.87 ± 0.39
Implicit semantics	LMTE **[2]**	38.92 ± 1.12	49.97 ± 1.16	63.35 ± 1.38	38.67 ± 0.46	41.72 ± 0.45
	ALE **[3]**	36.40 ± 1.03	50.43 ± 1.92	70.25 ± 1.97	42.52 ± 1.17	52.46 ± 0.37
	HLE **[3]**	33.56 ± 1.64	45.93 ± 2.56	64.66 ± 1.77	46.11 ± 2.65	**56.79 ± 2.05**
	AHLE **[3]**	38.01 ± 1.69	52.07 ± 1.19	71.53 ± 1.41	44.43 ± 0.66	54.39 ± 0.55
Explicit semantics	LME-MTL-S	45.03 ± 1.32	57.73 ± 1.75	74.43 ± 1.26	46.05 ± 0.89	51.08 ± 0.36
	LME-MTL-A	45.55 ± 1.71	58.60 ± 1.76	74.97 ± 1.15	44.23 ± 0.95	48.52 ± 0.29
USE	USE-No Reg.	45.93 ± 1.76	59.37 ± 1.32	74.97 ± 1.15	47.13 ± 0.62	51.04 ± 0.46
	USE-Reg.	**46.42 ± 1.33**	**59.54 ± 0.73**	**76.62 ± 1.45**	**47.39 ± 0.82**	53.35 ± 0.30

Variants of our Unified Semantic Embedding (**USE**) model:

- Ontology
- Attributes
- Parent + Sparse Attributes

[1] Mensink, Varbeek, Perronnin, Csurka Chapelle, TPAMI'13
[2] Weinberger, Chapelle, NIPS'09
[3] Akata, Perronnin, Harchaoui, Schmid, CVPR'13

Experiments

[Hwang et al., 2014]

Results with AWA (with latent attributes)

	Method	1
No semantics	Ridge Regression NCM **[1]** LME	38.93
Implicit semantics	LMTE **[2]** ALE **[3]** HLE **[3]** AHLE **[3]**	
Explicit semantics	LME-MTL-S LME-MTL-A	
USE	USE-No Reg. USE-Reg.	44.87 **+5.9%** **49.87** **+5.0%**

Variants of our Unified Semantic Embedding (**USE**) model:

Ontology
Attributes
Parent + Sparse Attributes

with **2 samples/category**

[1] Mensink, Varbeek, Perronnin, Csurka Chapelle, TPAMI'13

[2] Weinberger, Chapelle, NIPS'09

[3] Akata, Perronnin, Harchaoui, Schmid, CVPR'13

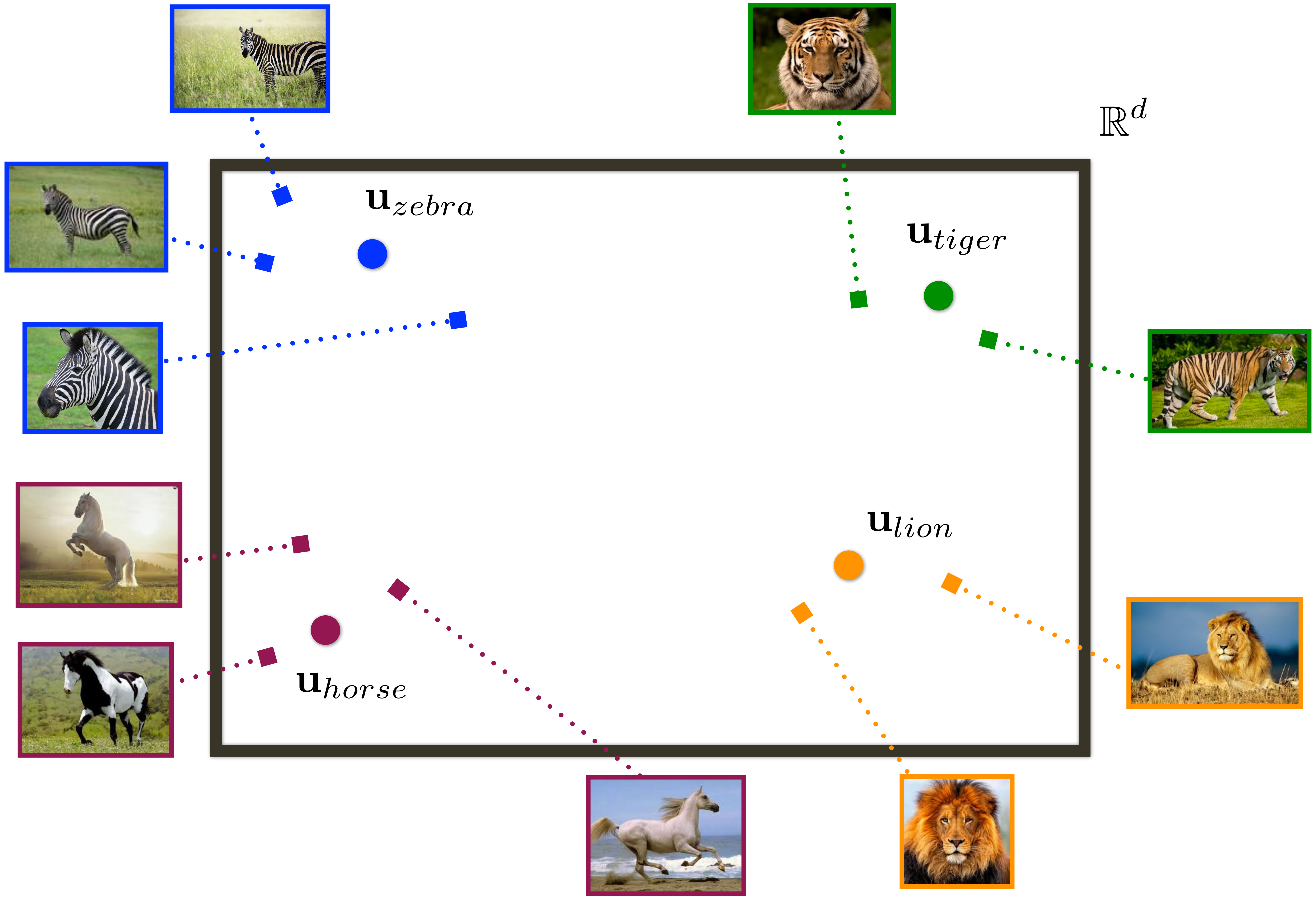
Semantic Embeddings

Image Embedding

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

Label Embedding

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

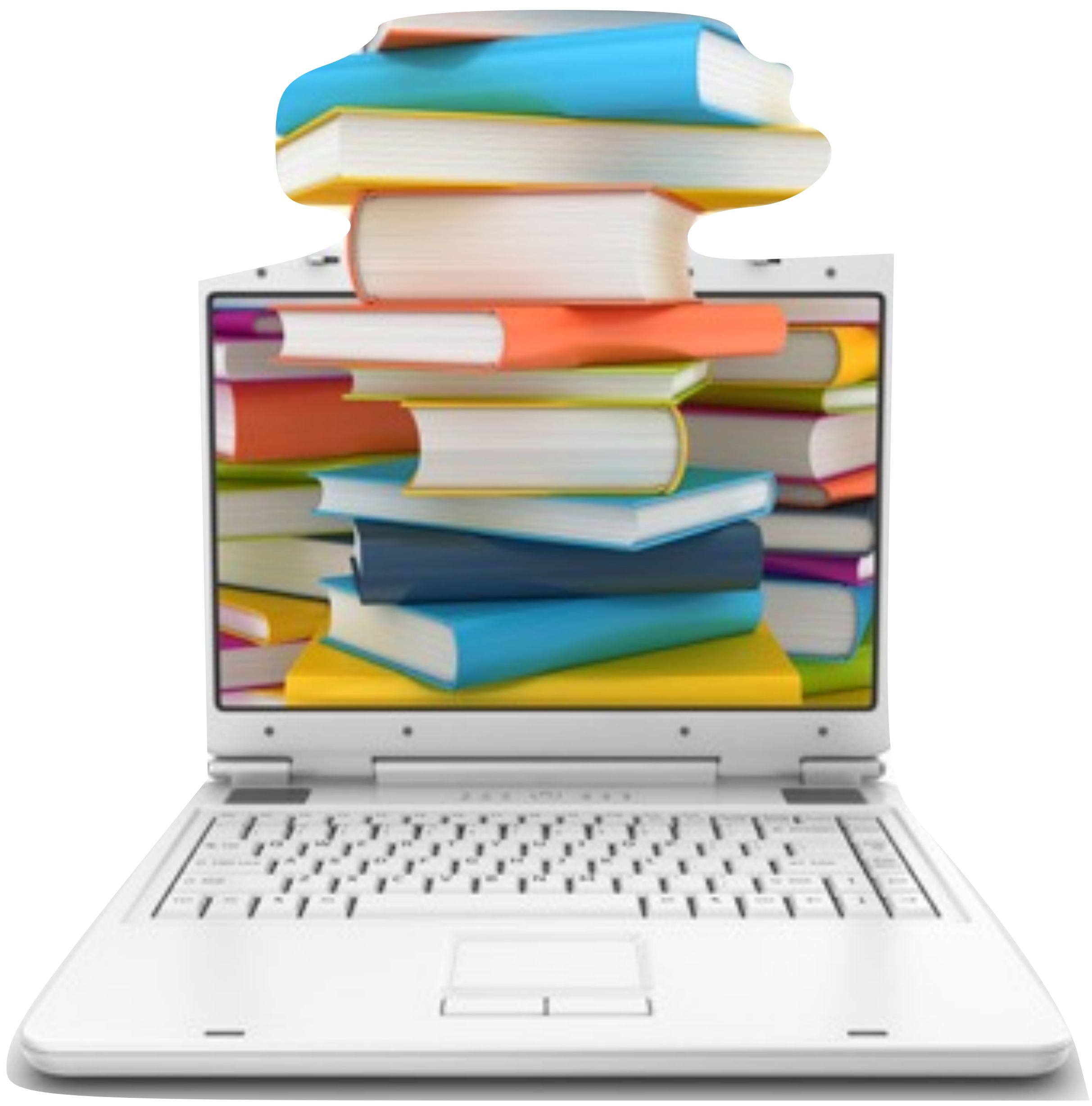


word2vec: Unsupervised Word Embedding

Distributional Semantics Hypothesis: words that are used and occur in the same context tend to have similar meaning

Label Embedding

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



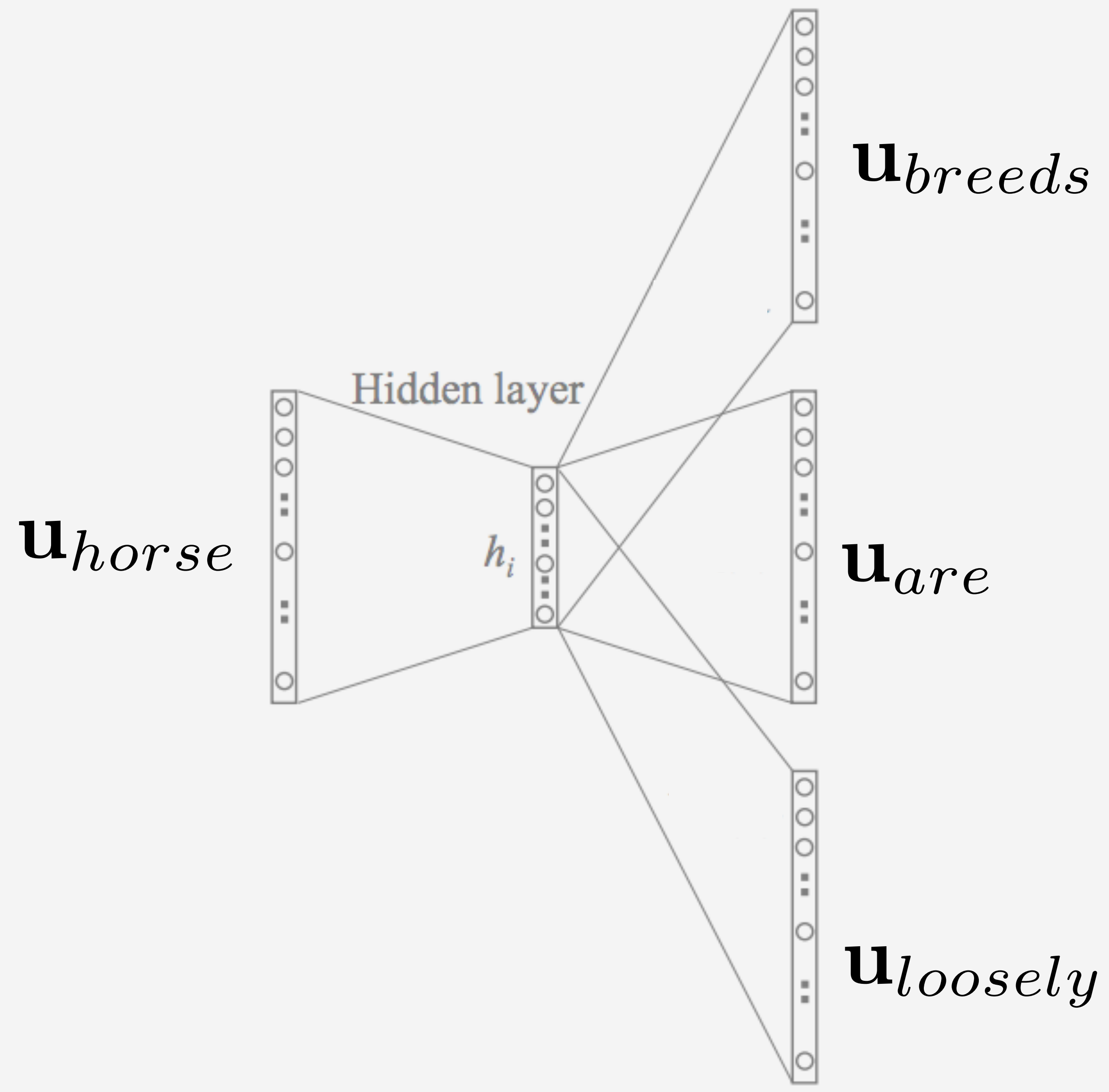
word2vec: Unsupervised Word Embedding

Distributional Semantics Hypothesis: words that are used and occur in the same context tend to have similar meaning

Label Embedding

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

e.g., Horse breeds are loosely divided into three categories

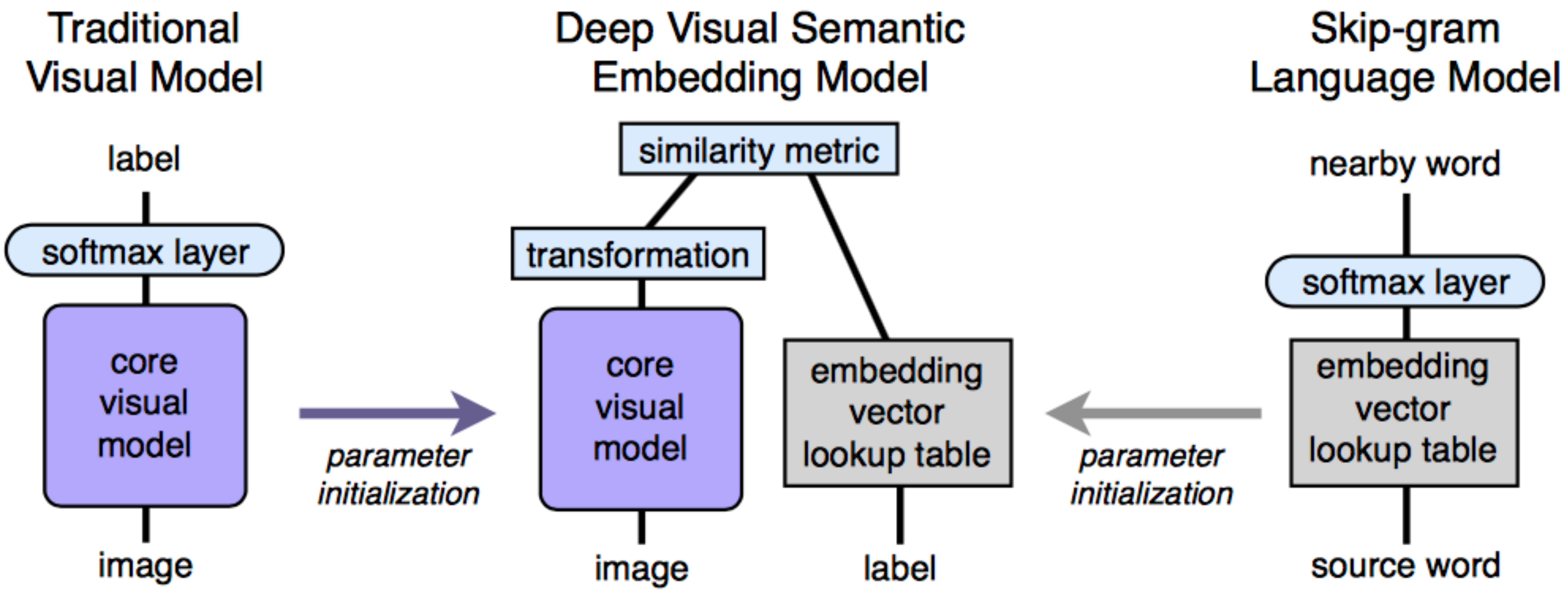


Skip-gram Model: unsupervised semantic representation for words

[Mikolov, Sutskever, Chen, Corrado, Dean, NIPS'13]

DeViSE: A Deep Visual-Semantic Embedding Model

[Frome et al., 2013]



$$loss(image, label) = \sum_{j \neq label} \max[0, margin - \vec{t}_{label} M \vec{v}(image) + \vec{t}_j M \vec{v}(image)]$$

DeViSE: A Deep Visual-Semantic Embedding Model

[Frome et al., 2013]

Supervised Results

Model type	dim	Flat hit@k (%)				Hierarchical precision@k			
		1	2	5	10	2	5	10	20
Softmax baseline	N/A	**55.6**	**67.4**	**78.5**	**85.0**	0.452	0.342	0.313	0.319
DeViSE	500	53.2	65.2	76.7	83.3	0.447	**0.352**	**0.331**	**0.341**
	1000	54.9	66.9	78.4	**85.0**	**0.454**	0.351	0.325	0.331
Random embeddings	500	52.4	63.9	74.8	80.6	0.428	0.315	0.271	0.248
	1000	50.5	62.2	74.2	81.5	0.418	0.318	0.290	0.292
Chance	N/A	0.1	0.2	0.5	1.0	0.007	0.013	0.022	0.042

Zero-shot Results

Model	200 labels	1000 labels
DeViSE	31.8%	9.0%
Mensink et al. 2012 [12]	35.7%	1.9%
Rohrbach et al. 2011 [17]	34.8%	-

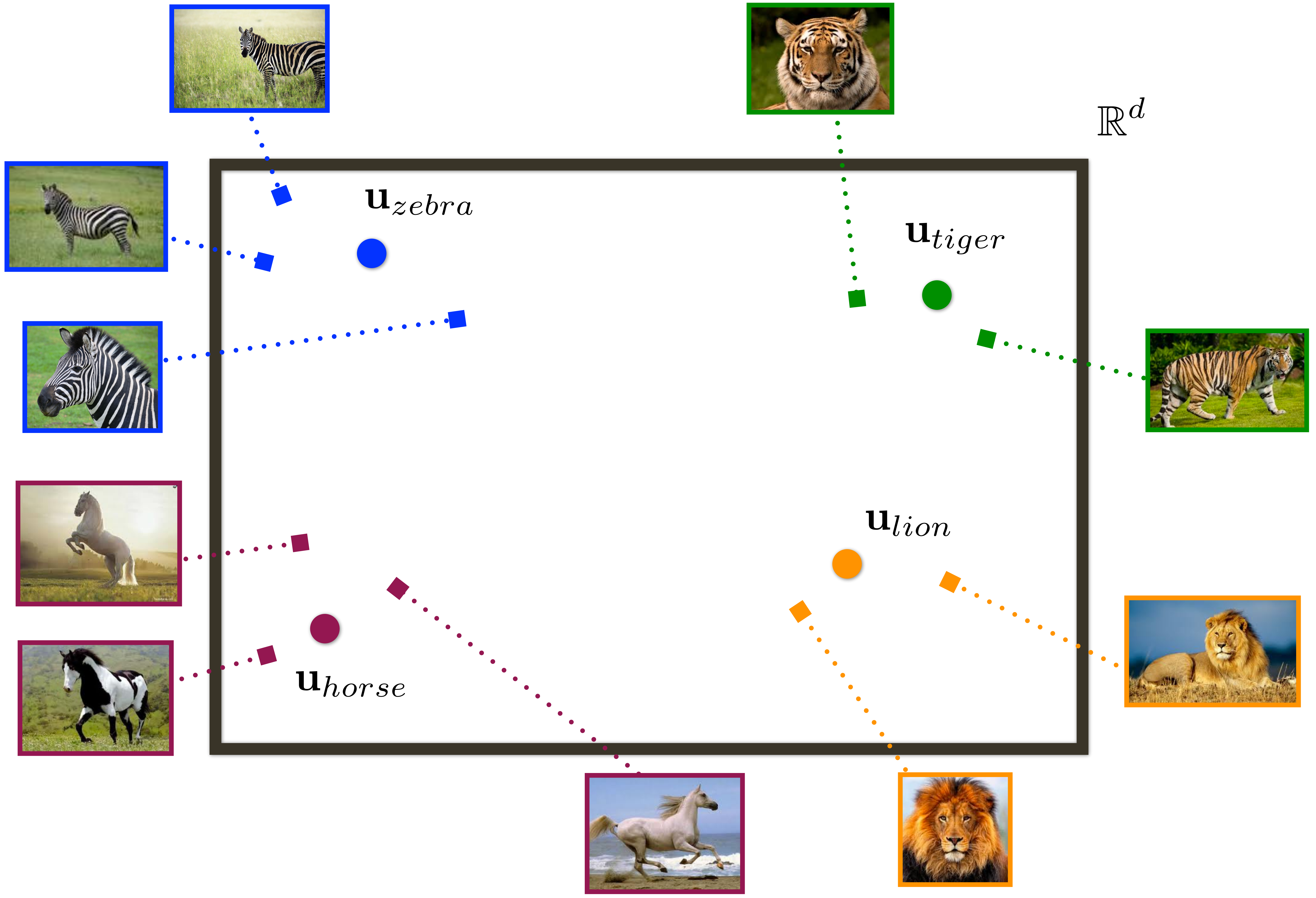
Semantic Embeddings

Image Embedding

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

Label Embedding

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



word2vec: Unsupervised Word Embedding

[Fu et al., 2016]

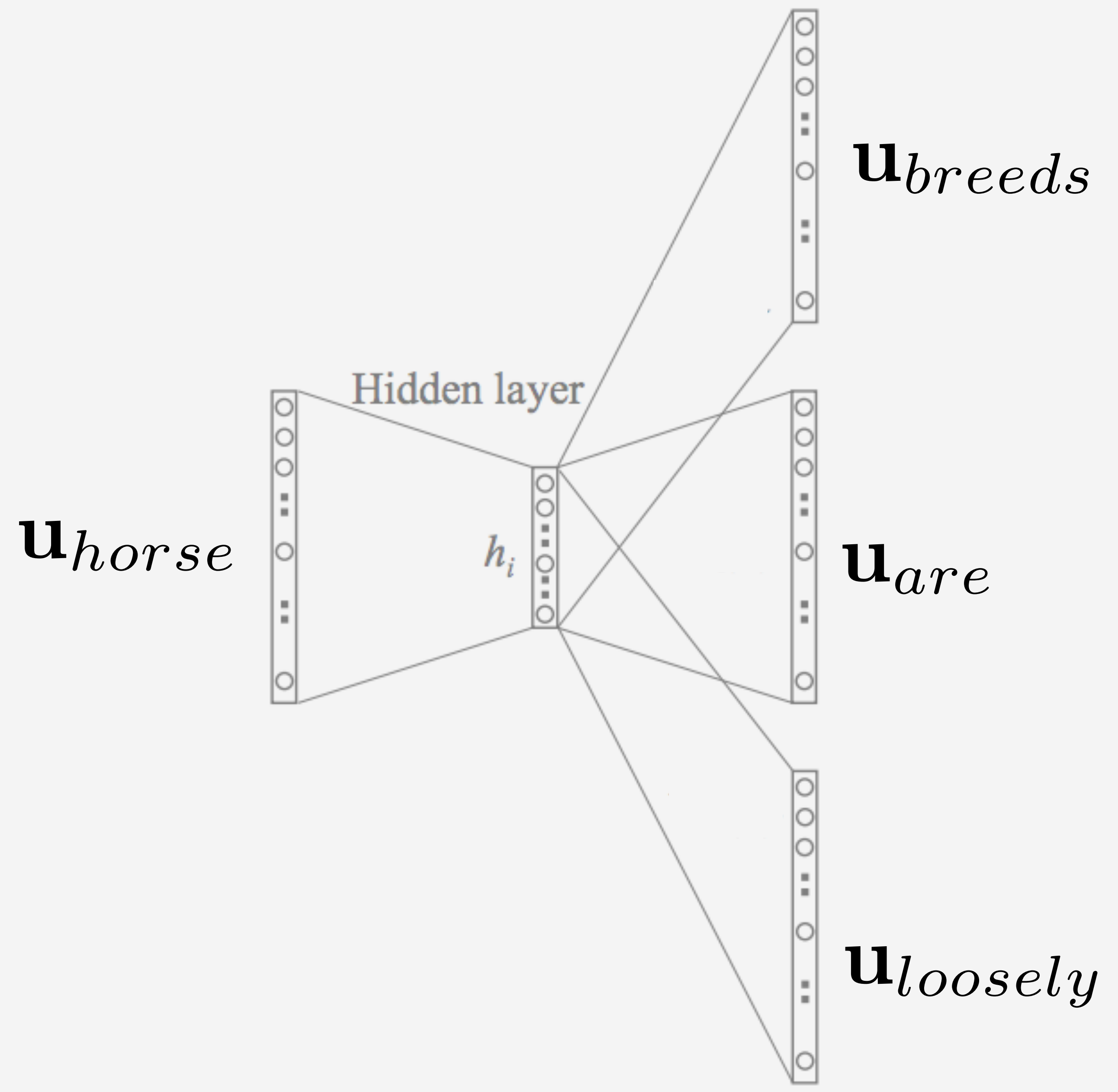
Distributional Semantics Hypothesis: words that are used and occur in the same context tend to have similar meaning

Label Embedding

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

$L = 310,000$

e.g., Horse breeds are loosely divided into three categories



Skip-gram Model: unsupervised semantic representation for words

(trained from 7 billion word linguistic corpus)

Semi-supervised **Vocabulary Informed** Learning

[Fu et al., 2016]

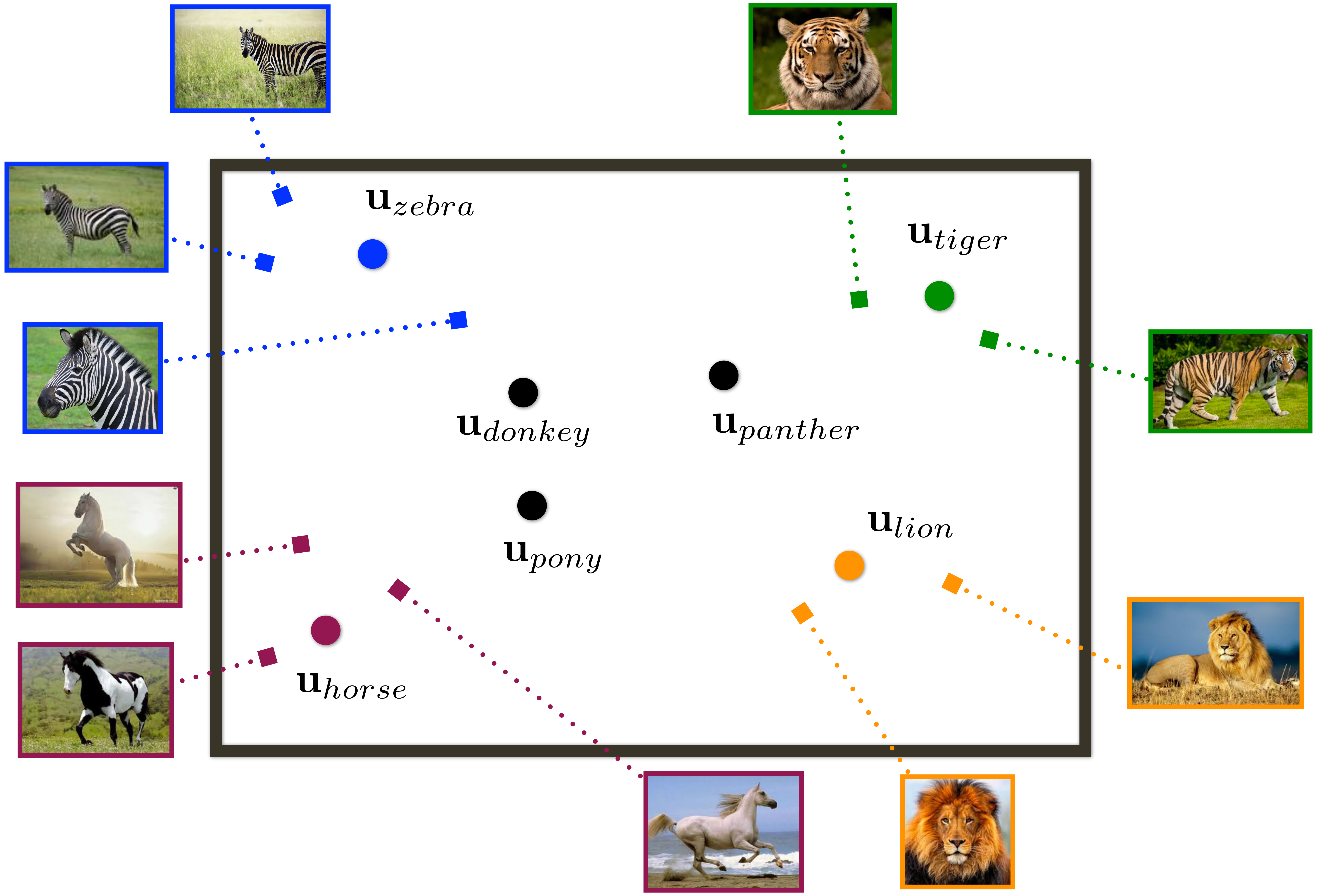
Image Embedding

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

Label Embedding

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

$L = 310,000$



Semi-supervised **Vocabulary Informed** Learning

[Fu et al., 2016]

Image Embedding

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

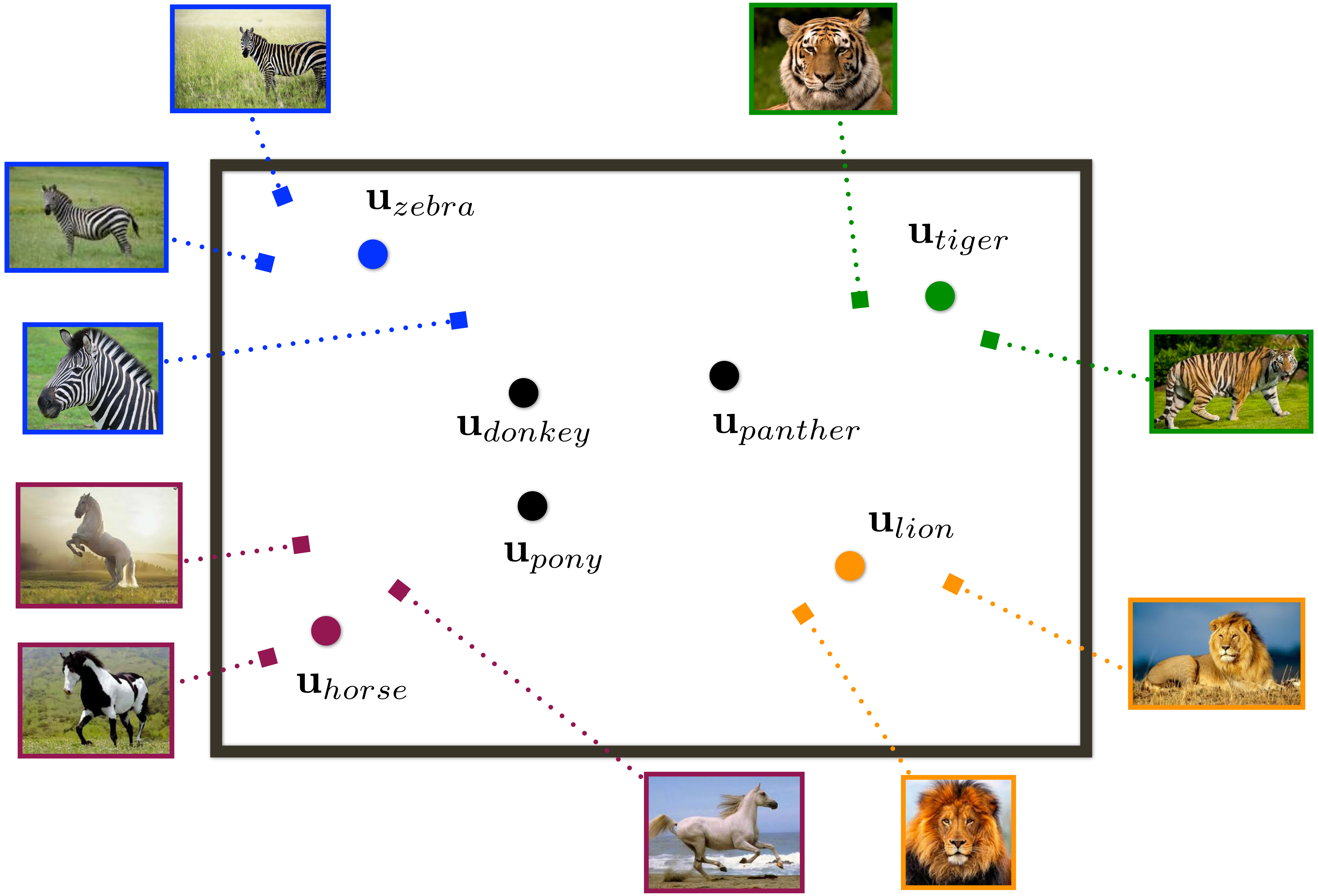
Label Embedding

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

$L = 310,000$

Similarity in Embedding Space

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



Semi-supervised **Vocabulary Informed** Learning

[Fu et al., 2016]

Image Embedding

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

Label Embedding

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

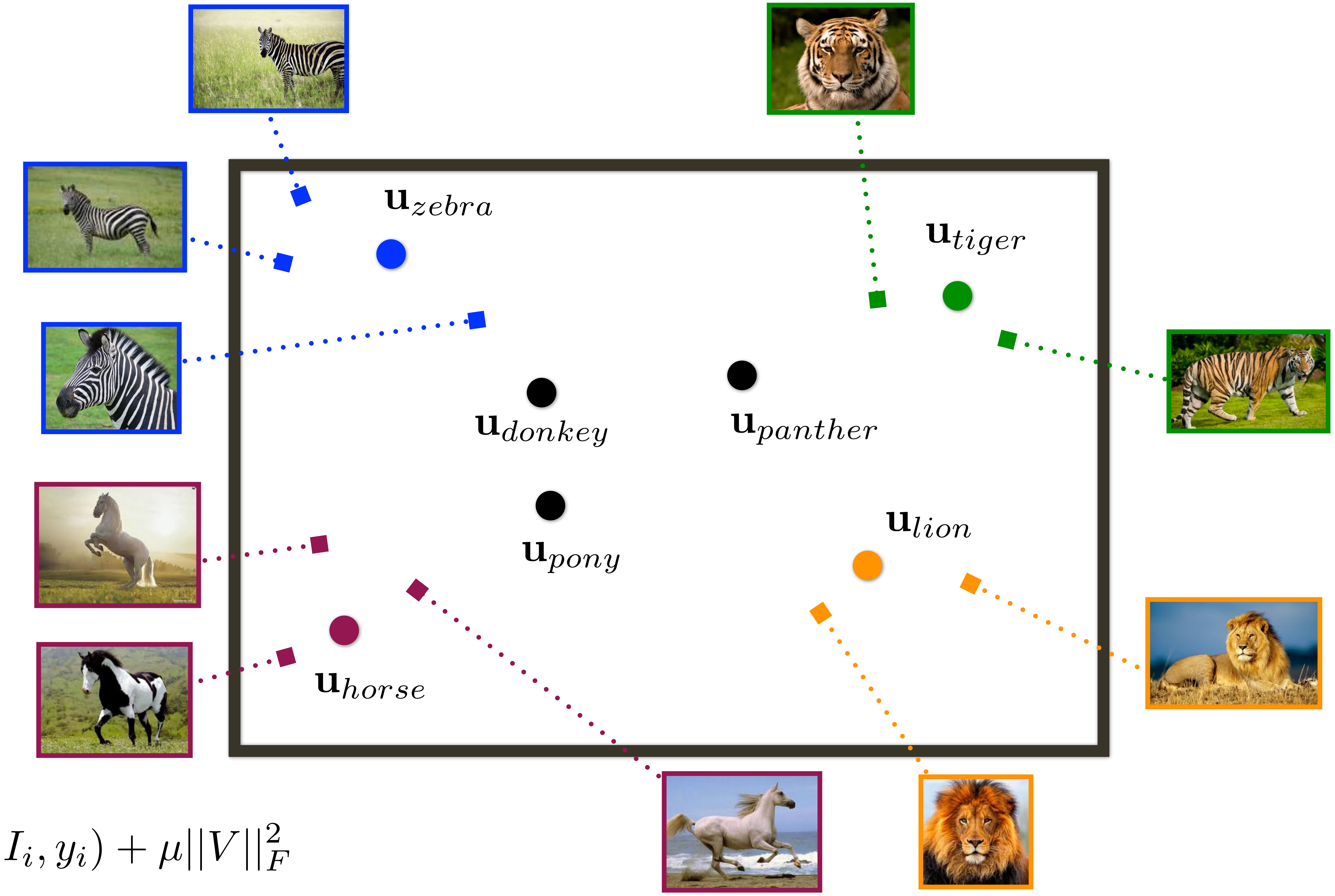
$L = 310,000$

Similarity in Embedding Space

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

Objective Function:

$$\min_{\mathbf{W}} \sum_i^N \mathcal{L}_C(\mathbf{W}, \mathbf{V}, I_i, y_i) + \mathcal{L}_R(\mathbf{W}, \mathbf{V}, I_i, y_i) + \mu||V||_F^2$$



Semi-supervised **Vocabulary Informed** Learning

[Fu et al., 2016]

Image Embedding

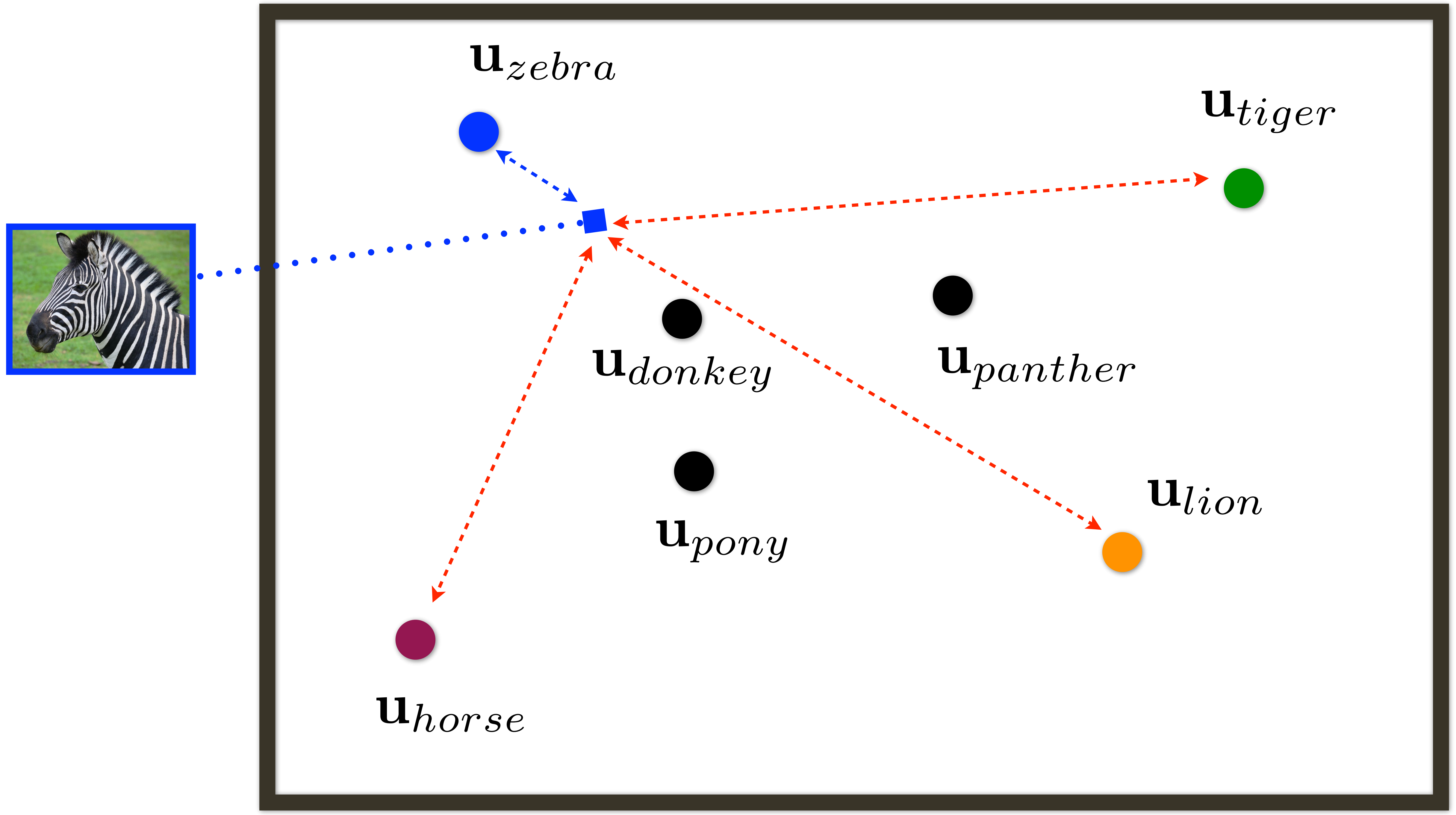
$$\Psi(I_i) = \mathbf{W} \cdot CNN(I_i; \mathbf{\Theta}) : \mathbb{R}^D \rightarrow \mathbb{R}^d$$

Label Embedding

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

$L = 310,000$

$$\mathcal{L}_C(\mathbf{W}, \mathbf{U}, \mathbf{x}_i, y_i) = \sum [1 + D(\mathbf{W}\mathbf{x}_i, \mathbf{u}_{y_i}) - D(\mathbf{W}\mathbf{x}_i, \mathbf{u}_c)]$$



Similarity in Embedding Space

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

Objective Function:

$$\min_{\mathbf{W}} \sum_i^N \mathcal{L}_C(\mathbf{W}, \mathbf{V}, I_i, y_i) + \mathcal{L}_R(\mathbf{W}, \mathbf{V}, I_i, y_i) + \mu ||V||_F^2$$

Semi-supervised **Vocabulary Informed** Learning

[Fu et al., 2016]

Image Embedding

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

Label Embedding

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

$L = 310,000$

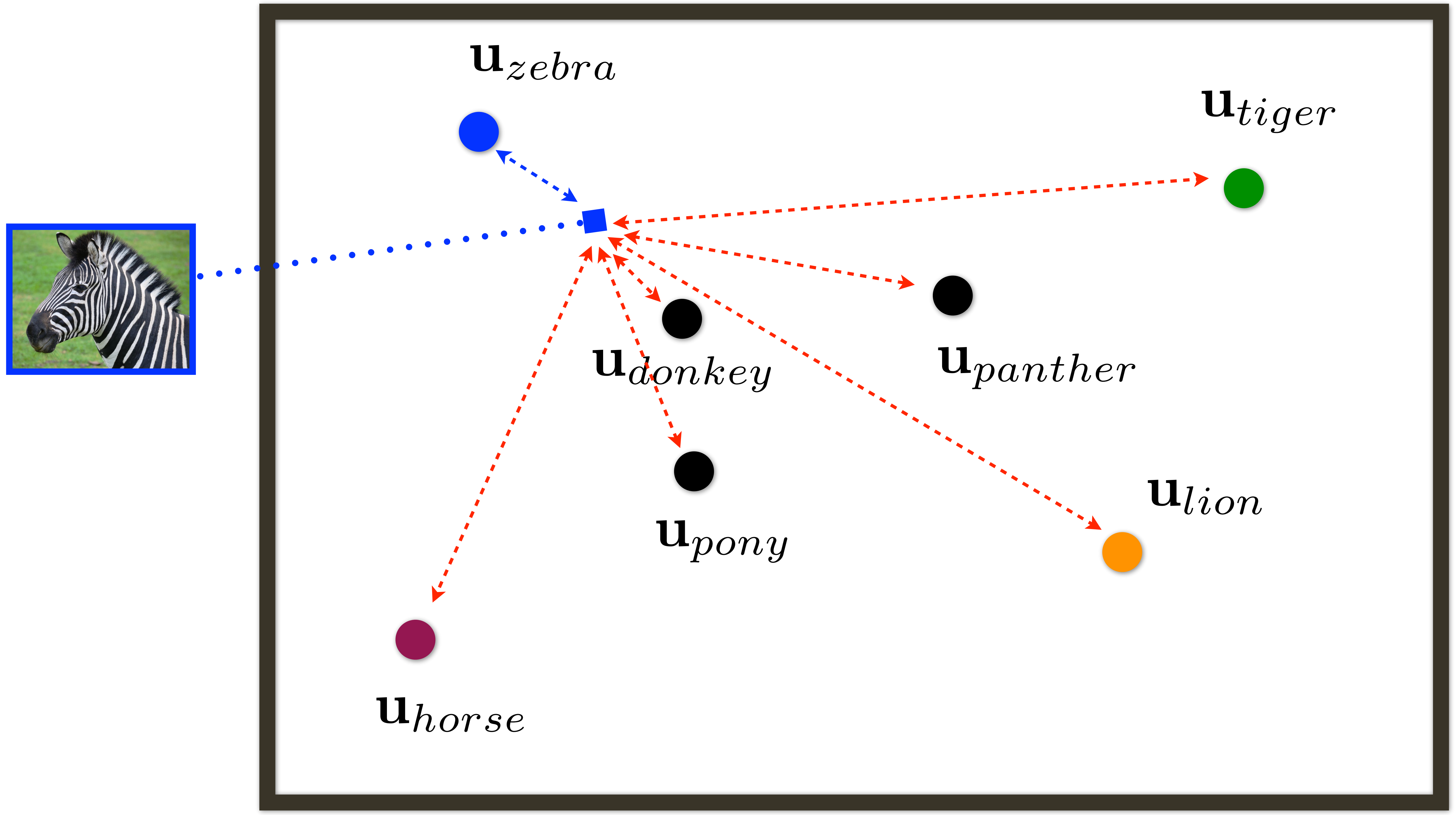
Similarity in Embedding Space

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

Objective Function:

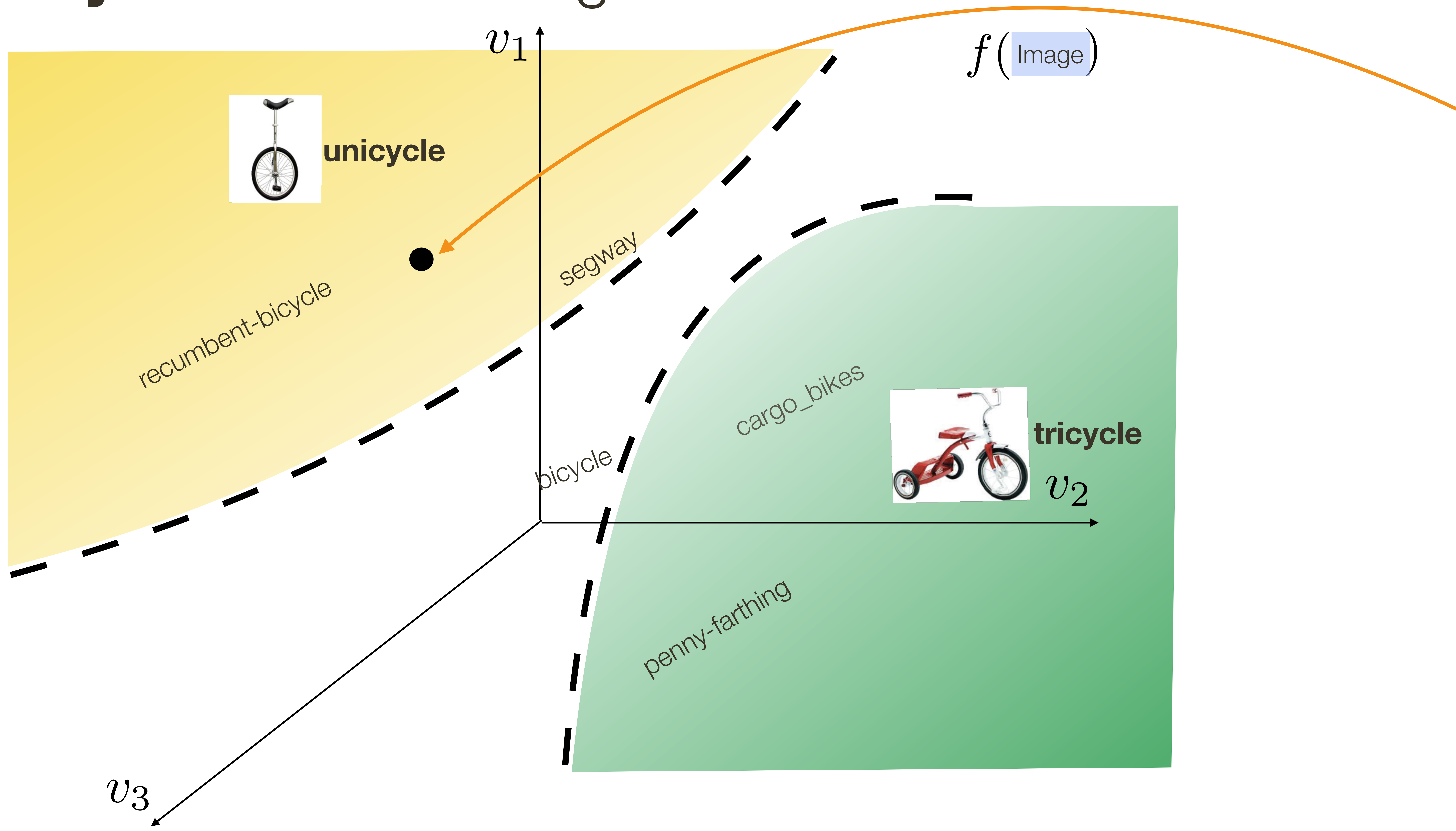
$$\min_{\mathbf{W}} \sum_i^N \mathcal{L}_C(\mathbf{W}, \mathbf{V}, I_i, y_i) + \mathcal{L}_R(\mathbf{W}, \mathbf{V}, I_i, y_i) + \mu||V||_F^2$$

$$\mathcal{L}_C(\mathbf{W}, \mathbf{U}, \mathbf{x}_i, y_i) = \sum [1 + D(\mathbf{W}\mathbf{x}_i, \mathbf{u}_{y_i}) - D(\mathbf{W}\mathbf{x}_i, \mathbf{u}_c)]$$



Vocabulary Informed Recognition

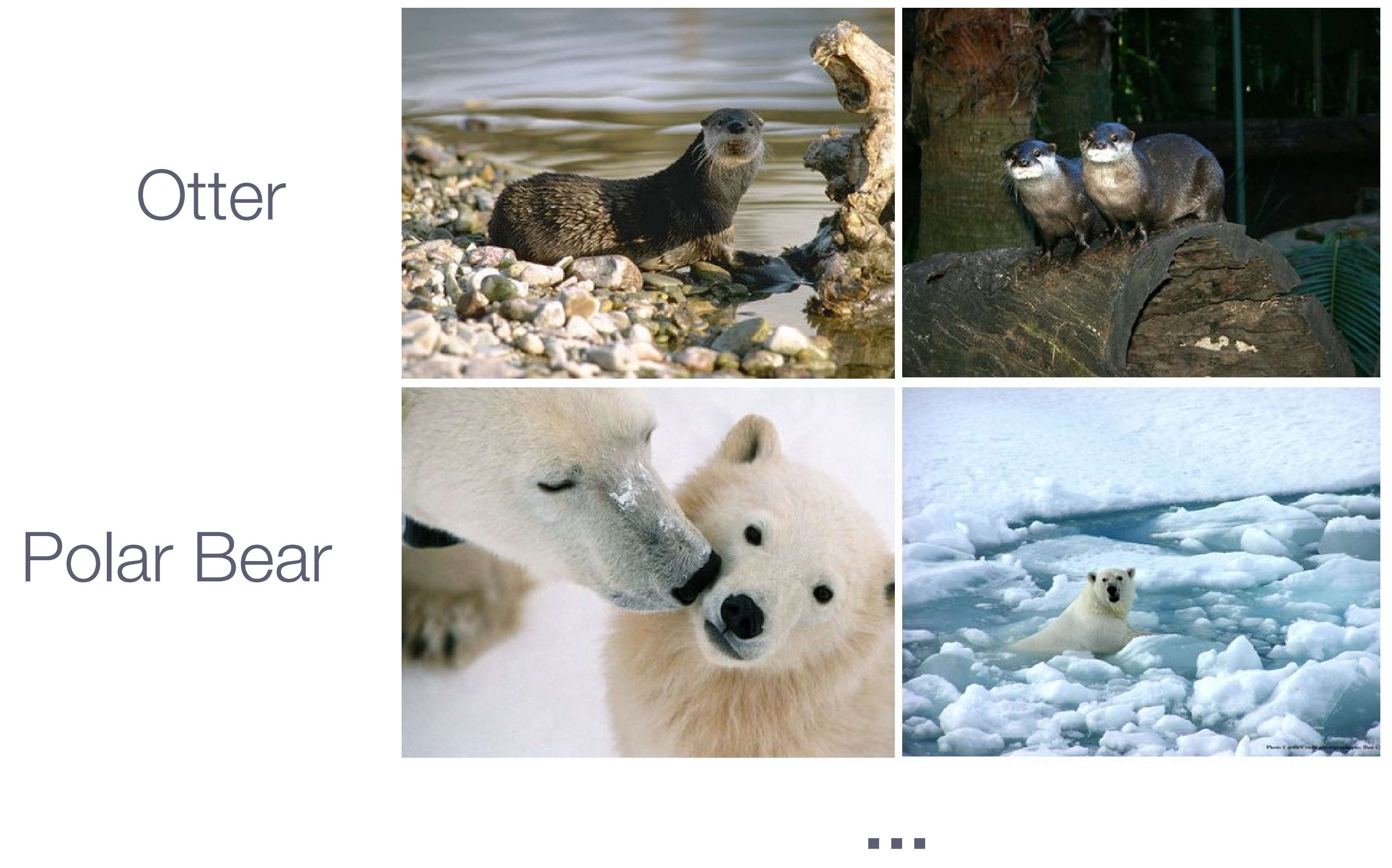
[Fu et al., 2016]



Experiments: Datasets

[Fu et al., 2016]

Animals with Attributes



Auxiliary: 40 Animal Classes (annotated)

Target: 10 Animal Classes (**NO** annotation)

[Lampert, Nickisch, Harmeling CVPR'09]

ImageNet

Auxiliary: 1,000 General Classes (annotated)

Target: 360 General Classes (**NO** annotation)

[Deng *et al.*, CVPR'09]

Experiments: Settings

[Fu et al., 2016]

AwA/ImageNet	No. Testing Classes			No. Testing Words	
	Auxiliary	Target	Total	Vocabulary	Chance(%)
SUPERVISED	✓		40/1000	40/1000	2.5/0.1
ZERO-SHOT		✓	10/360	10/360	10/0.28
OPEN-SET	✓	✓	50/1360	310K/310K	3.2E-04

The tasks are only separated in **evaluation**;
We train **one unified model** for all the settings

Experiments: Settings

[Fu et al., 2016]

Training

Otter

Polar Bear

Donkey

Poney

Panther

310, 000

Testing

Supervised

Zero-shot

Experiments: Settings

[Fu et al., 2016]

Training

Otter

Polar Bear

Donkey

Poney

Panther

310, 000

Testing

Open-set

Experiments: Settings

[Fu et al., 2016]

AwA/ImageNet	No. Testing Classes			No. Testing Words	
	Auxiliary	Target	Total	Vocabulary	Chance(%)
SUPERVISED	✓		40/1000	40/1000	2.5/0.1
ZERO-SHOT		✓	**10**/360	**10**/360	**10**/0.28
OPEN-SET	✓	✓	50/1360	310K/310K	3.2E-04

The tasks are only separated in **evaluation**;
We train **one unified model** for all the settings

Zero-shot Results

[Fu et al., 2016]

Results with AWA

Method	Features	Accuracy
SS-Voc: full instances	$\mathrm{CNN}_{\mathrm{OverFeat}}$	78.3 **+4.4%**
Akata *et al*. CVPR 2015	$\mathrm{CNN}_{\mathrm{GoogLeNet}}$	73.9
TMV-BLP (Fu *et al*. ECCV 2014)	$\mathrm{CNN}_{\mathrm{OverFeat}}$	69.9
AMP (SR+SE) (Fu *et al*. CVPR 2015)	$\mathrm{CNN}_{\mathrm{OverFeat}}$	66.0
DAP (Lampert *et al*. TPAMI 2013)	$\mathrm{CNN}_{\mathrm{VGG19}}$	57.5
PST (Rohrbach *et al*. NIPS 2013)	$\mathrm{CNN}_{\mathrm{OverFeat}}$	53.2
DS (Rohrbach *et al*. CVPR 2010)	$\mathrm{CNN}_{\mathrm{OverFeat}}$	52.7
IAP (Lampert *et al*. TPAMI 2013)	$\mathrm{CNN}_{\mathrm{OverFeat}}$	44.5
HEX (Deng *et al*. ECCV 2014)	$\mathrm{CNN}_{\mathrm{DECAF}}$	44.2

Zero-shot Results

[Fu et al., 2016]

Results with AWA

3.3% of training data

Method	Features	Accuracy
SS-Voc: full instances	$CNN_{OverFeat}$	78.3
800 instances (20 inst*40 class);	$CNN_{OverFeat}$	74.4
Akata *et al*. CVPR 2015	$CNN_{GoogLeNet}$	73.9
TMV-BLP (Fu *et al*. ECCV 2014)	$CNN_{OverFeat}$	69.9
AMP (SR+SE) (Fu *et al*. CVPR 2015)	$CNN_{OverFeat}$	66.0
DAP (Lampert *et al*. TPAMI 2013)	CNN_{VGG19}	57.5
PST (Rohrbach *et al*. NIPS 2013)	$CNN_{OverFeat}$	53.2
DS (Rohrbach *et al*. CVPR 2010)	$CNN_{OverFeat}$	52.7
IAP (Lampert *et al.* TPAMI 2013)	$CNN_{OverFeat}$	44.5
HEX (Deng *et al*. ECCV 2014)	CNN_{DECAF}	44.2

+0.5%

Zero-shot Results

[Fu et al., 2016]

Results with AWA

0.82% of training data

Method	Features	Accuracy
SS-Voc: full instances	CNN$_{\text{OverFeat}}$	78.3
800 instances (20 inst*40 class);	CNN$_{\text{OverFeat}}$	74.4
200 instances (5 inst*40 class);	CNN$_{\text{OverFeat}}$	68.9
Akata *et al*. CVPR 2015	CNN$_{\text{GoogLeNet}}$	73.9
TMV-BLP (Fu *et al*. ECCV 2014)	CNN$_{\text{OverFeat}}$	69.9
AMP (SR+SE) (Fu *et al*. CVPR 2015)	CNN$_{\text{OverFeat}}$	66.0
DAP (Lampert *et al*. TPAMI 2013)	CNN$_{\text{VGG19}}$	57.5
PST (Rohrbach *et al*. NIPS 2013)	CNN$_{\text{OverFeat}}$	53.2
DS (Rohrbach *et al*. CVPR 2010)	CNN$_{\text{OverFeat}}$	52.7
IAP (Lampert *et al.* TPAMI 2013)	CNN$_{\text{OverFeat}}$	44.5
HEX (Deng *et al*. ECCV 2014)	CNN$_{\text{DECAF}}$	44.2

Weakly-supervised **Visual Grounding** of Phrases

[Xiao et al., 2017]

Given **image-sentence pairs** learn how to **localize** arbitrary language phrase or sentence in new images

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

A large bus sitting next to a very tall building.

[Lin, Maire, Belongie, Hays, Perona, Ramanan, Dollar, Zitnick, ECCV'14]

Weakly-supervised **Visual Grounding** of Phrases

[Xiao et al., 2017]

Given **image-sentence pairs** learn how to **localize** arbitrary language phrase or sentence in new images

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

A large bus sitting next to a very tall building.

[Lin, Maire, Belongie, Hays, Perona, Ramanan, Dollar, Zitnick, ECCV'14]

a man

Weakly-supervised **Visual Grounding** of Phrases

[Xiao et al., 2017]

Given **image-sentence pairs** learn how to **localize** arbitrary language phrase or sentence in new images

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

A large bus sitting next to a very tall building.

[Lin, Maire, Belongie, Hays, Perona, Ramanan, Dollar, Zitnick, ECCV'14]

a table

Weakly-supervised **Visual Grounding** of Phrases

[Xiao et al., 2017]

Label Embedding

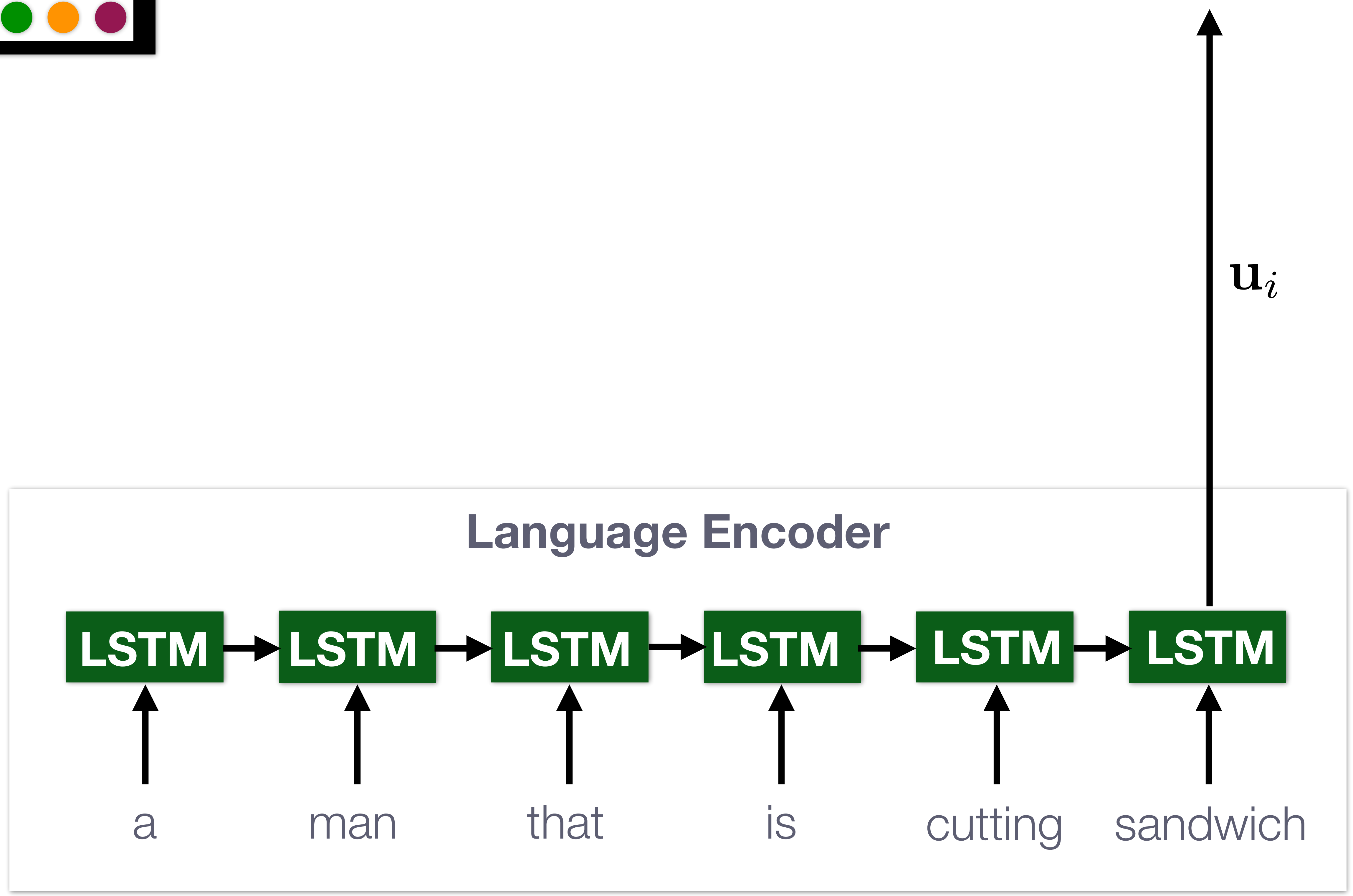
$$\Psi_L(phrase_i) = \mathbf{u}_i$$

Weakly-supervised **Visual Grounding** of Phrases

[Xiao et al., 2017]

Label Embedding

$$\Psi_L(phrase_i) = \mathbf{u}_i$$

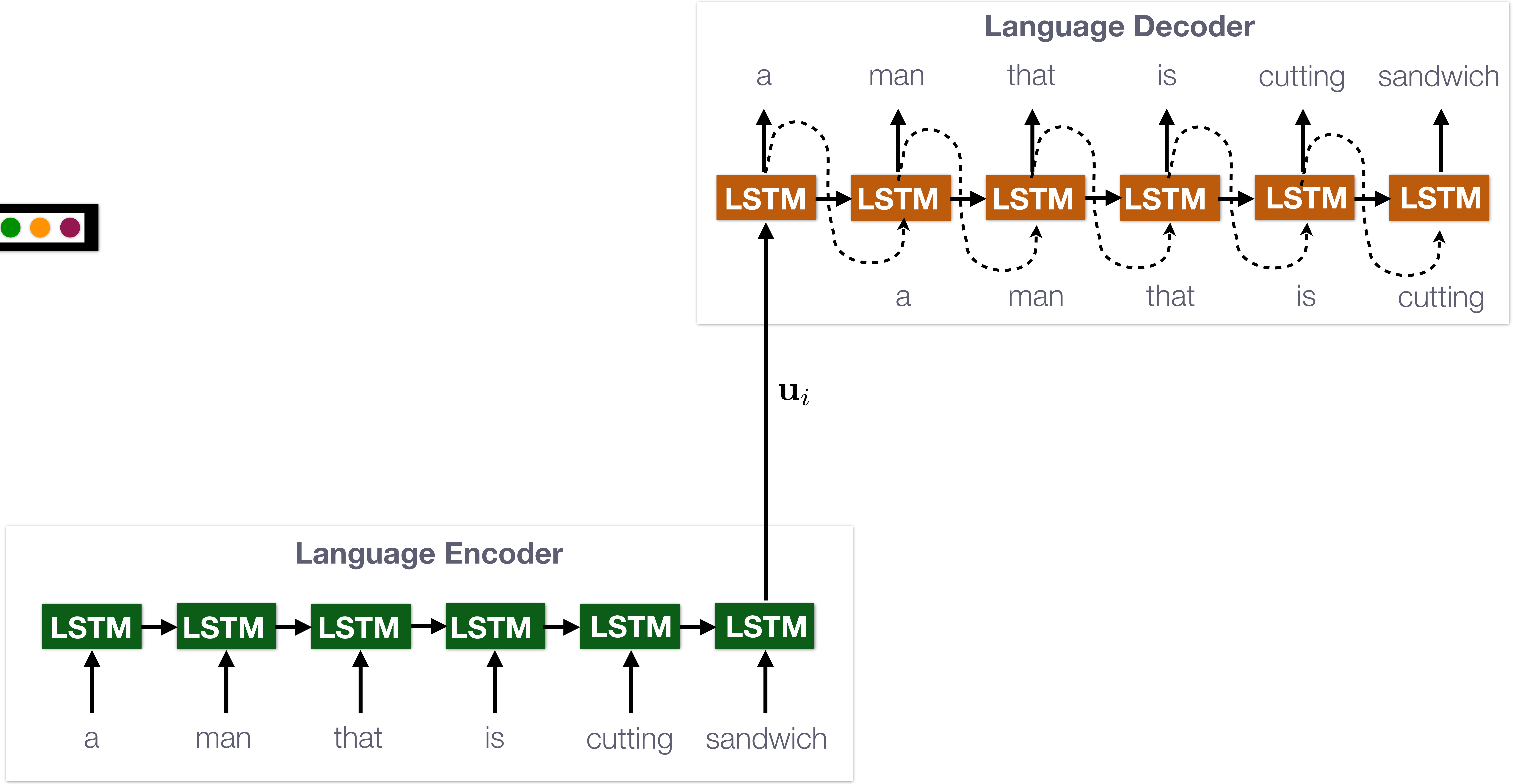


Weakly-supervised **Visual Grounding** of Phrases

[Xiao et al., 2017]

Label Embedding

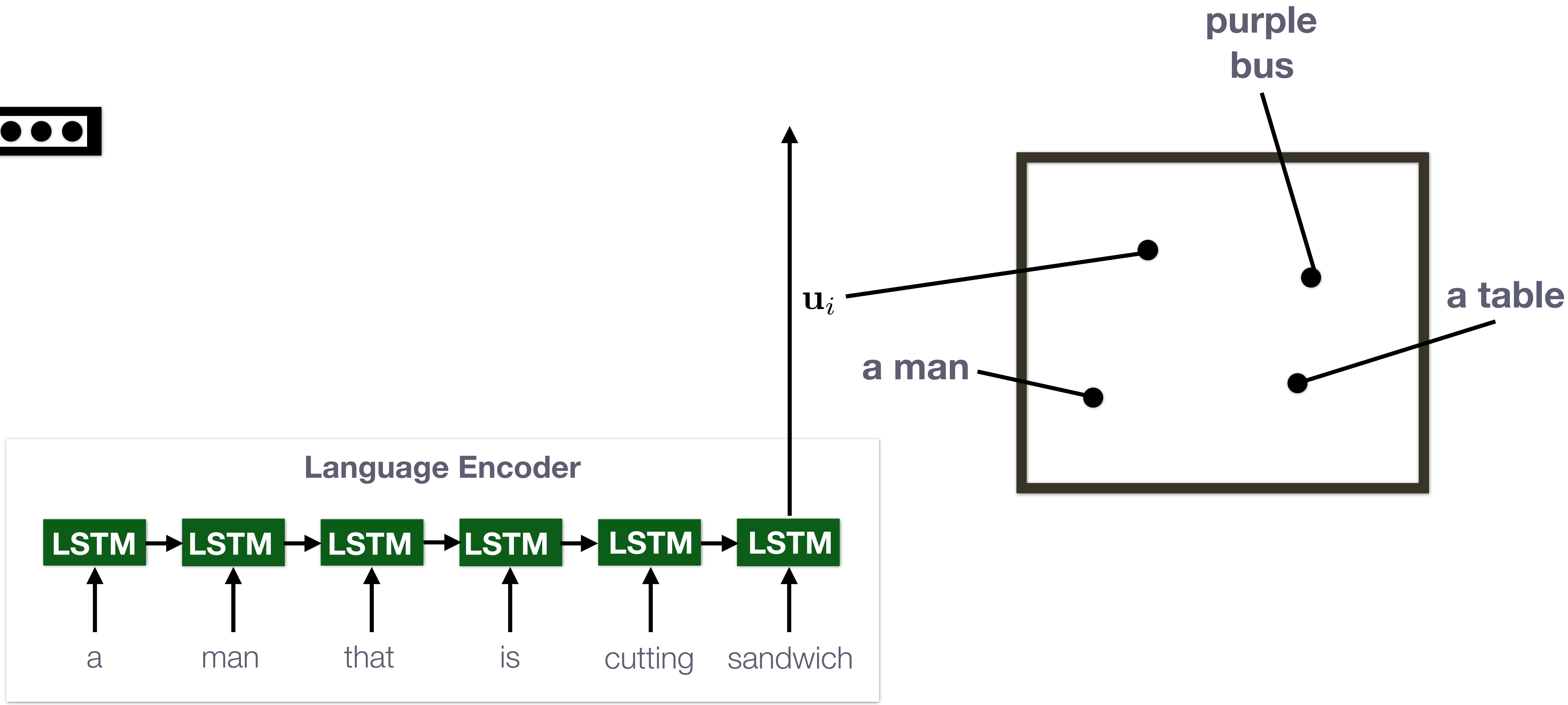
$$\Psi_L(phrase_i) = \mathbf{u}_i$$



Weakly-supervised **Visual Grounding** of Phrases [Xiao et al., 2017]

Label Embedding

$$\Psi_L(phrase_i) = \mathbf{u}_i$$

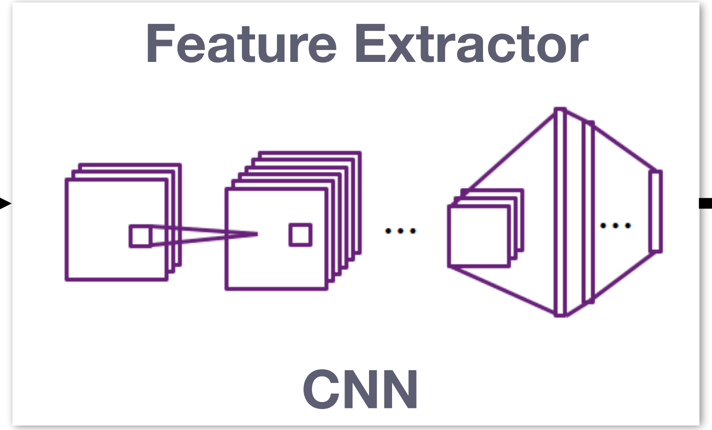


Weakly-supervised **Visual Grounding** of Phrases

[Xiao et al., 2017]

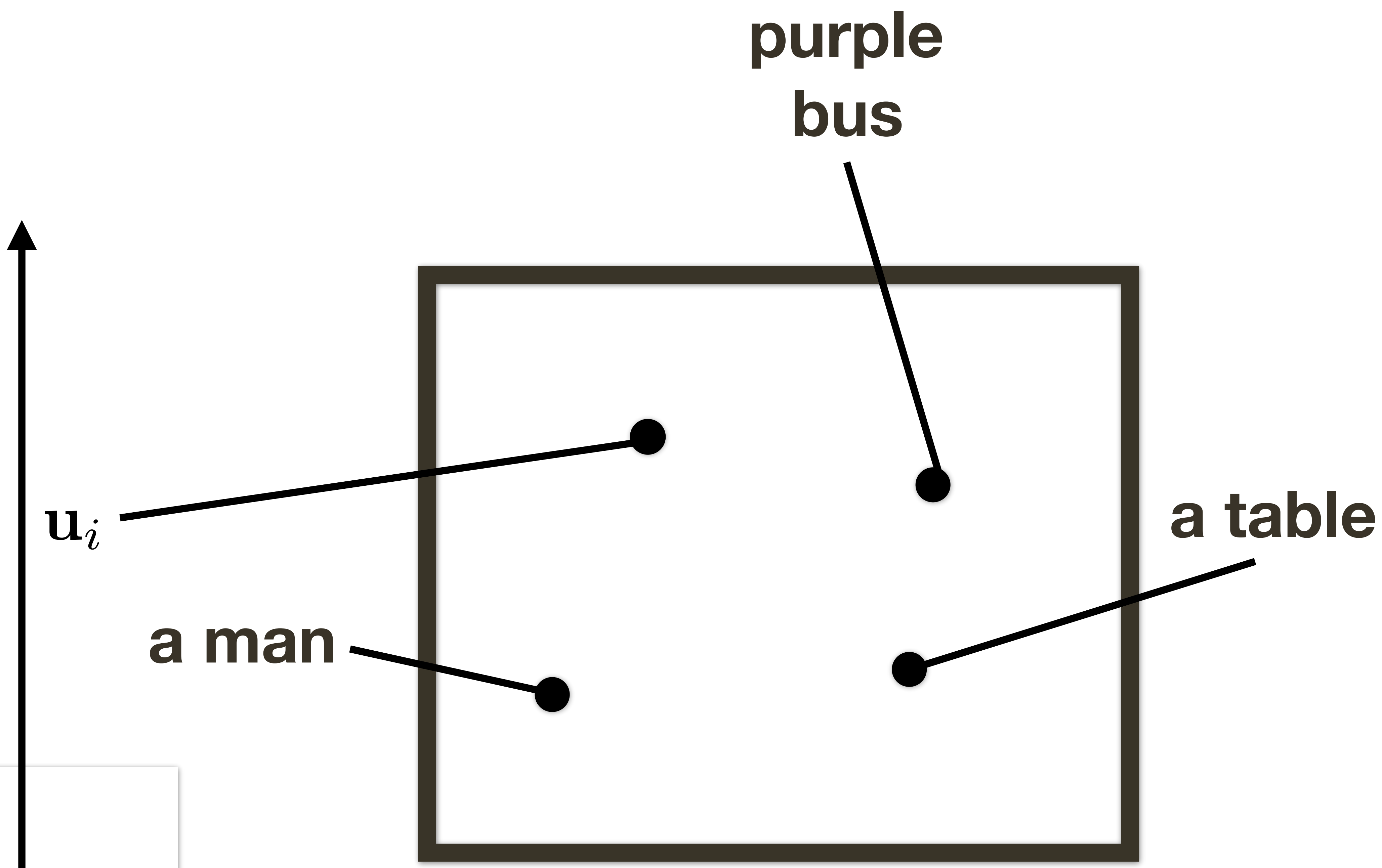
Image Embedding

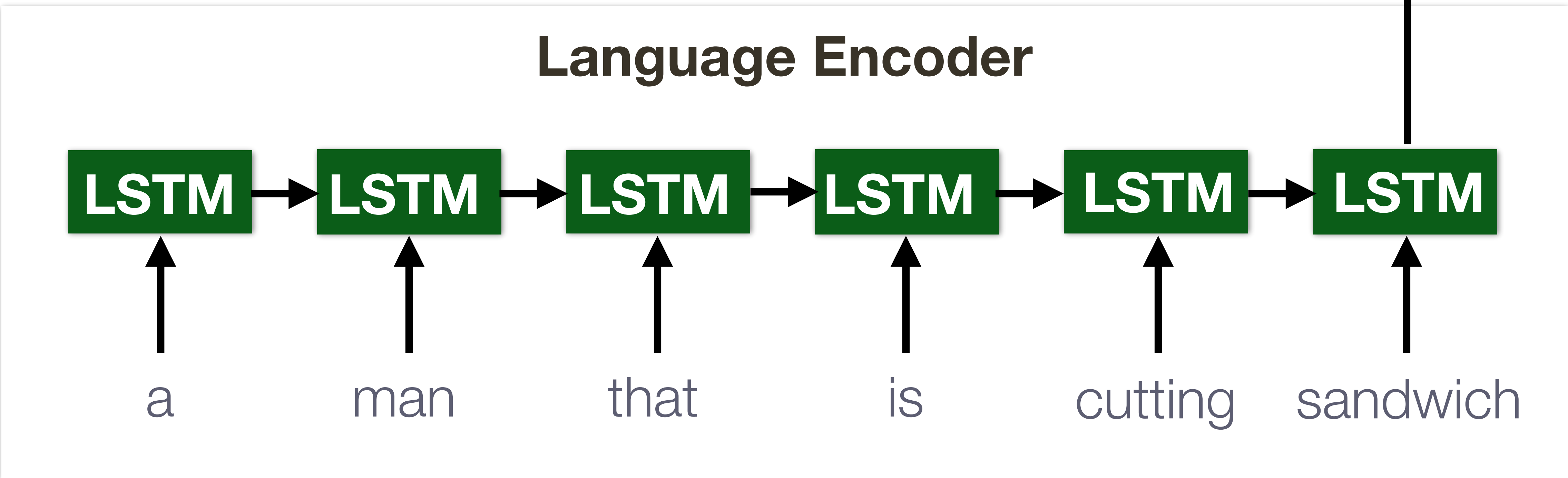
$$\Psi(I_i) = \mathbf{W} \cdot CNN(I_i; \mathbf{\Theta})$$



Label Embedding

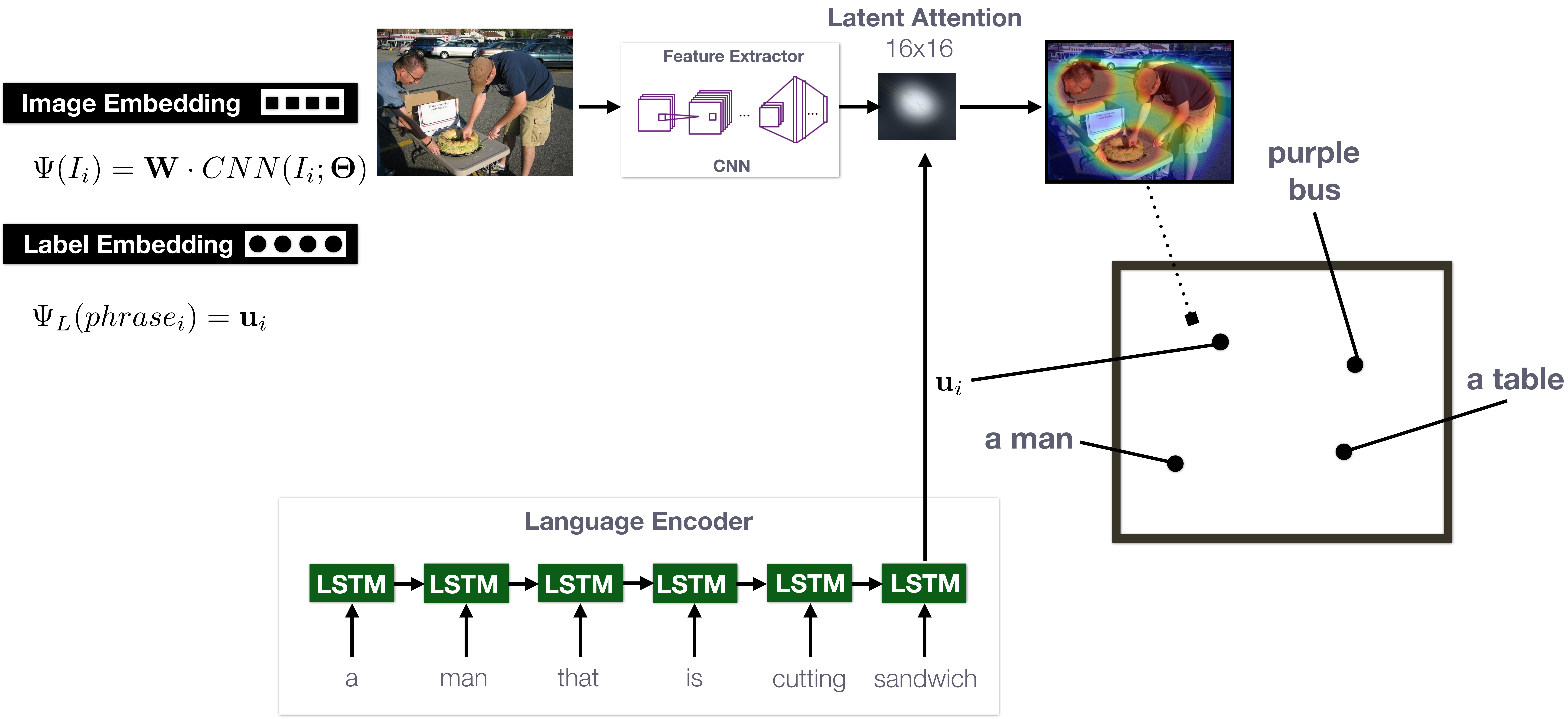
$$\Psi_L(phrase_i) = \mathbf{u}_i$$





Weakly-supervised **Visual Grounding** of Phrases

[Xiao et al., 2017]



Weakly-supervised **Visual Grounding** of Phrases

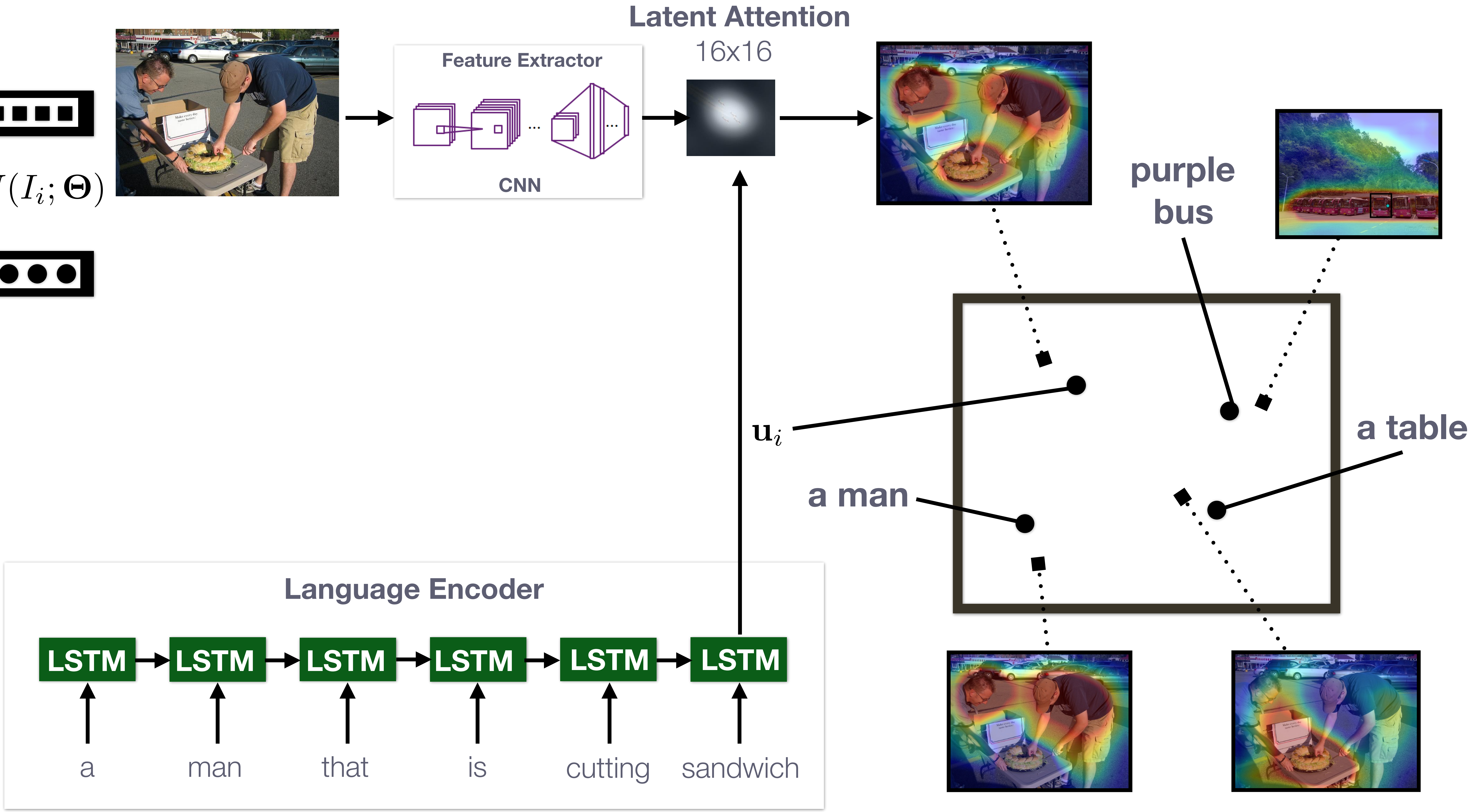
[Xiao et al., 2017]

Image Embedding

$$\Psi(I_i) = \mathbf{W} \cdot CNN(I_i; \mathbf{\Theta})$$

Label Embedding

$$\Psi_L(phrase_i) = \mathbf{u}_i$$



Weakly-supervised **Visual Grounding** of Phrases

[Xiao et al., 2017]

Image Embedding

$$\Psi(I_i) = \mathbf{W} \cdot CNN(I_i; \mathbf{\Theta})$$

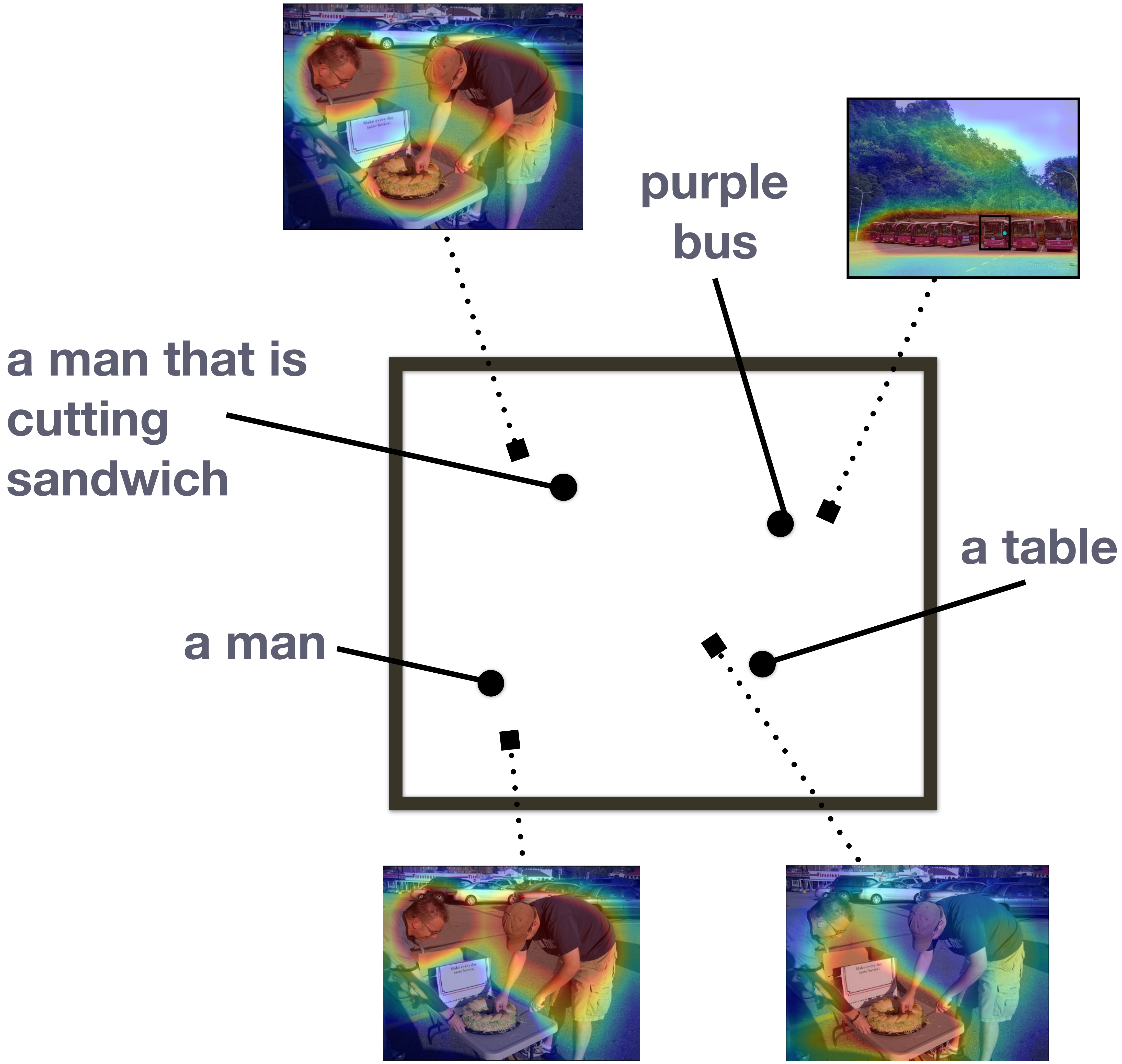
Label Embedding

$$\Psi_L(phrase_i) = \mathbf{u}_i$$

Similarity in Embedding Space

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

Objective Function:



Combination of previous discriminative similarity and **linguistic regularization**

Weakly-supervised **Visual Grounding** of Phrases

[Xiao et al., 2017]

For **noun phrases**:

- **siblings** should have **disjoin**
- **parents** should be **union of**

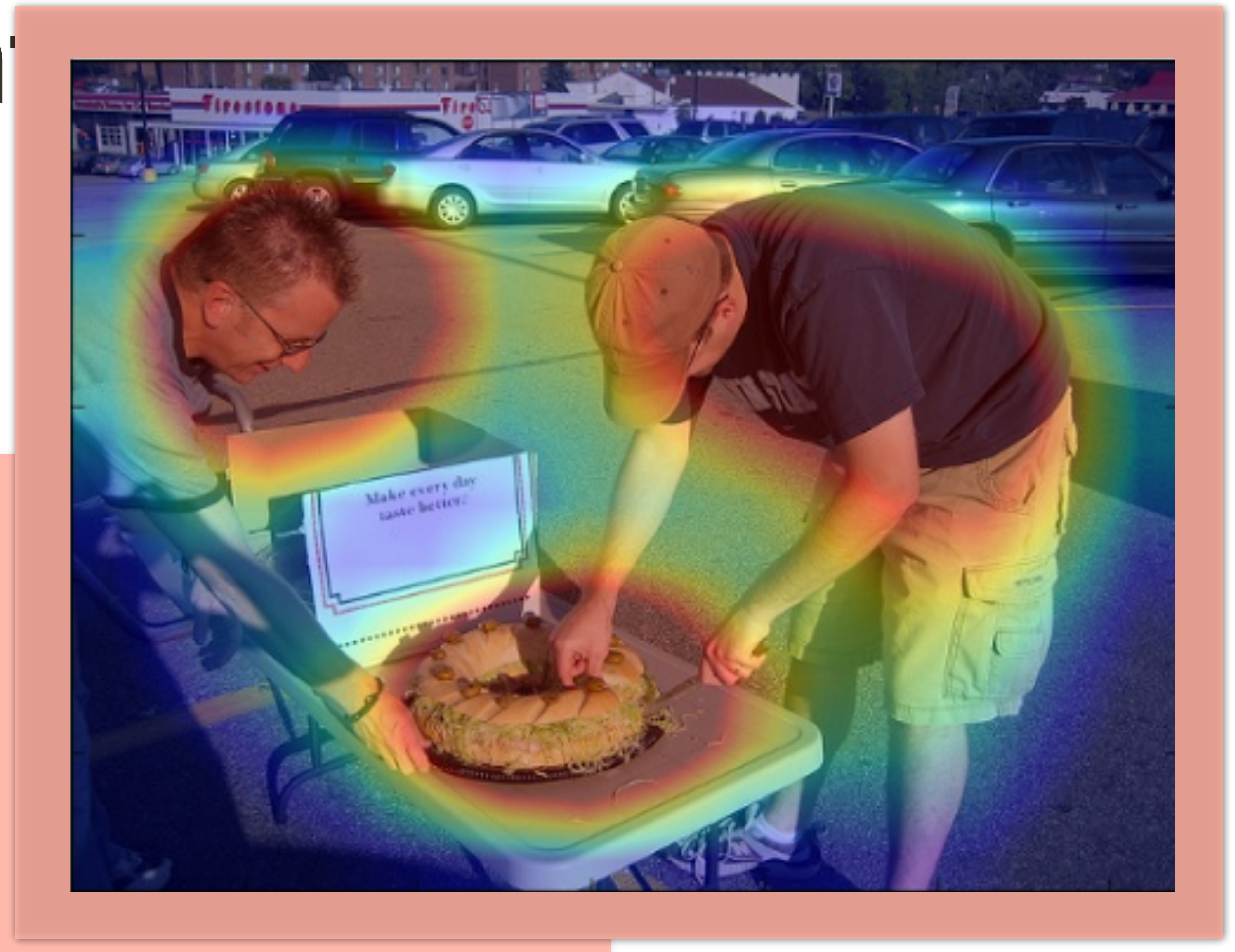


Image Embedding

$$\Psi(I_i) = \mathbf{W} \cdot CNN(I_i; \mathbf{\Theta})$$

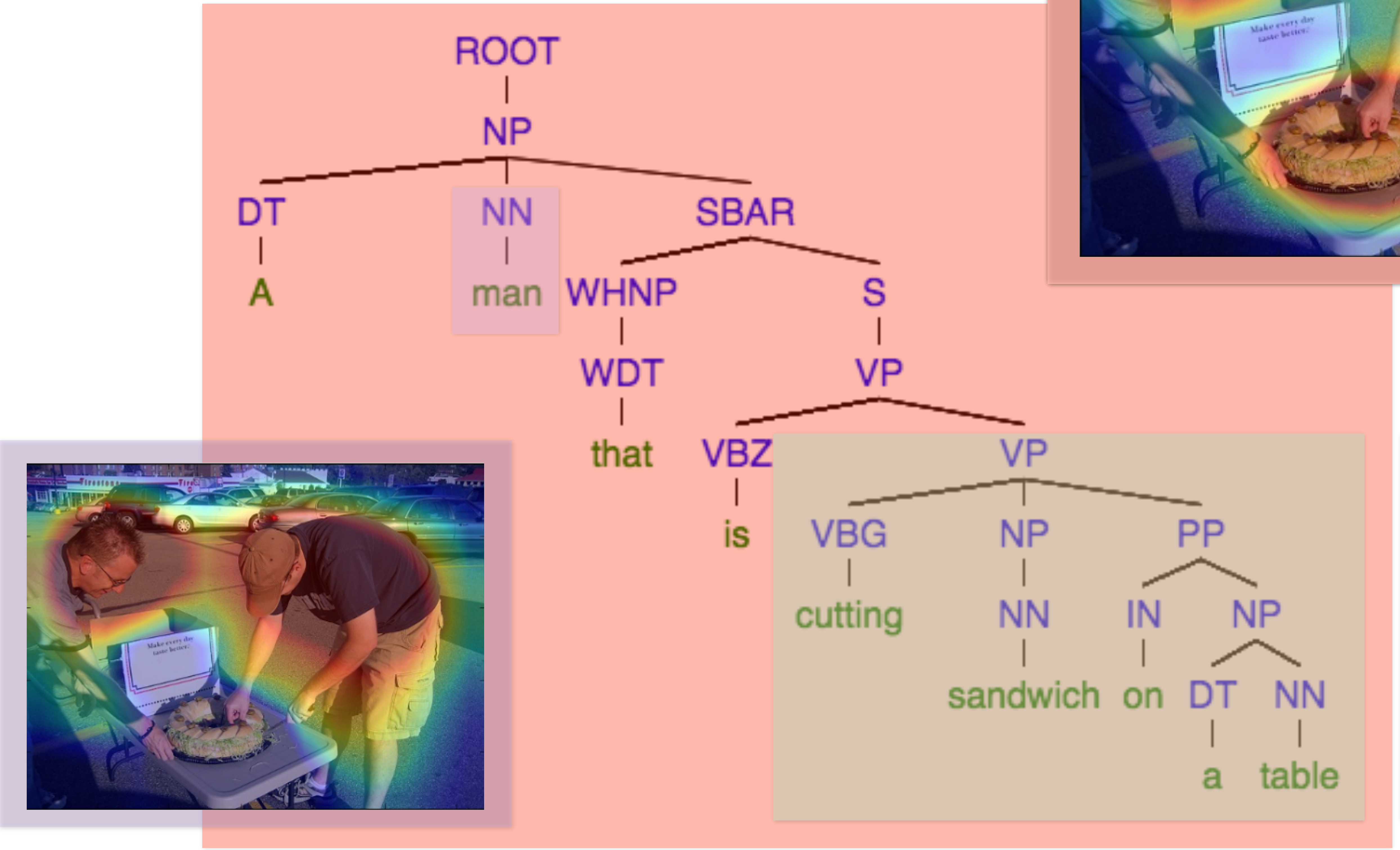
Label Embedding

$$\Psi_L(phrase_i) = \mathbf{u}_i$$

Similarity in Embedding Space

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

Objective Function:



Combination of previous discriminative similarity and **linguistic regularization**

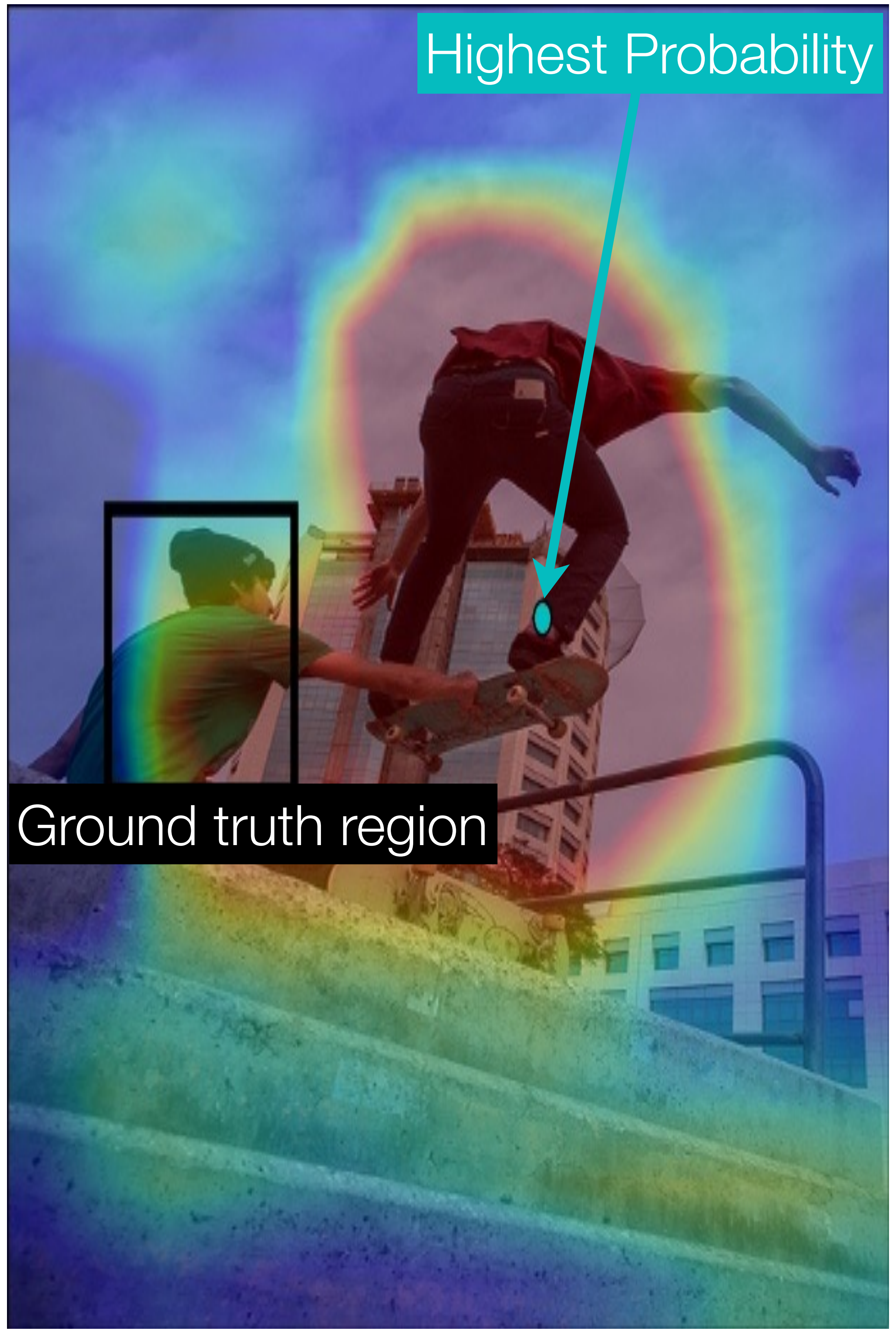
Qualitative **Results**

[Xiao et al., 2017]

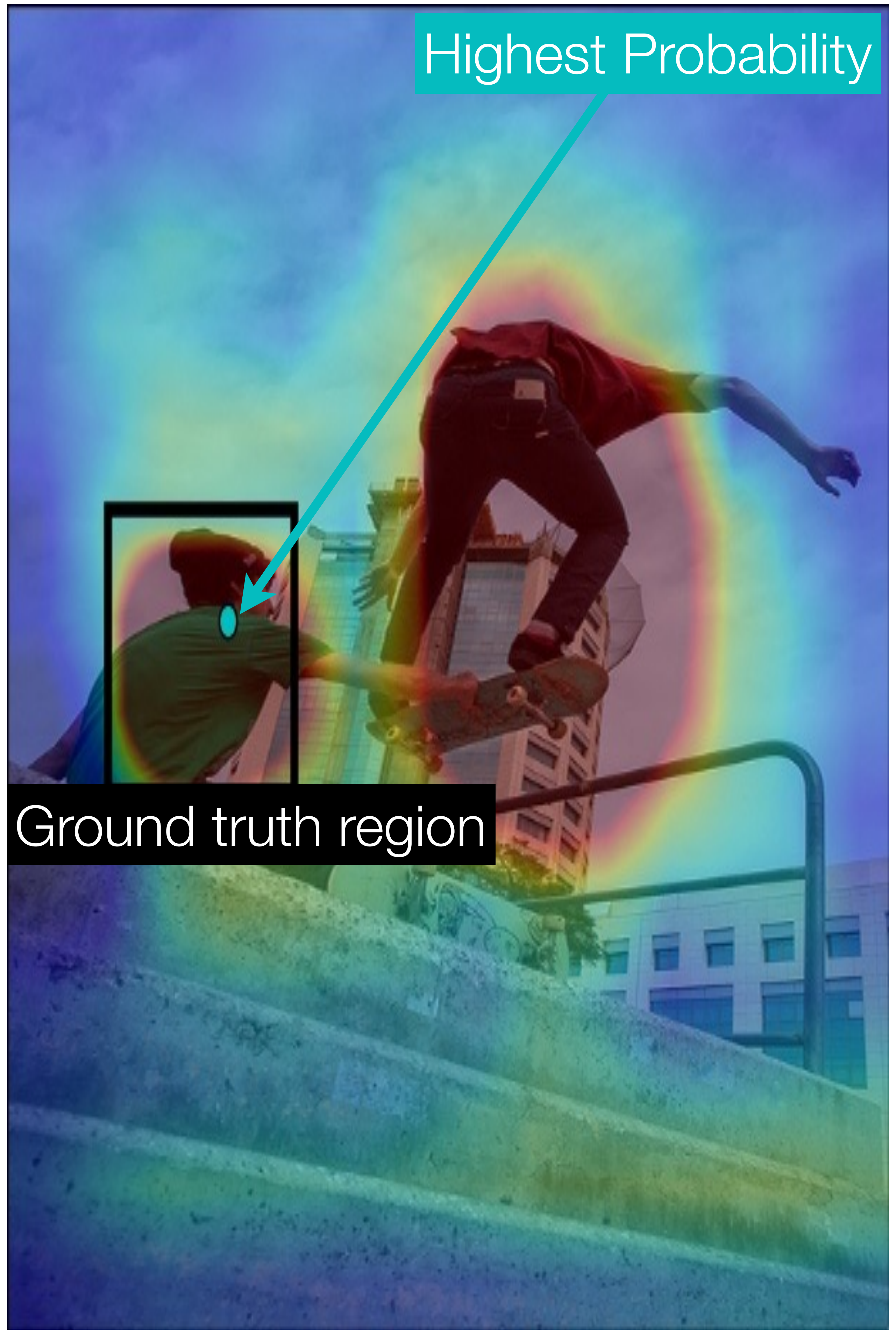
Input:

guy in green t-shirt holding skateboard

NO linguistic constraints



Our Model



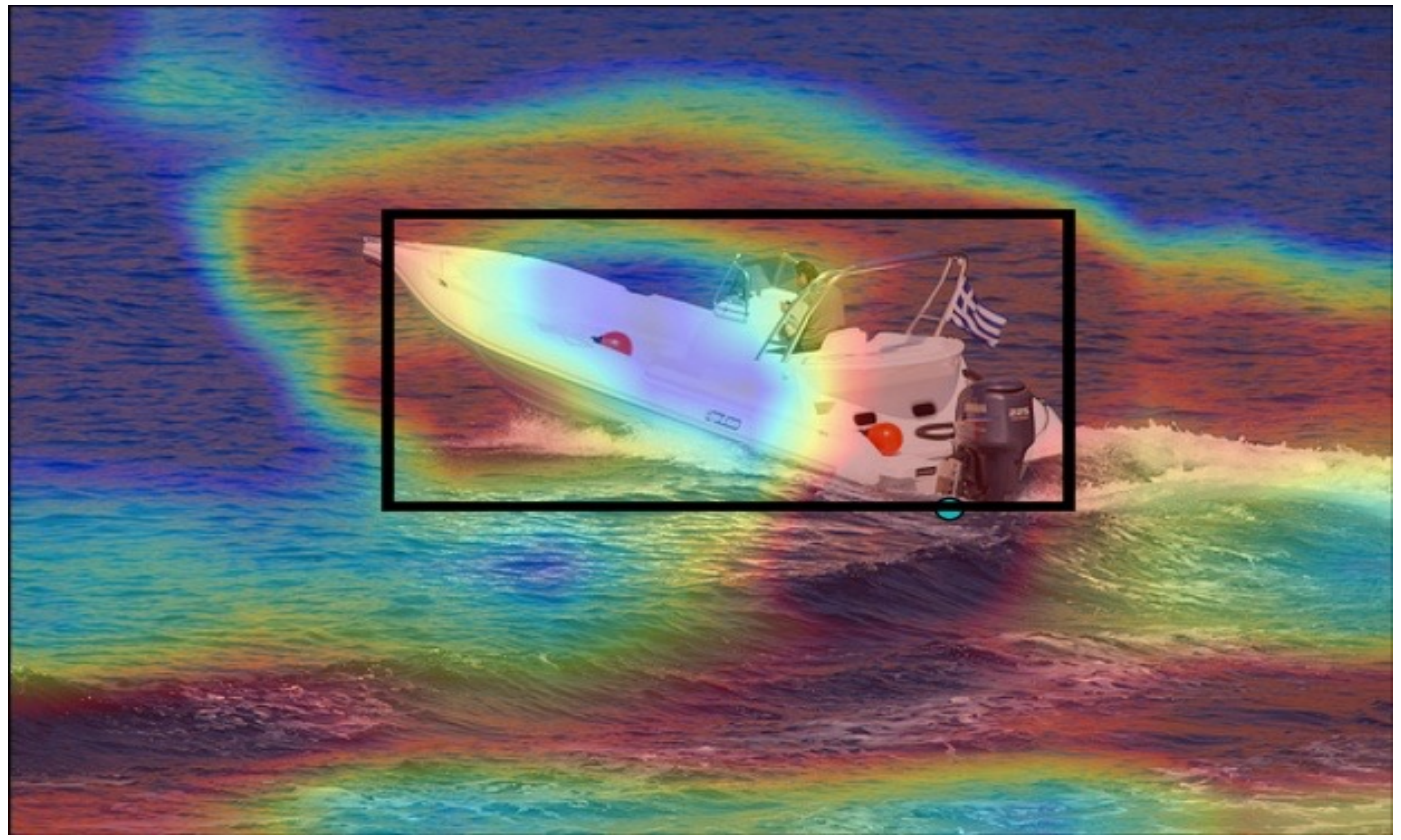
Qualitative **Results**

[Xiao et al., 2017]

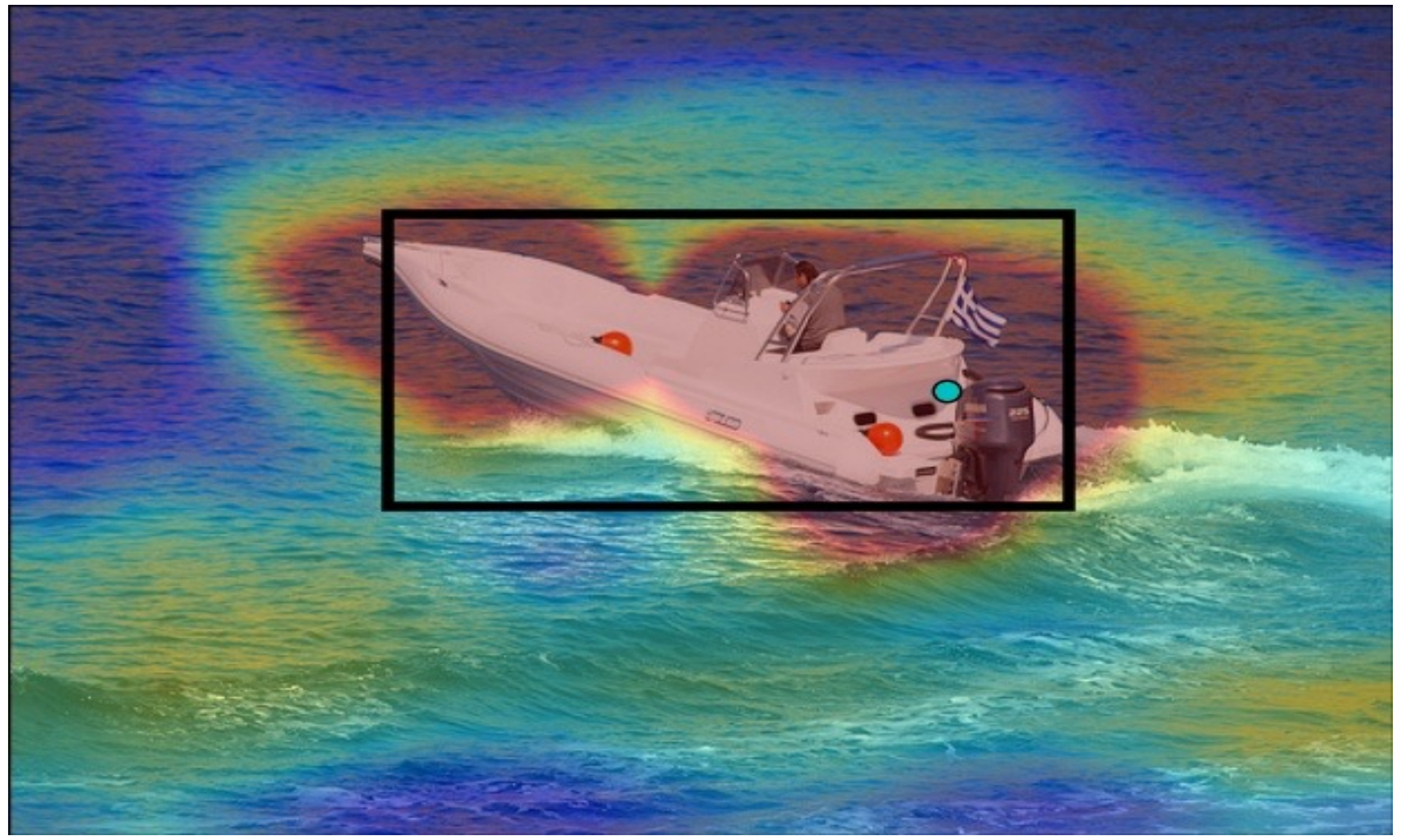
Input:

a person driving a boat

NO linguistic constraints



Our Model

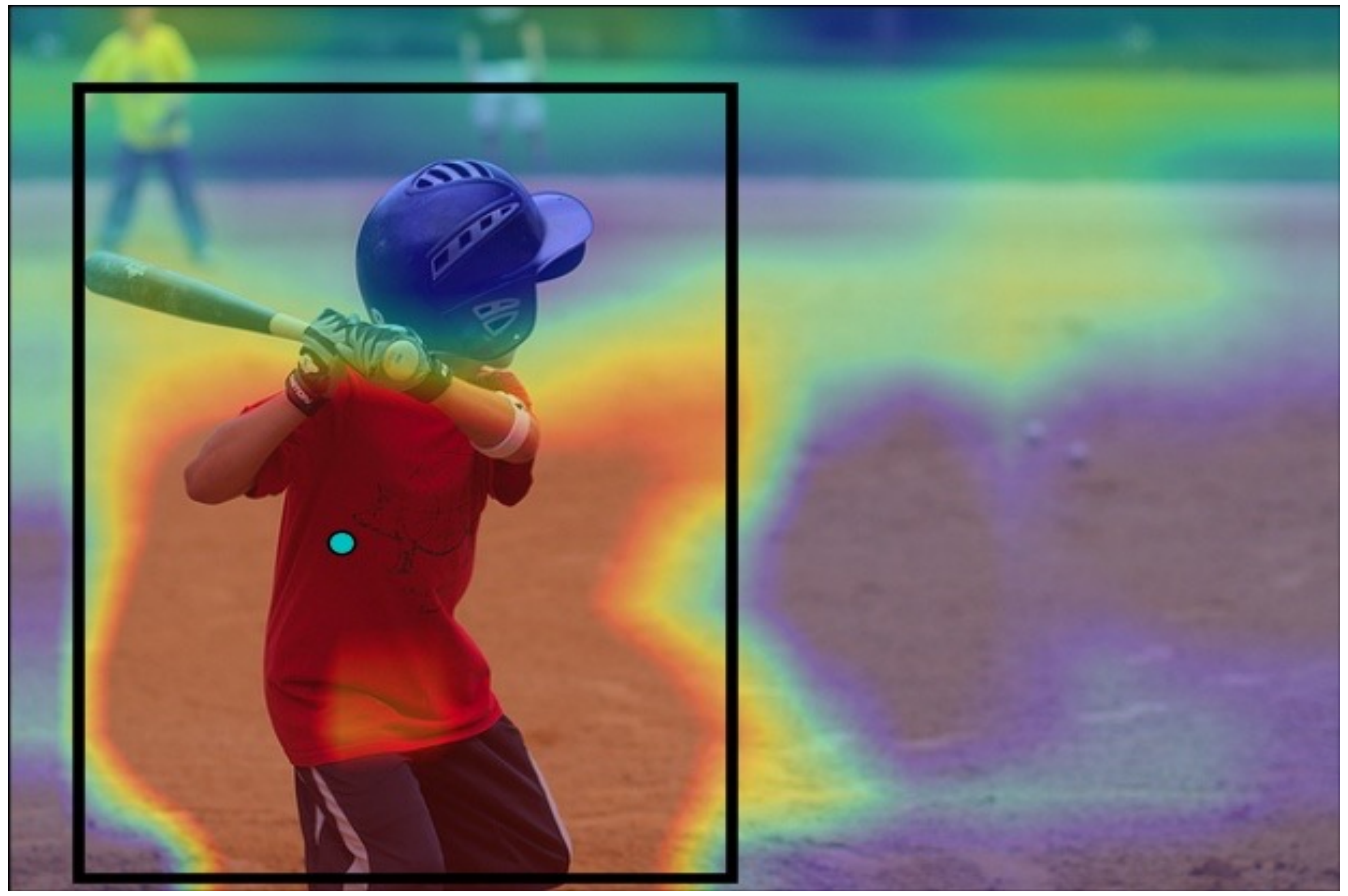


Qualitative **Results**

Input:

a child wearing black protective helmet

NO linguistic constraints [Xiao et al., 2017]



Our Model

Quantitative Results

[Xiao et al., 2017]

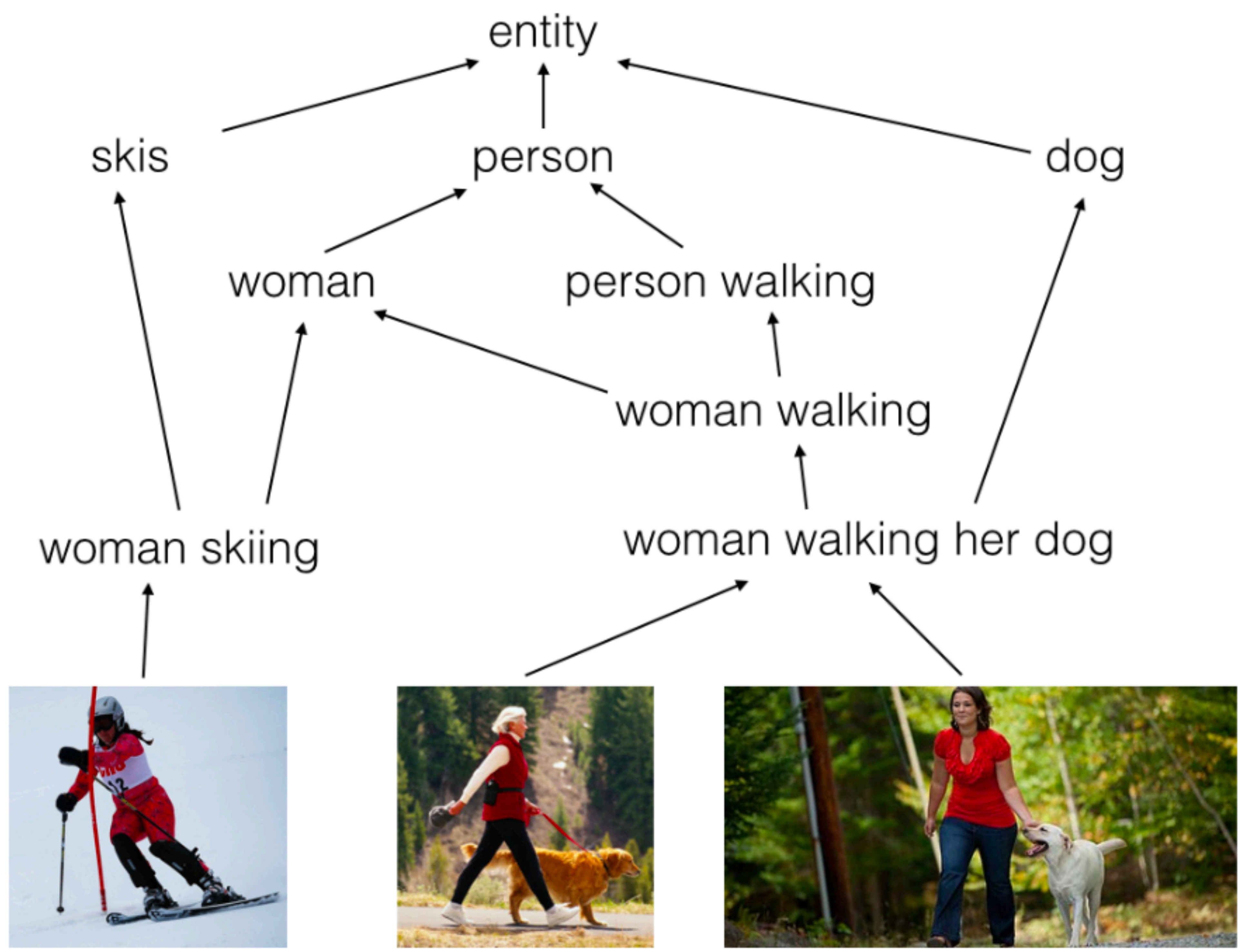
Segmentation performance on COCO dataset

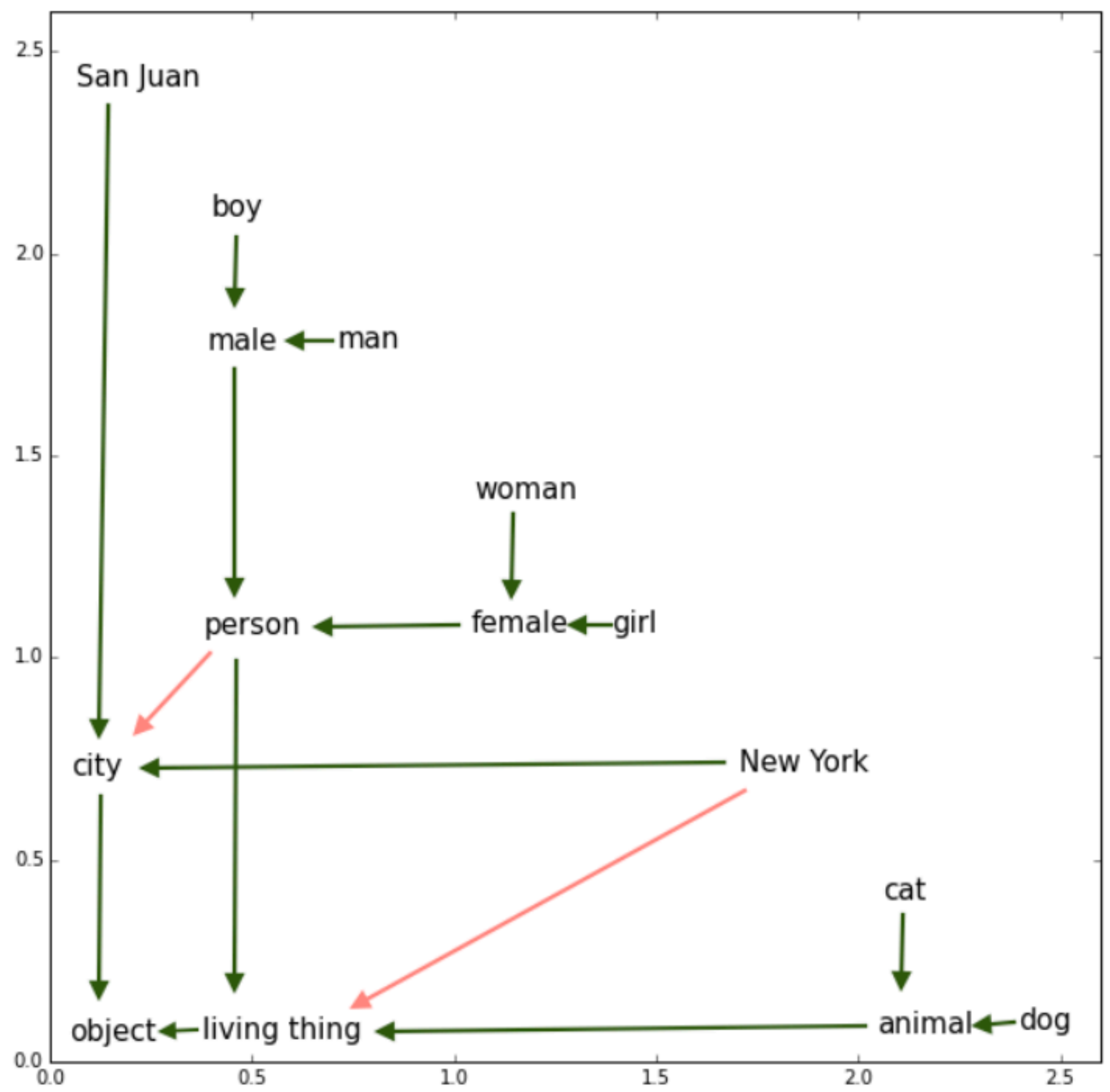
[Lin, Maire, Belongie, Hays, Perona, Ramanan, Dollar, Zitnick, ECCV'14]

	IoU@0.3	**IoU@0.4**	**IoU@0.5**	**Avg mAP**
Non-strcutred	0.302	0.199	0.110	0.203
Parent-Child	0.327	0.213	0.118	0.219
Sibling	0.316	0.203	0.114	0.211
Ours	**0.347**	**0.246**	**0.159**	**0.251**

Order Embeddings

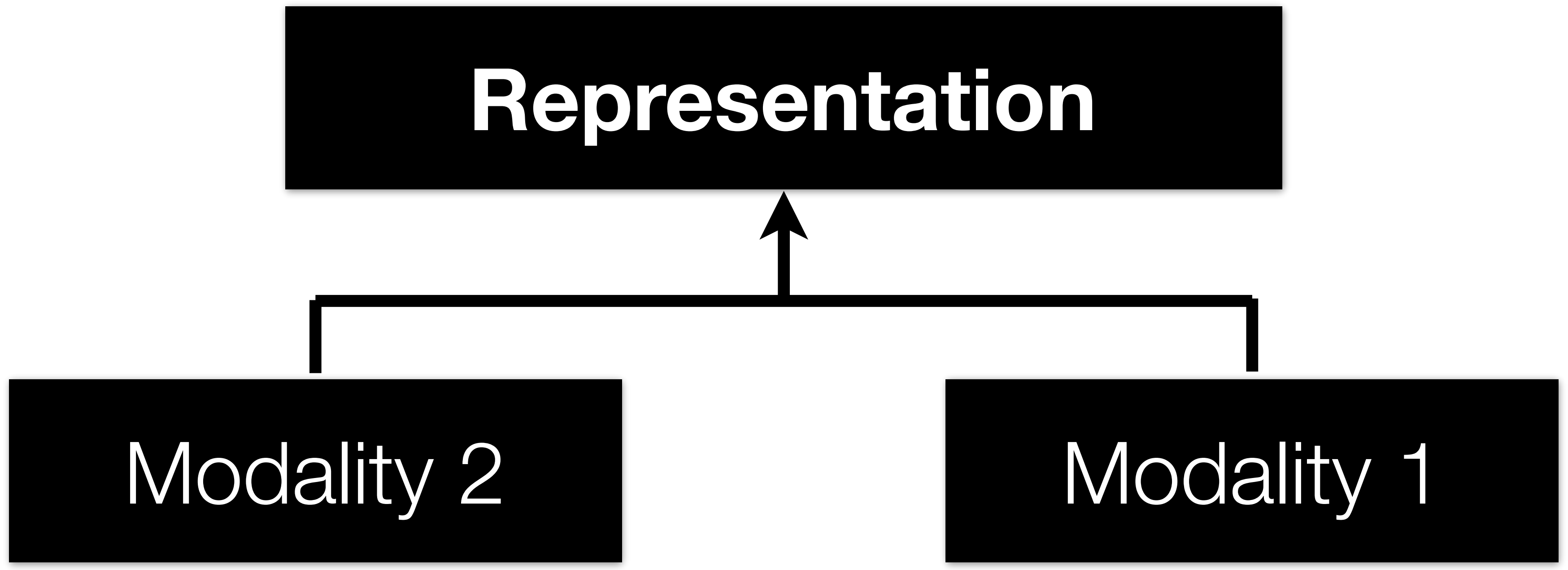
[Vendrov et al., 2016]





Multimodal Representation Types

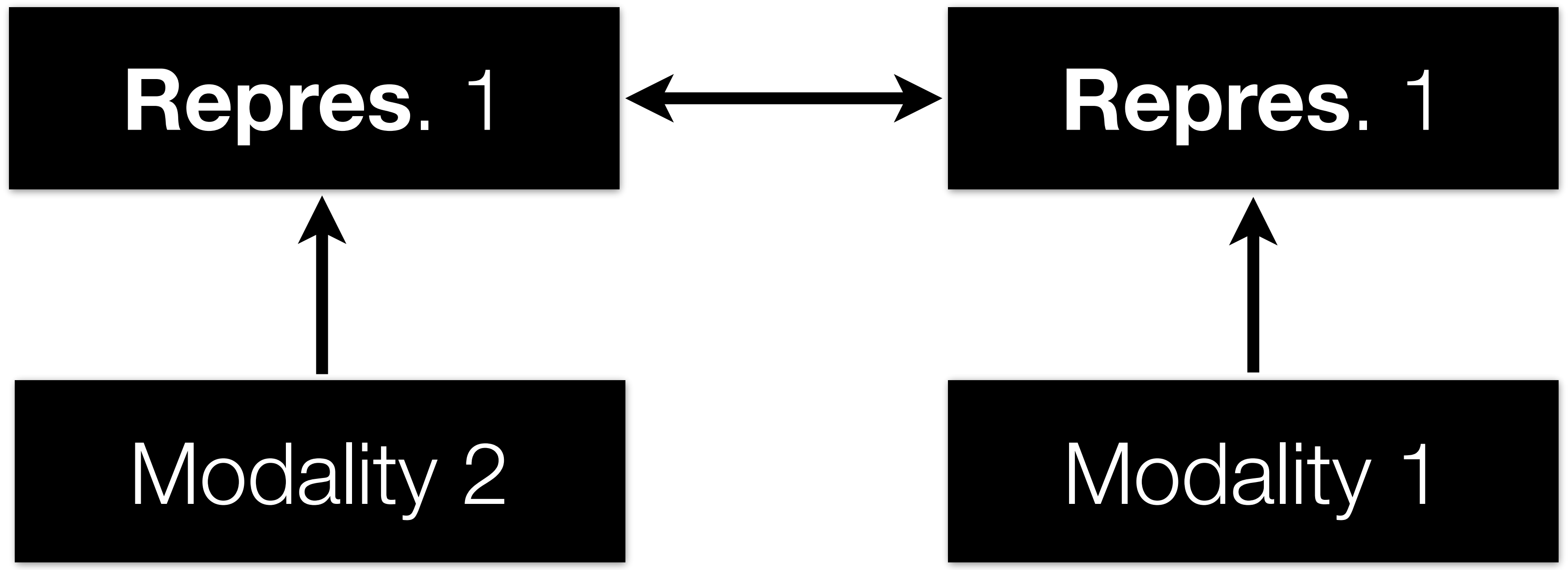
Joint representations:



— 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)

— CCA (unsupervised), joint embeddings (supervised)

*slide from Louis-Philippe Morency

Final Words …

Joint representations

- Project modalities to the same space
- Use when all the modalities are present during test time
- Suitable for multi-model fusion

Coordinated representations

- Project modalities to their own coordinated spaces
- Use when only one of the modalities is present during test-time
- Suitable for multimodal translation
- Good for multimodal retrieval

*slide from Louis-Philippe Morency