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*
Each *modality* can be pre-trained
  – using denoising autoencoder

To train the model, **reconstruct both modalities** using
  – both Audio & Video
  – just Audio
  – just Video
Supervised Joint Representation

For supervised learning 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 learning tasks, we need to join unimodal representations

- Simple **concatenation**
- Element-wise **multiplicative** interactions

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

[ Zadeh, Jones and Morency, EMNLP 2017 ]
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 learning tasks, we need to join unimodal representations

- Simple **concatenation**
- Element-wise **multiplicative** interactions

Encoder-decoder Architectures
**Multimodal Representation Types**

- **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**

\[ x_1^{(i)} \quad x_2^{(i)} \]

- **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 $f_1(x_1), f_2(x_2)$ for each view that maximize correlation:

$$\text{corr}(f_1(x_1), f_2(x_2)) = \frac{\text{cov}(f_1(x_1), f_2(x_2))}{\sqrt{\text{var}(f_1(x_1)) \cdot \text{var}(f_2(x_2))}}$$

*slide from Andrew, Arora, Bilmes, Livescu*
Correlated Representations

**Goal:** Find representations $f_1(x_1), f_2(x_2)$ for each view that maximize correlation:

$$
\text{corr}(f_1(x_1), f_2(x_2)) = \frac{\text{cov}(f_1(x_1), f_2(x_2))}{\sqrt{\text{var}(f_1(x_1)) \cdot \text{var}(f_2(x_2))}}
$$

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 \( f_1(x_1), f_2(x_2) \) for each view that maximize correlation:

\[
corr(f_1(x_1), f_2(x_2)) = \frac{\text{cov}(f_1(x_1), f_2(x_2))}{\sqrt{\text{var}(f_1(x_1)) \cdot \text{var}(f_2(x_2))}}
\]

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 \textbf{linear} correlated representations, i.e.,

\[
\begin{align*}
  f_1(x_1) &= W_1^T x_1 \\
  f_2(x_2) &= W_2^T x_2
\end{align*}
\]

where

\[
W_1 \in \mathbb{R}^{d_1 \times k} \\
W_2 \in \mathbb{R}^{d_2 \times k}
\]
CCA: Canonical Correlation Analysis

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

\[ f_1(x_1) = W_1^T x_1 \]
\[ f_2(x_2) = W_2^T x_2 \]

where

\[ W_1 \in \mathbb{R}^{d_1 \times k} \]
\[ W_2 \in \mathbb{R}^{d_2 \times k} \]

The first columns \((w_{1,:1}, w_{2,:1})\) of the matrices \(W_1\) and \(W_2\) are found to maximize the **correlation of the projections**:

\[ (w_{1,:1}, w_{2,:1}) = \arg \max \text{corr}(w_{1,:1}^T X_1, w_{2,:1}^T X_2) \]

*slide from Andrew, Arora, Bilmes, Livescu*
CCA: Canonical Correlation Analysis

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

\[
\begin{align*}
f_1(x_1) &= W_1^T x_1 \\
f_2(x_2) &= W_2^T x_2
\end{align*}
\]

where \( W_1 \in \mathbb{R}^{d_1 \times k} \) and \( W_2 \in \mathbb{R}^{d_2 \times k} \)

The first columns \((w_{1,:1}, w_{2,:1})\) of the matrices \( W_1 \) and \( W_2 \) are found to maximize the correlation of the projections:

\[
(w_{1,:1}, w_{2,:1}) = \arg \max \text{corr}(w_{1,:1}^T X_1, w_{2,:1}^T X_2)
\]

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

\[
\text{corr}(w_{1,:i}^T X_1, w_{1,:j}^T X_1) = \text{corr}(w_{2,:i}^T X_2, w_{2,:j}^T X_2) = 0
\]

*slide from Andrew, Arora, Bilmes, Livescu*
CCA Illustration

\[ f_1(X_1) = w_1^T X_1 \quad \text{max corr} \quad f_2(X_2) = w_2^T X_2 \]

\[ X_1 \in \mathbb{R}^2 \quad \quad X_2 \in \mathbb{R}^2 \]

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:

\[
\Sigma_{11} = \frac{1}{N-1} \sum_{i=1}^{N} (x_1^{(i)} - \bar{x}_1)(x_1^{(i)} - \bar{x}_1)^T + r_1 I
\]

\[
\Sigma_{12} = \frac{1}{N-1} \sum_{i=1}^{N} (x_1^{(i)} - \bar{x}_1)(x_2^{(i)} - \bar{x}_2)^T
\]

\[
\Sigma_{22} = \frac{1}{N-1} \sum_{i=1}^{N} (x_2^{(i)} - \bar{x}_2)(x_2^{(i)} - \bar{x}_2)^T + r_2 I
\]

*slide from Andrew, Arora, Bilmes, Livescu
CCA: Canonical Correlation Analysis

1. Estimate \textbf{covariance matrix} with regularization:

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

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

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

\[
\Sigma = \begin{bmatrix}
\Sigma_{11} & \Sigma_{12} \\
\Sigma_{12} & \Sigma_{22}
\end{bmatrix}
\]

\[
\mathbf{W}_1^* \quad \mathbf{W}_2^*
\]

\[
\begin{bmatrix}
1 & 0 & 0 & \lambda_1 & 0 & 0 \\
0 & 1 & 0 & 0 & \lambda_2 & 0 \\
0 & 0 & 1 & 0 & 0 & \lambda_3 \\
\lambda_1 & 0 & 0 & 1 & 0 & 0 \\
0 & \lambda_2 & 0 & 0 & 1 & 0 \\
0 & 0 & \lambda_3 & 0 & 0 & 1
\end{bmatrix}
\]
CCA: Canonical Correlation Analysis

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

\[
\Sigma_{11} = \frac{1}{N-1} \sum_{i=1}^{N} (x_1^{(i)} - \bar{x}_1)(x_1^{(i)} - \bar{x}_1)^T + r_1 I
\]

\[
\Sigma_{12} = \frac{1}{N-1} \sum_{i=1}^{N} (x_1^{(i)} - \bar{x}_1)(x_2^{(i)} - \bar{x}_2)^T
\]

\[
\Sigma_{21} = \frac{1}{N-1} \sum_{i=1}^{N} (x_2^{(i)} - \bar{x}_2)(x_1^{(i)} - \bar{x}_1)^T
\]

\[
\Sigma_{22} = \frac{1}{N-1} \sum_{i=1}^{N} (x_2^{(i)} - \bar{x}_2)(x_2^{(i)} - \bar{x}_2)^T + r_2 I
\]

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

*slide from Andrew, Arora, Bilmes, Livescu*
CCA: Canonical Correlation Analysis

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

\[
\Sigma_{11} = \frac{1}{N-1} \sum_{i=1}^{N} (x_1^{(i)} - \bar{x}_1)(x_1^{(i)} - \bar{x}_1)^T + r_1 \mathbf{I}
\]

\[
\Sigma_{12} = \frac{1}{N-1} \sum_{i=1}^{N} (x_1^{(i)} - \bar{x}_1)(x_2^{(i)} - \bar{x}_2)^T
\]

\[
\Sigma_{12} = \frac{1}{N-1} \sum_{i=1}^{N} (x_2^{(i)} - \bar{x}_2)(x_2^{(i)} - \bar{x}_2)^T + r_2 \mathbf{I}
\]

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:

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

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

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

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

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*
There maybe non-linear functions $f_1(x_1), f_2(x_2)$ 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

Canonical Correlation Analysis

View 1

View 2

*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
Correlated Representations vs. Joint Embeddings

**Correlated Representations**: Find representations $f_1(x_1), f_2(x_2)$ for each view that maximize correlation:

$$\text{corr}(f_1(x_1), f_2(x_2)) = \frac{\text{cov}(f_1(x_1), f_2(x_2))}{\sqrt{\text{var}(f_1(x_1)) \cdot \text{var}(f_2(x_2))}}$$

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

$$\min_{f_1, f_2} D(f_1(x_1^{(i)}), f_2(x_2^{(i)}))$$
Joint Embeddings

Distance($s,t$)

$W_4$ $\uparrow$
$H_3$

$W_3$ $\uparrow$
$H_2$

$W_2$ $\uparrow$
$H_1$

$W_1$ $\uparrow$
Input $s$

$W_4$ $\uparrow$
$H_3$

$W_3$ $\uparrow$
$H_2$

$W_2$ $\uparrow$
$H_1$

$W_1$ $\uparrow$
Input $t1$

Image features $s$

Text: *a parrot rides a tricycle*
Joint Embeddings

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

- day + night =
- flying + sailing =
- bowl + box =
- box + bowl =

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

Problem: For each image predict which category it belongs to out of a fixed set.
**Problem:** For each image predict which category it belongs to out of a fixed set
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 $\mathbb{R}^d$. 

\[ \text{Images and class labels are embedded into the same space $\mathbb{R}^d$.} \]
**Discriminative Embeddings**

Images and class labels are embedded into the same space

\[ \Psi(I_i) = W \cdot \text{CNN}(I_i; \Theta): \mathbb{R}^D \rightarrow \mathbb{R}^d \]
**Discriminative Embeddings**

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

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

![Diagram showing the process of embedding images and class labels into the same space using a feature extractor and a CNN.](image-url)
**Discriminative Embeddings**

Images and class labels are embedded into the same space

\[ \Psi(I_i) = W \cdot CNN(I_i; \Theta) : \mathbb{R}^D \rightarrow \mathbb{R}^d \]
**Discriminative Embeddings**

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

**Image Embedding**

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

**Label Embedding**

\[ \Psi_L(\text{word}_i) = \mathbf{u}_i : \{1, \ldots, L\} \rightarrow \mathbb{R}^d \]
**Discriminative Embeddings**

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

**Image Embedding**

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

**Label Embedding**

\[
\Psi_L(\text{word}_i) = u_i : \{1, \ldots, L\} \rightarrow \mathbb{R}^d
\]

**Similarity in Embedding Space**

\[
D(u, u') = \|u - u'\|_2^2
\]
**Discriminative Embeddings**

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

**Image Embedding**

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

**Label Embedding**

\[
\Psi_L(word_i) = u_i : \{1, \ldots, L\} \rightarrow \mathbb{R}^d
\]

**Similarity in Embedding Space**

\[
D(u, u') = \frac{u}{\|u\|} \cdot \frac{u'}{\|u'\|}
\]
**Discriminative Embeddings**

**Image Embedding**

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

**Label Embedding**

$$\Psi_L(word_i) = u_i : \{1, \ldots, L\} \rightarrow \mathbb{R}^d$$

**Similarity in Embedding Space**

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

**Image Categorization / Annotation**

which object category does image belong to?
Discriminative Embeddings

**Image Embedding**
\[ \Psi(I_i) = W \cdot CNN(I_i; \Theta) : \mathbb{R}^D \rightarrow \mathbb{R}^d \]

**Label Embedding**
\[ \Psi_L(word_i) = u_i : \{1, \ldots, L\} \rightarrow \mathbb{R}^d \]

**Similarity in Embedding Space**
\[ D(u, u') = \|u - u'\|_2^2 \]

Image Categorization / Annotation
which object category does image belong to?

Distance can be interpreted as probability
**Discriminative Embeddings**

### Image Embedding

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

### Label Embedding

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

### Similarity in Embedding Space

$$D(u, u') = \|u - u'\|_2$$

**Search by Image**

most similar image to a query?

- Zebra
- Tiger
- Horse
- Lion
**Discriminative Embeddings**

**Image Embedding**

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

**Label Embedding**

\[ \Psi_L(word_i) = u_i : \{1, ..., L\} \rightarrow \mathbb{R}^d \]

**Similarity in Embedding Space**

\[ D(u, u') = \|u - u'\|_2^2 \]

---

**Search by Label**

most representative image for a label?
**Discriminative Embeddings**

**Image Embedding**

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

**Label Embedding**

\[
\Psi_L(word_i) = u_i : \{1, \ldots, L\} \rightarrow \mathbb{R}^d
\]

**Similarity in Embedding Space**

\[
D(u, u') = ||u - u'||_2^2
\]

**Objective Function:**

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

**Why not minimize distance directly?**

\[
\mathcal{L}_C(W, U, I_i, y_i) = \sum [1 + D(\Psi(I_i), u_{y_i}) - D(\Psi(I_i), u_{y_{c}})]
\]
**Discriminative Embeddings**

**Image Embedding**

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

**Label Embedding**

\[ \Psi_L(word_i) = u_i : \{1, \ldots, L\} \rightarrow \mathbb{R}^d \]

**Similarity in Embedding Space**

\[ D(u, u') = \frac{u}{||u||} \cdot \frac{u'}{||u'||} \]

**Objective Function:**

\[ \min_{W, U} \sum_{i=1}^{N} \mathcal{L}_C(W, U, I_i, y_i) + \lambda_1 ||W||_F^2 + \lambda_2 ||U||_F^2 \]

\[ \mathcal{L}_C(W, U, I_i, y_i) = \sum \max\{0, \alpha - D(\Psi(I_i), u_{y_i}) + D(\Psi(I_i), u_{y_c})\} \]
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.

As granularity of categories increases, the amount of data per category decreases.
Inspiration from Human Structured Semantics

A truck (US, CA, AU, NZ) or lorry (UK and Ireland) is a motor vehicle designed to transport cargo. Trucks vary greatly in size, power, and configuration, with the smallest being mechanically similar to an automobile. Commercial trucks can be very large and powerful, and may be configured to mount specialized equipment, such as in the case of fire trucks and concrete mixers and suction excavators. Modern trucks are largely powered by diesel engines exclusively, although small to medium size trucks with gasoline engines exist in America. In the European Union, vehicles with a gross combination mass of up to 3,500 kilograms (7,716 lb) are known as light commercial vehicles, and those over as large goods vehicles.

For legal purposes motor vehicles are often identified within a number of vehicle classes including automobiles or cars, buses, motorcycles, off highway vehicles, light trucks or light duty trucks, and trucks or lorries. These classifications vary according to the legal codes of each country. ISO 3833:1977 is the standard for road vehicle types, terms and definitions.
Inspiration from Human Structured Semantics

Parent Category + Attributes

**Truck**

- **motor vehicle designed to transport cargo**
- **self-propelled, wheeled vehicle that does not operate on rails**

[ Hwang et al., 2014 ]
Unified Semantic Embedding

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

**Image Embedding**

\[ \Psi_I(I_i) = W \cdot \text{CNN}(I_i) : \mathbb{R}^D \to \mathbb{R}^d \]

**Label Embedding**

\[ \Psi_L(word_i) = u_i : \{1, ..., L\} \to \mathbb{R}^d \]

**Similarity in Embedding Space**

\[ D(u, u') = ||u - u'||_2^2 \]

**Objective Function:**

\[
\min_{W, U, B} \sum_{i=1}^{N} L_C(W, U, I_i, y_i) + \lambda \sum \|W_i\|_2^2 + L_A(W, U, I_i, y_i) + \mathcal{R}(U, B)
\]

Each sample is closer to the parent category than to a sibling category.
Unified Semantic Embedding

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

**Image Embedding**

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

**Label Embedding**

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

**Similarity in Embedding Space**

$$D(u, u') = ||u - u'||_2^2$$

**Objective Function:**

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

$$\mathcal{L}_S(W, U, x_i, y_i) = \sum_{s \in P_{y_i}} \sum_{c \in S_s} [1 + ||W x_i - u_s||_2^2 - ||W x_i - u_c||_2^2]$$

**Taxonomies:**

- big cat
- tiger
- lion

![Diagram showing the relationship between image embedding and label embedding in the context of semantic similarity and regularization.](Image)
Unified Semantic Embedding

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

**Image Embedding**

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

**Label Embedding**

\[ \Psi_L(word_i) = u_i : \{1, \ldots, L\} \rightarrow \mathbb{R}^d \]

**Attribute Embedding**

\[ \Psi_A(attr_i) = a_i : \{1, \ldots, A\} \rightarrow \mathbb{R}^d, s.t. \|a_i\|^2 \leq 1 \]

**Similarity in Embedding Space**

\[ D(u, u') = \|u - u'\|_2^2 \]

**Objective Function:**

\[
\min_{W, U, B} \sum_{i=1}^{N} \left( \mathcal{L}_C(W, U, I_i, y_i) + \mathcal{L}_S(W, U, I_i, y_i) + \mathcal{L}_A(W, U, I_i, y_i) + \mathcal{R}(U, B) + \lambda_1\|W\|_F^2 + \lambda_2\|U\|_F^2 \right)
\]
Unified Semantic Embedding

**Image Embedding**  
\[ \Psi_I(I_i) = W \cdot CNN(I_i) : \mathbb{R}^D \rightarrow \mathbb{R}^d \]

**Label Embedding**  
\[ \Psi_L(\text{word}_i) = u_i : \{1, \ldots, L\} \rightarrow \mathbb{R}^d \]

**Attribute Embedding**  
\[ \Psi_A(\text{attr}_i) = a_i : \{1, \ldots, A\} \rightarrow \mathbb{R}^d, \text{s.t.} \|a_i\|^2 \leq 1 \]

**Similarity in Embedding Space**  
\[ D(u, u') = \|u - u'|^2_2 \]

**Objective Function:**  
\[ \min_{W, U, B} \sum_i^N \mathcal{L}_C(W, U, I_i, y_i) + \mathcal{L}_S(W, U, I_i, y_i) + \mathcal{L}_A(W, U, I_i, y_i) + \mathcal{R}(U, B) + \lambda_1 \|W\|^2_F + \lambda_2 \|U\|^2_F \]

\[ \mathcal{R}(U, B) = \sum_c \|u_c - u_p - U^A \beta_c\|^2 + \gamma_2 \|\beta_c + \beta_o\|^2 \]

Each category is a parent + sparse subset of attribute bases
Unified Semantic Embedding

**Image Embedding**

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

**Label Embedding**

\[ \Psi_L(word_i) = u_i : \{1, ..., L\} \rightarrow \mathbb{R}^d \]

**Attribute Embedding**

\[ \Psi_A(attr_i) = a_i : \{1, ..., A\} \rightarrow \mathbb{R}^d, \text{s.t. } ||a_i||^2 \leq 1 \]

**Similarity in Embedding Space**

\[ D(u, u') = ||u - u'||^2_2 \]

**Objective Function:**

\[
\min_{W, U, B} \sum_{i=1}^{N} \mathcal{L}_C(W, U, I_i, y_i) + \mathcal{L}_S(W, U, I_i, y_i) + \mathcal{L}_A(W, U, I_i, y_i) + R(U, B) + \lambda_1 ||W||^2_F + \lambda_2 ||U||^2_F
\]

[ Hwang et al., 2014 ]
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)

- Animal
  - Odd-toed ungulate
  - Primate
    - hands
    - bipedal
  - Equine
    - ungulate
    - lean
    - active

- Musteline Mammal
- Deer
- Skunk
- Otter
- Deer
- Moose

- Otter: quadrupedal, flippers, furry, ocean
- Skunk: stripes
- Deer: spots, nests, longneck, yellow, hooves
- Moose: arctic, stripes, black

[ Hwang et al., 2014 ]

An animal that has horns, brown, big, quadrupedal, new

Furry, black, brown, tail, . . .

An animal that swims, fish, water, new world, small, flippers,

Ground-truth attributes

Explicit embedding of semantic entities using our method improved both the top-1 accuracy and

Semantics

Static

Implicit

Semantics

We did extensive parameter search for the ALE variants.

Note that some attributes that are ranked less relevant by humans were selected for their dis-

For example, our method select attributes such as

Incorporate semantics

For skunk, instead of attributes common and nondiscriminative such as

Explicit embedding of semantic entities using our method improved both the top-1 accuracy and

Hierarchical precision, with USE variants achieving the best performance in both. Specifically,

To show the effectiveness of using superclass+attributes in the description, we report the learned

Corrected attributes are colored in red.

Explicit embedding of semantic entities using our method improved both the top-1 accuracy and

Incorporate semantics

For skunk, instead of attributes common and nondiscriminative such as

Incorrect attributes are colored in red.
Experiments
Results with AWA (with latent attributes)

Model **benefits:**
- highly interpretable
- efficient in learning
Experiments
Results with AWA (with latent attributes)

Model benefits:

- highly interpretable
- efficient in learning

alternative attribute-based representations
Experiments

Results with AWA (with latent attributes)

<table>
<thead>
<tr>
<th>Method</th>
<th>Flat hit @ k (%)</th>
<th>Hierarchical precision @ k (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>No semantics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ridge Regression</td>
<td>38.39 ± 1.48</td>
<td>48.61 ± 1.29</td>
</tr>
<tr>
<td>NCM [1]</td>
<td>43.49 ± 1.23</td>
<td>57.45 ± 0.91</td>
</tr>
<tr>
<td>LME</td>
<td>44.76 ± 1.77</td>
<td>58.08 ± 2.05</td>
</tr>
<tr>
<td>Implicit semantics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LMTE [2]</td>
<td>38.92 ± 1.12</td>
<td>49.97 ± 1.16</td>
</tr>
<tr>
<td>ALE [3]</td>
<td>36.40 ± 1.03</td>
<td>50.43 ± 1.92</td>
</tr>
<tr>
<td>HLE [3]</td>
<td>33.56 ± 1.64</td>
<td>45.93 ± 2.56</td>
</tr>
<tr>
<td>AHLE [3]</td>
<td>38.01 ± 1.69</td>
<td>52.07 ± 1.19</td>
</tr>
<tr>
<td>Explicit semantics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LME-MTL-S</td>
<td>45.03 ± 1.32</td>
<td>57.73 ± 1.75</td>
</tr>
<tr>
<td>LME-MTL-A</td>
<td>45.55 ± 1.71</td>
<td>58.60 ± 1.76</td>
</tr>
<tr>
<td>USE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>USE-No Reg.</td>
<td>45.93 ± 1.76</td>
<td>59.37 ± 1.32</td>
</tr>
<tr>
<td>USE-Reg.</td>
<td><strong>46.42 ± 1.33</strong></td>
<td><strong>59.54 ± 0.73</strong></td>
</tr>
</tbody>
</table>

Variants of our Unified Semantic Embedding (USE) model:

- Ontology
- Attributes
- Parent + Sparse Attributes

References:

### Results with AWA (with latent attributes)

<table>
<thead>
<tr>
<th>Method</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ridge Regression</td>
<td>38.93</td>
</tr>
<tr>
<td>NCM [1]</td>
<td></td>
</tr>
<tr>
<td>LME</td>
<td></td>
</tr>
<tr>
<td>LMTE [2]</td>
<td></td>
</tr>
<tr>
<td>ALE [3]</td>
<td></td>
</tr>
<tr>
<td>HLE [3]</td>
<td></td>
</tr>
<tr>
<td>AHLE [3]</td>
<td></td>
</tr>
<tr>
<td>LME-MTL-S</td>
<td></td>
</tr>
<tr>
<td>USE-No Reg.</td>
<td>44.87+5.9%</td>
</tr>
<tr>
<td>USE-Reg.</td>
<td>49.87+5.0%</td>
</tr>
</tbody>
</table>

**Variants of our Unified Semantic Embedding (USE) model:**
- Ontology
- Attributes
- Parent + Sparse Attributes

---

Semantic Embeddings

Image Embedding

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

Label Embedding

$$\Psi_L(word_i) = u_i : \{1, ..., L\} \rightarrow \mathbb{R}^d$$
**word2vec:** Unsupervised Word Embedding

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

\[
\Psi_L(word_i) = u_i : \{1, ..., L\} \rightarrow \mathbb{R}^d
\]
**word2vec**: Unsupervised Word Embedding

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

\[
\Psi_L(\text{word}_i) = \mathbf{u}_i : \{1, ..., L\} \rightarrow \mathbb{R}^d
\]

**Label Embedding**

- 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

\[ \text{loss(image, label)} = \sum_{j \neq \text{label}} \max[0, \text{margin} - \tilde{t}_{\text{label}} M \tilde{v} \text{(image)} + \tilde{t}_{j} M \tilde{v} \text{(image)}] \]
DeViSE: A Deep Visual-Semantic Embedding Model

[Frome et al., 2013]

**Supervised Results**

<table>
<thead>
<tr>
<th>Model type</th>
<th>dim</th>
<th>Flat hit@k (%)</th>
<th>Hierarchical precision@k</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Softmax baseline</td>
<td>N/A</td>
<td>55.6</td>
<td>67.4</td>
</tr>
<tr>
<td>DeVISE</td>
<td>500</td>
<td>53.2</td>
<td>65.2</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>54.9</td>
<td>66.9</td>
</tr>
<tr>
<td>Random embeddings</td>
<td>500</td>
<td>52.4</td>
<td>63.9</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>50.5</td>
<td>62.2</td>
</tr>
<tr>
<td>Chance</td>
<td>N/A</td>
<td>0.1</td>
<td>0.2</td>
</tr>
</tbody>
</table>

**Zero-shot Results**

<table>
<thead>
<tr>
<th>Model</th>
<th>200 labels</th>
<th>1000 labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeVISE</td>
<td>31.8%</td>
<td>9.0%</td>
</tr>
<tr>
<td>Mensink et al. 2012 [12]</td>
<td>35.7%</td>
<td>1.9%</td>
</tr>
<tr>
<td>Rohrbach et al. 2011 [17]</td>
<td>34.8%</td>
<td>-</td>
</tr>
</tbody>
</table>
Semantic Embeddings

Image Embedding

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

Label Embedding

\[ \Psi_L(\text{word}_i) = \mathbf{u}_i : \{1, \ldots, L\} \rightarrow \mathbb{R}^d \]
**word2vec:** Unsupervised Word Embedding

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

\[ \Psi_L(\text{word}_i) = u_i : \{1, \ldots, L\} \rightarrow \mathbb{R}^d \]

\[ L = 310,000 \]

**Label Embedding**

\[ u_{\text{horse}} \]
\[ u_{\text{breeds}} \]
\[ u_{\text{are}} \]
\[ u_{\text{loosely}} \]

**Skip-gram Model:** unsupervised semantic representation for words
(trained from 7 billion word linguistic corpus)

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

[Fu et al., 2016]
Semi-supervised **Vocabulary Informed** Learning

Image Embedding

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

Label Embedding

\[ \Psi_L(\text{word}_i) = \mathbf{u}_i : \{1, \ldots, L\} \rightarrow \mathbb{R}^d \]

\[ L = 310,000 \]
Semi-supervised **Vocabulary Informed** Learning

[ Fu et al., 2016 ]

**Image Embedding**

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

**Label Embedding**

\[ \Psi_L(word_i) = u_i : \{1, \ldots, L\} \rightarrow \mathbb{R}^d \]

\[ L = 310,000 \]

**Similarity in Embedding Space**

\[ D(u, u') = \|u - u'\|_2^2 \]
Semi-supervised **Vocabulary Informed** Learning [Fu et al., 2016]

### Image Embedding

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

### Label Embedding

$$\Psi_L(\text{word}_i) = u_i : \{1, \ldots, L\} \to \mathbb{R}^d$$

$L = 310,000$

### Similarity in Embedding Space

$$D(u, u') = ||u - u'||_2^2$$

### Objective Function:

$$\min_W \sum_i^N L_C(W, V, I_i, y_i) + L_R(W, V, I_i, y_i) + \mu ||V||_F^2,$$
Semi-supervised **Vocabulary Informed** Learning [Fu et al., 2016]

**Image Embedding**

\[ \Psi(I_i) = W \cdot CNN(I_i; \Theta) : \mathbb{R}^D \to \mathbb{R}^d \]

**Label Embedding**

\[ \Psi_L(word_i) = u_i : \{1, ..., L\} \to \mathbb{R}^d \]

**Similarity in Embedding Space**

\[ D(u, u') = ||u - u'||^2_2 \]

**Objective Function:**

\[ \min_W \sum_i^N \mathcal{L}_C(W, V, I_i, y_i) + \mathcal{L}_R(W, V, I_i, y_i) + \mu ||V||^2_F \]

\[ \mathcal{L}_C(W, U, x_i, y_i) = \sum [1 + D(Wx_i, u_{y_i}) - D(Wx_i, u_c)] \]
Semi-supervised **Vocabulary Informed** Learning  

**Image Embedding**

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

**Label Embedding**

\[ \Psi_L(\text{word}_i) = u_i : \{1, ..., L\} \rightarrow \mathbb{R}^d \]

**Similarity in Embedding Space**

\[ D(u, u') = ||u - u'||_2^2 \]

**Objective Function:**

\[ \min_W \sum_i^N \mathcal{L}_C(W, V, I_i, y_i) + \mathcal{L}_R(W, V, I_i, y_i) + \mu ||V||_F^2 \]

\[ \mathcal{L}_C(W, U, x_i, y_i) = \sum [1 + D(Wx_i, u_{y_i}) - D(Wx_i, u_c)] \]

[Fu et al., 2016]
Vocabulary Informed Recognition

[ Fu et al., 2016 ]
Experiments: Datasets

Animals with Attributes

Otter

Polar Bear

...  

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

<table>
<thead>
<tr>
<th>AwA/ImageNet</th>
<th>No. Testing Classes</th>
<th>No. Testing Words</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Auxiliary</td>
<td>Target</td>
</tr>
<tr>
<td>SUPERVISED</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ZERO-SHOT</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>OPEN-SET</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

The tasks are only separated in **evaluation**; we train **one unified model** for all the settings.
Experiments: Settings

Training

Otter
Polar Bear
Donkey
Poney
Panther

310,000

Testing

Supervised

Zero-shot

[ Fu et al., 2016 ]
Conventional object detectors for all these would require millions of training images for each class to be learned. However, with modern machine learning techniques, powerful low-level features enable and efficient detectors are available, based on the combination of powerful low-level features and abstract well from examples, they are also capable of transcending the specific learning task at hand, they can be pre-learned, and when looking for “large gray animals with long trunks”, we will reliably identify them. For “red traffic sign with white writing”, we will be able to detect them image, collections have been formed and annotated on this paradigm and propose a system that is able to detect object classes, for which it had not seen a single training example. This setup has hardly been studied in computer vision research, but it is the rule rather than the exception, because the world contains tens of thousands of different object classes and for only a very few of them image, collections have been formed and annotated. It has been estimated that humans distinguish between over 30,000 animal classes. But, instead of creating a separate training classifier for each of them would be as tedious as for all of them. From image datasets unrelated to the current task. Afterwards, new classes can be detected based on their attribute representation, without the need for a new training phase. In order to evaluate our method and to facilitate research in this area, we have assembled a new large-scale dataset, consisting of arbitrary semantic attributes, like shape, color, or even geographic information. Because such properties are specific to object classes, in particular faces and vehicles, reliable images have made large progress over the last years. For example, learning object detection system that does not require any training images of the target classes. By using an attribute layer it is indeed possible to build a pre-learned, attribute-based classification. Experiments: From the phrase “eight-sided animal images that match the 50 classes in Osherson’s classical table of how strongly humans associate 85 semantic attributes with animal classes. Our experiments show that attributes can be used to perform object detection when provided with a high-level description. Therefore, numerous techniques have been developed, some of which we will discuss in Section 3.
### Experiments: Settings

<table>
<thead>
<tr>
<th>AwA/ImageNet</th>
<th>No. Testing Classes</th>
<th>No. Testing Words</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Auxiliary</td>
<td>Target</td>
</tr>
<tr>
<td>SUPERVISED</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>ZERO-SHOT</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>OPEN-SET</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

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

[ Fu et al., 2016 ]
<table>
<thead>
<tr>
<th>Method</th>
<th>Features</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS-Voc: full instances</td>
<td>CNN\textsubscript{OverFeat}</td>
<td>78.3</td>
</tr>
<tr>
<td>Akata et al. CVPR 2015</td>
<td>CNN\textsubscript{GoogLeNet}</td>
<td>73.9</td>
</tr>
<tr>
<td>TMV-BLP (Fu et al. ECCV 2014)</td>
<td>CNN\textsubscript{OverFeat}</td>
<td>69.9</td>
</tr>
<tr>
<td>AMP (SR+SE) (Fu et al. CVPR 2015)</td>
<td>CNN\textsubscript{OverFeat}</td>
<td>66.0</td>
</tr>
<tr>
<td>DAP (Lampert et al. TPAMI 2013)</td>
<td>CNN\textsubscript{VGG19}</td>
<td>57.5</td>
</tr>
<tr>
<td>PST (Rohrbach et al. NIPS 2013)</td>
<td>CNN\textsubscript{OverFeat}</td>
<td>53.2</td>
</tr>
<tr>
<td>DS (Rohrbach et al. CVPR 2010)</td>
<td>CNN\textsubscript{OverFeat}</td>
<td>52.7</td>
</tr>
<tr>
<td>IAP (Lampert et al. TPAMI 2013)</td>
<td>CNN\textsubscript{OverFeat}</td>
<td>44.5</td>
</tr>
<tr>
<td>HEX (Deng et al. ECCV 2014)</td>
<td>CNN\textsubscript{DECAF}</td>
<td>44.2</td>
</tr>
<tr>
<td>Method</td>
<td>Features</td>
<td>Accuracy</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>----------------</td>
<td>----------</td>
</tr>
<tr>
<td>SS-Voc: full instances</td>
<td>CNNOverFeat</td>
<td>78.3</td>
</tr>
<tr>
<td>800 instances (20 inst*40 class);</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Akata et al. CVPR 2015</td>
<td>CNNGoogLeNet</td>
<td>73.9</td>
</tr>
<tr>
<td>TMV-BLP (Fu et al. ECCV 2014)</td>
<td>CNNOverFeat</td>
<td>69.9</td>
</tr>
<tr>
<td>AMP (SR+SE) (Fu et al. CVPR 2015)</td>
<td>CNNOverFeat</td>
<td>66.0</td>
</tr>
<tr>
<td>DAP (Lampert et al. TPAMI 2013)</td>
<td>CNNVGG19</td>
<td>57.5</td>
</tr>
<tr>
<td>PST (Rohrbach et al. NIPS 2013)</td>
<td>CNNOverFeat</td>
<td>53.2</td>
</tr>
<tr>
<td>DS (Rohrbach et al. CVPR 2010)</td>
<td>CNNOverFeat</td>
<td>52.7</td>
</tr>
<tr>
<td>IAP (Lampert et al. TPAMI 2013)</td>
<td>CNNOverFeat</td>
<td>44.5</td>
</tr>
<tr>
<td>HEX (Deng et al. ECCV 2014)</td>
<td>CNNDECAF</td>
<td>44.2</td>
</tr>
</tbody>
</table>

Note: 3.3% of training data [Fu et al., 2016]
## Zero-shot Results

### Results with AWA

<table>
<thead>
<tr>
<th>Method</th>
<th>Features</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS-Voc: full instances</td>
<td>CNN_{OverFeat}</td>
<td>78.3</td>
</tr>
<tr>
<td>800 instances (20 inst*40 class)</td>
<td>CNN_{OverFeat}</td>
<td>74.4</td>
</tr>
<tr>
<td>200 instances (5 inst*40 class)</td>
<td>CNN_{OverFeat}</td>
<td>68.9</td>
</tr>
<tr>
<td>Akata et al. CVPR 2015</td>
<td>CNN_{GoogLeNet}</td>
<td>73.9</td>
</tr>
<tr>
<td>TMV-BLP (Fu et al. ECCV 2014)</td>
<td>CNN_{OverFeat}</td>
<td>69.9</td>
</tr>
<tr>
<td>AMP (SR+SE) (Fu et al. CVPR 2015)</td>
<td>CNN_{OverFeat}</td>
<td>66.0</td>
</tr>
<tr>
<td>DAP (Lampert et al. TPAMI 2013)</td>
<td>CNN_{VGG19}</td>
<td>57.5</td>
</tr>
<tr>
<td>PST (Rohrbach et al. NIPS 2013)</td>
<td>CNN_{OverFeat}</td>
<td>53.2</td>
</tr>
<tr>
<td>DS (Rohrbach et al. CVPR 2010)</td>
<td>CNN_{OverFeat}</td>
<td>52.7</td>
</tr>
<tr>
<td>IAP (Lampert et al. TPAMI 2013)</td>
<td>CNN_{OverFeat}</td>
<td>44.5</td>
</tr>
<tr>
<td>HEX (Deng et al. ECCV 2014)</td>
<td>CNN_{DECAF}</td>
<td>44.2</td>
</tr>
</tbody>
</table>

0.82% of training data

[Fu et al., 2016]
Weakly-supervised **Visual Grounding** of Phrases

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

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

[ Xiao et al., 2017 ]
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  

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.
- A table

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

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

[ Xiao et al., 2017 ]

Label Embedding

\[ \Psi_L(\text{phrase}_i) = u_i \]
Weakly-supervised **Visual Grounding** of Phrases

[ Xiao et al., 2017 ]

\[ \Psi_L(\text{phrase}_i) = u_i \]
Weakly-supervised **Visual Grounding** of Phrases

\[ \Psi_L(\text{phrase}_i) = \mathbf{u}_i \]
Weakly-supervised Visual Grounding of Phrases

$$\Psi_L(\text{phrase}_i) = u_i$$

Language Encoder

LSTM → LSTM → LSTM → LSTM → LSTM

a → man → that → is → cutting → sandwich

Label Embedding

purple bus

a table

a man
Weakly-supervised **Visual Grounding** of Phrases

[ Xiao et al., 2017 ]

**Image Embedding**

\[ \Psi(I_i) = W \cdot CNN(I_i; \Theta) \]

**Label Embedding**

\[ \Psi_L(\text{phrase}_i) = u_i \]
Weakly-supervised **Visual Grounding** of Phrases

\[ \Psi(I_i) = W \cdot CNN(I_i; \Theta) \]

\[ \Psi_L(\text{phrase}_i) = u_i \]

**Image Embedding**

**Label Embedding**

**Language Encoder**

\[ \text{a man that is cutting sandwich} \]

**Visual Attributes by MIL**

- **Attributes:**
  - bananas: 1
  - market: 0.99
  - table: 0.51
  - people: 0.43

**Visual representation by DCNN**

- **Attributes:**
  - dog: 0.95
  - frisbee: 0.83
  - outdoor: 0.82
  - grass: 0.81
  - leap: 0.45

**Latent Attention**

16x16

**Feature Extractor**

**CNN**

purple bus

a table

a man

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

**Image Embedding**

\[ \Psi(I_i) = W \cdot CNN(I_i; \Theta) \]

**Label Embedding**

\[ \Psi_L(\text{phrase}_i) = u_i \]

**Language Encoder**

- LSTM → LSTM → LSTM → LSTM → LSTM → LSTM
- a → man → that → is → cutting → sandwich

**Latent Attention**

16x16

- purple bus
- a table
- a man

**Visual Attributes by MIL**

- Visual representation by DCNN

- Attributes:
  - [bananas: 1]
  - [market: 0.99]
  - [table: 0.51]
  - [people: 0.43]
Weakly-supervised **Visual Grounding** of Phrases [Xiao et al., 2017]

**Image Embedding**

\[
\Psi(I_i) = W \cdot CNN(I_i; \Theta)
\]

**Label Embedding**

\[
\Psi_L(phrase_i) = u_i
\]

**Similarity in Embedding Space**

\[
D(u, u') = \|u - u'\|^2_2
\]

**Objective Function:**

Combination of previous discriminative similarity and **linguistic regularization**
Weakening-supervised **Visual Grounding** of Phrases

**For noun phrases:**
- **siblings** should have **disjoint** masks
- **parents** should be **union of children** masks

**Image Embedding**
\[ \Psi(I_i) = W \cdot CNN(I_i; \Theta) \]

**Label Embedding**
\[ \Psi_L(\text{phrase}_i) = u_i \]

**Similarity in Embedding Space**
\[ D(u, u') = \|u - u'\|_2^2 \]

**Objective Function:**
Combination of previous discriminative similarity and **linguistic regularization**
Qualitative Results

Input:
guy in green t-shirt holding skateboard

NO linguistic constraints

Our Model

[ Xiao et al., 2017 ]
Qualitative Results

Input:

a person driving a boat

NO linguistic constraints

[ Xiao et al., 2017 ]
Qualitative **Results**

**Input:**

a child wearing black protective helmet

**NO** linguistic constraints

[ Xiao et al., 2017 ]
## Quantitative Results

Segmentation performance on COCO dataset

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

<table>
<thead>
<tr>
<th></th>
<th>IoU@0.3</th>
<th>IoU@0.4</th>
<th>IoU@0.5</th>
<th>Avg mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-structured</td>
<td>0.302</td>
<td>0.199</td>
<td>0.110</td>
<td>0.203</td>
</tr>
<tr>
<td>Parent-Child</td>
<td>0.327</td>
<td>0.213</td>
<td>0.118</td>
<td>0.219</td>
</tr>
<tr>
<td>Sibling</td>
<td>0.316</td>
<td>0.203</td>
<td>0.114</td>
<td>0.211</td>
</tr>
<tr>
<td>Ours</td>
<td>0.347</td>
<td>0.246</td>
<td>0.159</td>
<td>0.251</td>
</tr>
</tbody>
</table>

[Xiao et al., 2017]
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*