Research and Implementation of Multi-Dataset Training for Image Classification with Discrepant Taxonomies

Master Thesis Presentation

Björn Buschkämper

Technical Faculty, Bielefeld University

September 19, 2025

Outline

The Challenge: Limited Scope of Traditional Models

- Traditional image classification models are trained on specific datasets
- Each model recognizes only a predefined set of categories
- Multiple models needed for different domains = inefficient storage and deployment

Current Approaches:

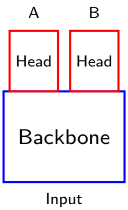
- **Transfer Learning**: Adapt pre-trained models to new tasks
- **Multi-head Architecture**: Shared backbone + task-specific heads

Problem: Still requires separate models or heads for each domain

Our Idea: Universal Model vs. Multi-Head

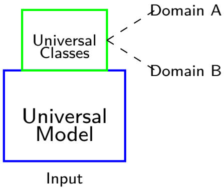
Multi-Head Approach

- Shared backbone
- Task-specific heads
- Automatic feature distillation
- Domain alignment challenges



Universal Model Approach

- Single shared model
- Universal output layer
- Predefined concept mapping
- Static domain conversion



Our task: Discover inter-dataset class relationships to build a universal taxonomy

Our Method in 3 Steps

1. **Cross-Domain Prediction**: Train domain-specific models, run inference on other domains
2. **Relationship Building**: Extract meaningful relationships from cross-domain predictions and create universal taxonomy
3. **Universal Model Training**: Create and train a single model using universal taxonomy

Changes to Method of Bevandic et al.

1. **Weighted Relationships:** Instead of binary, unweighted relationships, we use weighted relationships to capture the strength of associations between classes.
2. **Relationship Selection Methods:** We try multiple methods to select the most relevant relationships from noisy cross-domain predictions.
3. **Discrete Probability Loss Function:** We use a loss function that allows training with probability distributions as targets, enabling the model to learn from the uncertainty in class relationships.
4. **Adapted to Image Classification:** The original method was designed for image segmentation, we adapt it for image classification tasks.

Cross-Domain Prediction Process

Goal: Discover relationships between classes from different datasets

Process:

1. Train domain-specific models
2. Run each model on images from *all other* domains, building prediction matrices $M_{ab}(i,j)$ = number of times class c_i^a predicted as class c_j^b
3. Create probability matrices $P_{ab}(i,j)$ = probability of classifying class c_i^a as class c_j^b

$$P_{ab}(i,j) = \frac{M_{ab}(i,j)}{\sum_{k=1}^{|C_a|} M_{ab}(i,k)} \tag{1}$$

Example: Caltech-256 class "car" predicated by CIFAR-100 model as "vehicle" (80%), "bike" (18%), "butterfly" (2%)

Challenge: Selecting Relevant Relationships

Problems with raw probability matrices:

- Noisy predictions from imperfect models
- Unknown number of true relationships
- Different datasets have different scales of similarity

Solution: Develop multiple relationship selection methods and compare their effectiveness

Relationship Selection Methods Explained

1. **Most Common Foreign Prediction (MCFP) by Bevandic et al.**:

$$\text{select\_relationships}(P_{ab}) = \{(i,j) \mid j = \text{argmax}_{j'} P_{ab}(i,j')\} \tag{2}$$

2. **Naive Thresholding**:

$$\text{select\_relationships}(P_{ab}) = \{(i,j) \mid P_{ab}(i,j) \geq t\} \tag{3}$$

3. **Density Thresholding**: Select minimum relationships covering $p\%$ of probability mass
4. **Relationship Hypothesis**: Assumes relationships based on shared concepts should have equal probabilities. For each class, find optimal k relationships by minimizing:

$$\sum_{j=1}^{k} \left| X_i(j) - \frac{1}{k} \right| + \sum_{j=k+1}^{|C_b|} X_i(j) \tag{4}$$

where $X_i(j)$ are sorted probabilities in descending order.

Universal Taxonomy Building Rules

How do we convert relationship graphs into universal taxonomies?

We have developed four key rules to systematically transform cross-domain relationship graphs into universal taxonomies:

1. **Isolated Node Rule**: Handle classes with no relationships
2. **Bidirectional Relationship Rule**: Handle mutual relationships
3. **Transitive Cycle Rule**: Prevent invalid cycles
4. **Unilateral Relationship Rule**: Handle subset relationships

Let's examine each rule in detail...

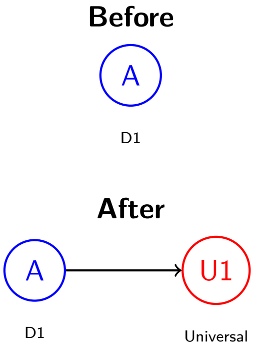
Rule 1: Isolated Node Rule

Problem: Classes with no relationships to other domains

Solution:

- Create new universal class for each standalone domain class
- Ensures all classes are represented in universal taxonomy

Example: A specialized class like "Joshua tree" that has no equivalent in other datasets gets its own universal class.



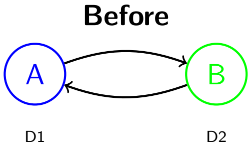
Rule 2: Bidirectional Relationship Rule

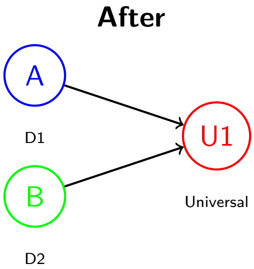
Problem: Classes with mutual relationships (A $\leftrightarrow$ B)

Solution:

- Create single universal class C
- Map both domain classes to this universal class: A $\rightarrow$ C, B $\rightarrow$ C
- Indicates classes share a common concept

Example: "car" from Caltech-256 and "automobile" from Caltech-101 have bidirectional relationships, so they merge into a single universal class.





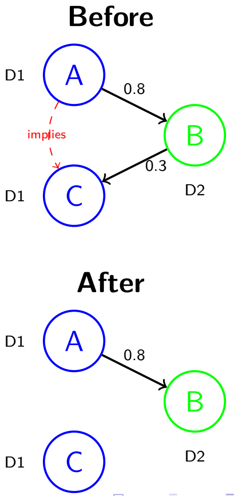
Rule 3: Transitive Cycle Rule

Problem: Invalid cycles (A $\rightarrow$ B $\rightarrow$ C where A, C are from same domain)

Solution:

- Classes within same domain should be disjoint
- Cycle hints at subset between A and C
- Remove weaker link in cycle

Example: If "truck" from Caltech-256 maps to "automobile" from Caltech-101 which maps to "car" from Caltech-256, we remove the weaker link to prevent the invalid implication that "truck" and "car" have a relationship.



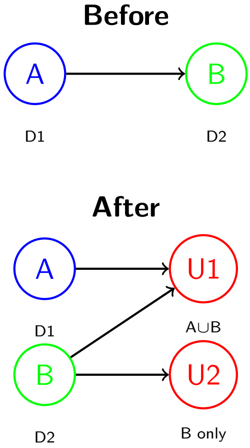
Rule 4: Unilateral Relationship Rule

Problem: Handle subset relationships (A $\rightarrow$ B)

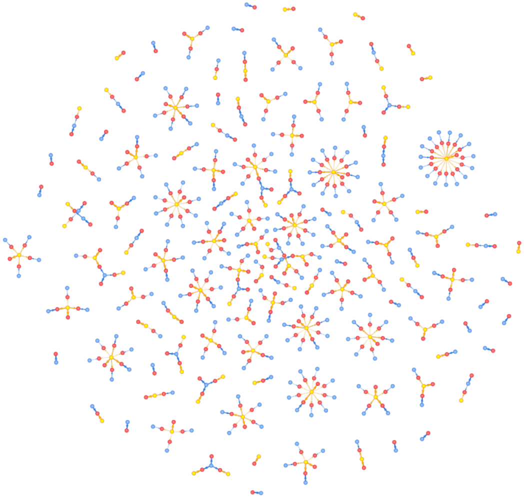
Solution:

- Create universal class for shared concepts (A $\cup$ B)
- Create universal class for unique concepts (B only)
- Preserves both general and specific knowledge
- Handles hierarchical relationships

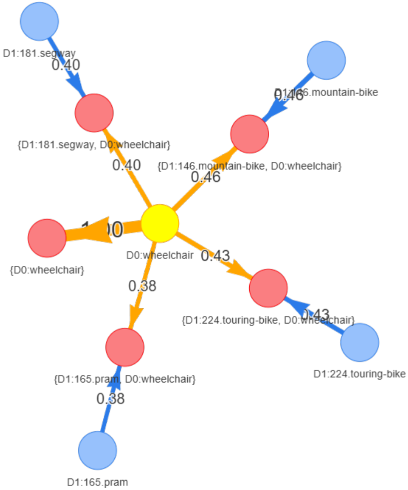
Example: "dog" from Caltech-101 relates to "animal" from CIFAR-100. We create both a "dog" universal class and an "animal (non-dog)" class.



Taxonomy Visualization (Caltech-101 + Caltech-256)



Good Cluster



Bad Cluster

D1:194.socks
D1:223.top-hat
D1:243.welding-mask
D1:226.traffic-light
D1:141.microscope
D1:003.backpack
D1:012.binoculars
D1:161.photocopier
D1:183.sextant
D1:219.theodolite
D1:019.boxing-glove
D1:070.fire-extinguisher
D1:215.telephone-box
D1:120.joy-stick
D1:176.saddle
D0:binocular
{D1:194.socks, D0:binocular}
{D1:223.top-hat, D0:binocular}
{D1:243.welding-mask, D0:binocular}
{D1:226.traffic-light, D0:binocular}
{D0:binocular, D1:141.microscope}
{D1:003.backpack, D0:binocular}
{D0:binocular}
{D1:012.binoculars, D0:binocular}
{D1:161.photocopier, D0:binocular}
{D1:183.sextant, D0:binocular}
{D1:219.theodolite, D0:binocular}
{D1:019.boxing-glove, D0:binocular}
{D1:070.fire-extinguisher, D0:binocular}
{D1:176.saddle, D0:binocular}

Building the Universal Model

- Every universal class corresponds to one output neuron
- Each domain class maps to one or more universal classes
- Matrix M_i maps domain i classes to universal classes

Target Generation: Convert domain labels to universal class distributions

$$\mathbf{t} = \hat{M}_i[j,:] \quad \text{where } \hat{M}_i(j,u) = \frac{M_i(j,u)}{\sum_{u'} M_i(j,u')} \tag{5}$$

Loss Function:

$$\mathcal{L} = -\sum_{u=1}^{|U|} \mathbf{t}(u)\log(\mathbf{p}(u)) \tag{6}$$

Multi-Domain Training Process

Training Procedure:

1. Concatenate domain datasets
2. Each sample: $(\text{image}, (\text{domain\_id}, \text{label})) \rightarrow (\text{image}, \text{universal\_target})$
3. Train universal model on unified dataset

Inference:

$$\mathbf{d}_i = M_i^T \mathbf{p} \tag{7}$$

$$\hat{c}_i = \text{argmax}(\mathbf{d}_i) \tag{8}$$

where $\mathbf{p}$ are universal class predictions and $\hat{c}_i$ is the predicted class in domain i.

Universal Model Performance

Datasets: Caltech-101, Caltech-256, CIFAR-100

Cal-101 + Cal-256 Results

Taxonomy	Cal-101	Cal-256
Hypothesis	91.81 (+0.00)	82.84 (+13.36)
MCFP	91.23 (-0.58)	80.75 (+11.27)
MCFP Binary	92.73 (+0.92)	**89.71 (+20.23)**
Density	92.96 (+1.15)	81.54 (+12.06)
Naive	**93.19 (+1.38)**	82.25 (+12.77)

Cal-101 + Cal-256 + CIFAR Results

Taxonomy	Cal-101	Cal-256	CIFAR
Hypothesis	68.74 (-23.07)	58.17 (-11.31)	69.03 (+8.55)
MCFP	83.28 (-8.53)	76.50 (+7.02)	76.10 (+15.62)
MCFP Binary	94.58 (+2.77)	85.13 (+15.65)	82.71 (+22.23)
Density	95.39 (+3.58)	83.53 (+14.05)	**83.14 (+22.66)**
Naive	**95.50 (+3.69)**	**85.36 (+15.88)**	72.56 (+12.08)

Baselines: Cal-101: 91.81%, Cal-256: 69.48%, CIFAR-100: 60.48%

Key Findings:

- Universal models **outperform** single-domain baselines
- Different relationship selection methods excel on different datasets

Key Contributions

1. **Novel Weighted Graph Approach**: We have a *weighted* relationship graph instead of binary edges
2. **Comprehensive Relationship Selection Methods**: Four different methods to select relevant relationships from noisy predictions
3. **Reusable Taxonomy Framework**: Our code is adaptable to new taxonomy building rules, relationship selection methods, datasets and model architectures
4. **Full-automatic, multi-dataset training pipeline**: We provide a complete pipeline for training on multiple datasets without manual intervention or preprocessing

Future Work

Findings:

- Bad clusters still exist in universal taxonomies
- Multi-domain universal model training outperforms single-domain training
- No single relationship selection method is best for all cases

Future Work:

- Adaptive or hybrid relationship selection method to get best performance
- Extended tests on larger, more diverse taxonomies and datasets
- Application to other tasks (e.g., object detection, image segmentation)

Thank you for your attention!

Questions?

GitHub Repository:
`https://github.com/bbuschkaemper/master-thesis`