Learning to Detect Unseen Object Classes by Between-Class Attribute Transfer

by Christoph H. Lampert, Hannes Nickisch, Stefan Harmeling

presented by Abhishek Sinha

Problem Definition

Learning with Disjoint Training and Test Classes: Let $(x_1, l_1), \ldots, (x_n, l_n) \subset \mathcal{X} \times \mathcal{Y}$ be training samples where $\mathcal{X}$ is an arbitrary feature space and $\mathcal{Y} = \{y_1, \ldots, y_K\}$ consists of K discrete classes. The task is to learn a classifier $f : \mathcal{X} \to \mathcal{Z}$ for a label set $\mathcal{Z} = \{z_1, \ldots, z_L\}$ that is disjoint from $\mathcal{Y}$[1].

Lampert, Nickisch et. al.

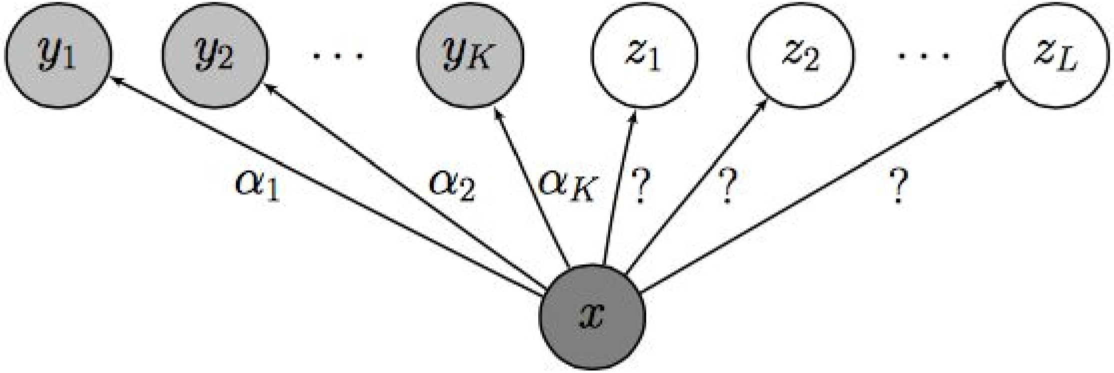
Problem Definition (Continued)

Attribute-Based Classification:
Given the situation of *learning with disjoint training and test classes*. If for each class $z \in \mathcal{Z}$ and $y \in \mathcal{Y}$ an *attribute representation* $a \in \mathcal{A}$ is available, then we can learn a non-trivial classifier $\alpha : \mathcal{X} \to \mathcal{Z}$ by transferring information between $\mathcal{Y}$ and $\mathcal{Z}$ through $\mathcal{A}$.

Lampert, Nickisch et. al.

Algorithm

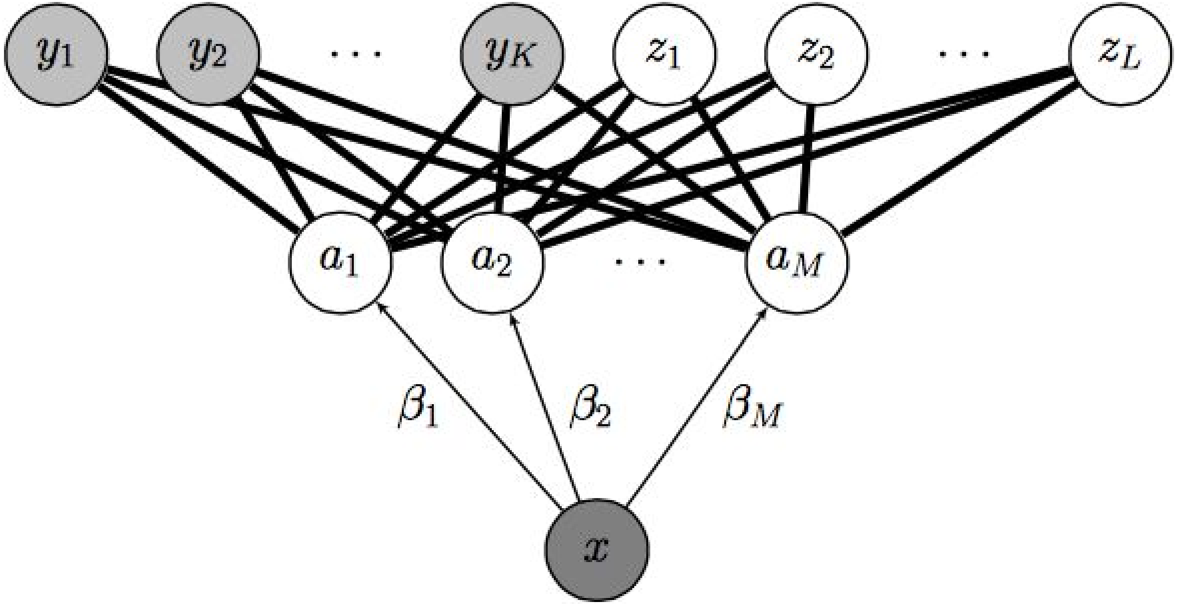
Flat Classification



(a) Flat multi-class classification

Lampert, Nickisch et. al.

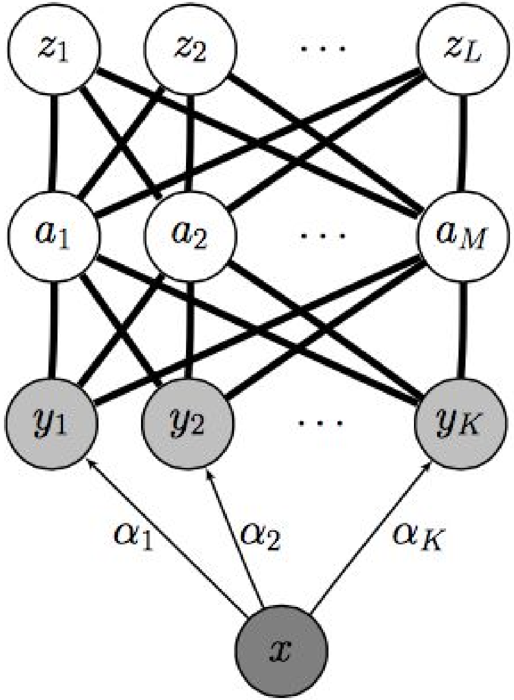
DAP



(b) Direct attribute prediction (DAP)

Lampert, Nickisch et. al.

IAP



Indirect attribute prediction (IAP)

Lampert, Nickisch et. al.

Experiments

Outline

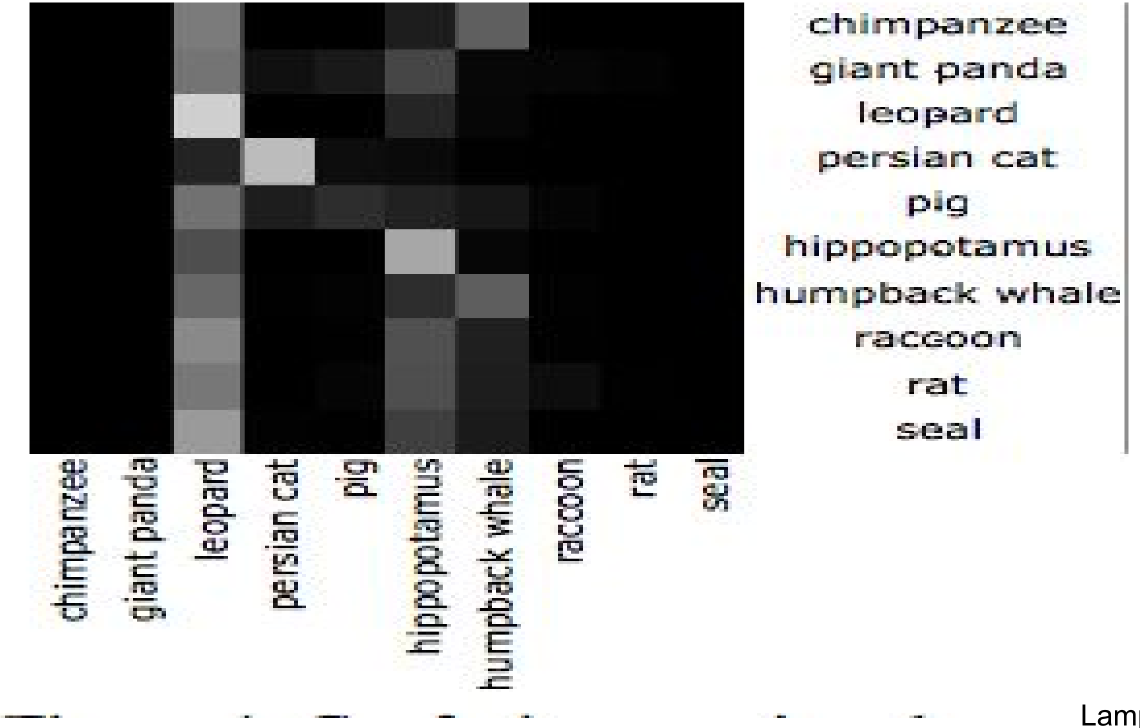
- Intermediate Layer Representations
- Impact of overlap among training and test classes
- Impact of correlation among attributes
- Results on a new dataset - SUN Attribute Database

Intermediate Layer Representations

Setup

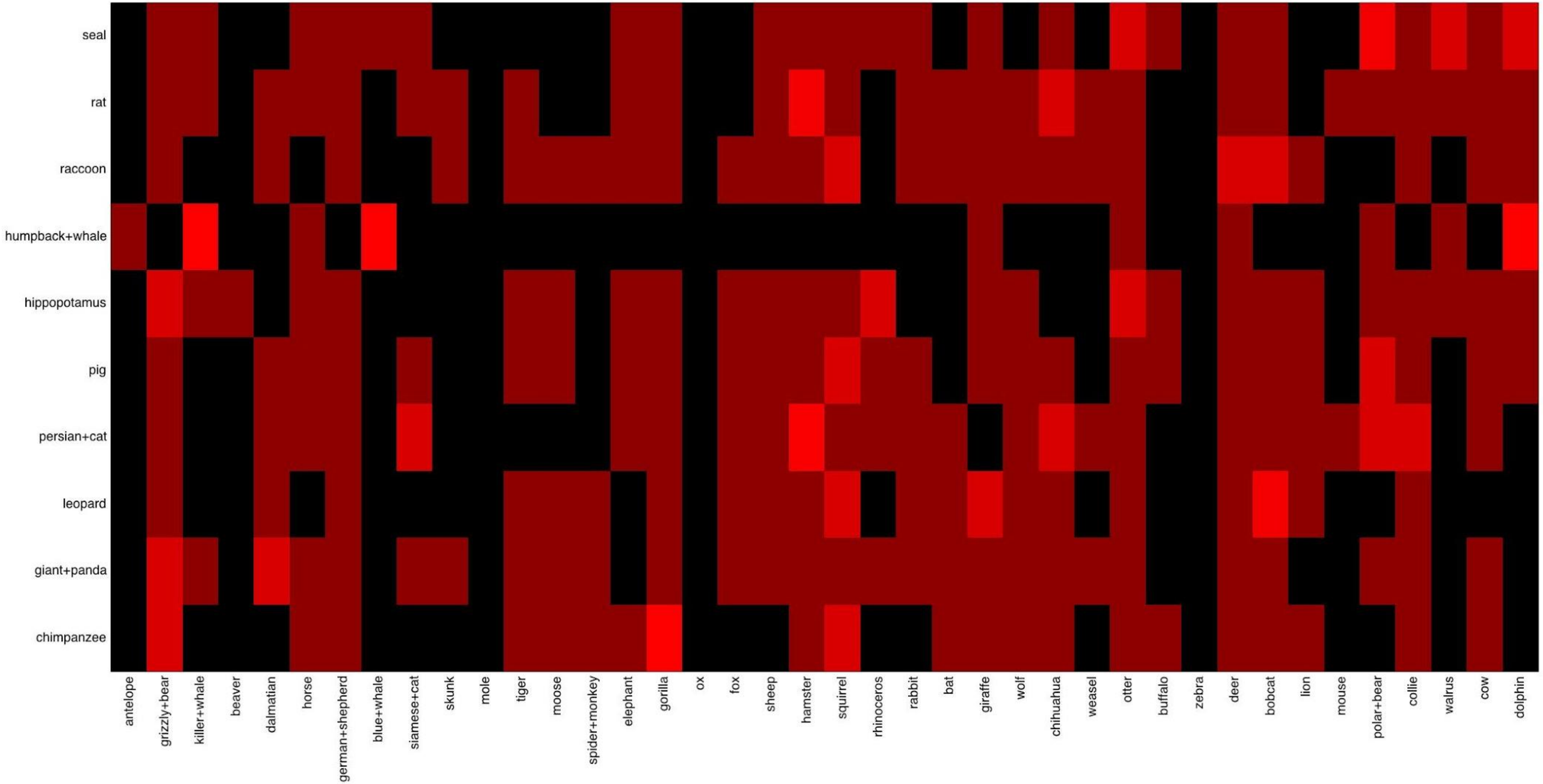
- Took the same training/test split as the paper
- Visualized the intermediate representations generated by IAP
  - HeatMap of test classes vs training classes to visualize the training class layer
  - HeatMap of test classes vs attributes to visualize the attribute layer.

Original Confusion Matrix

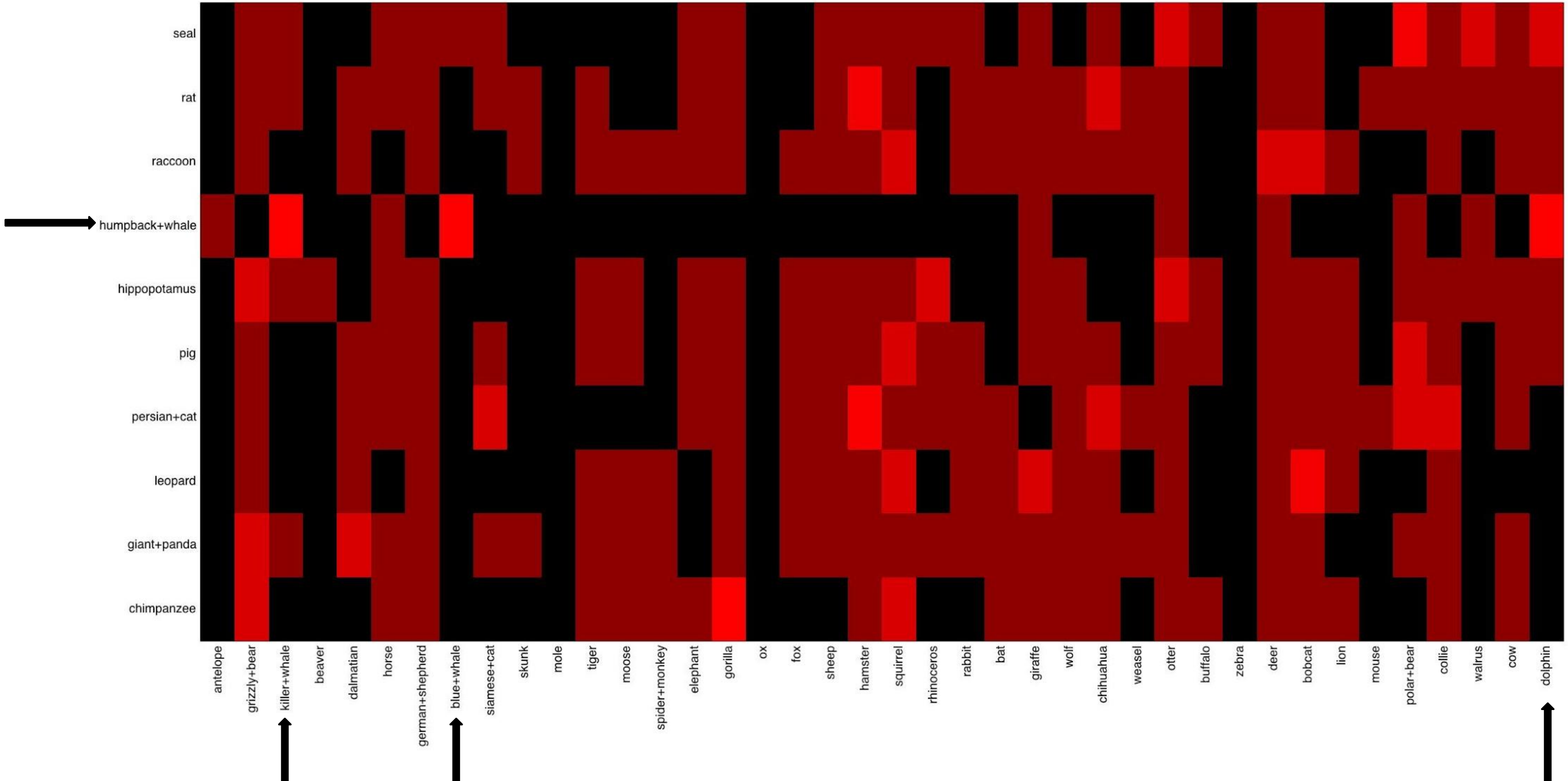


Lampert, Nickisch et. al.

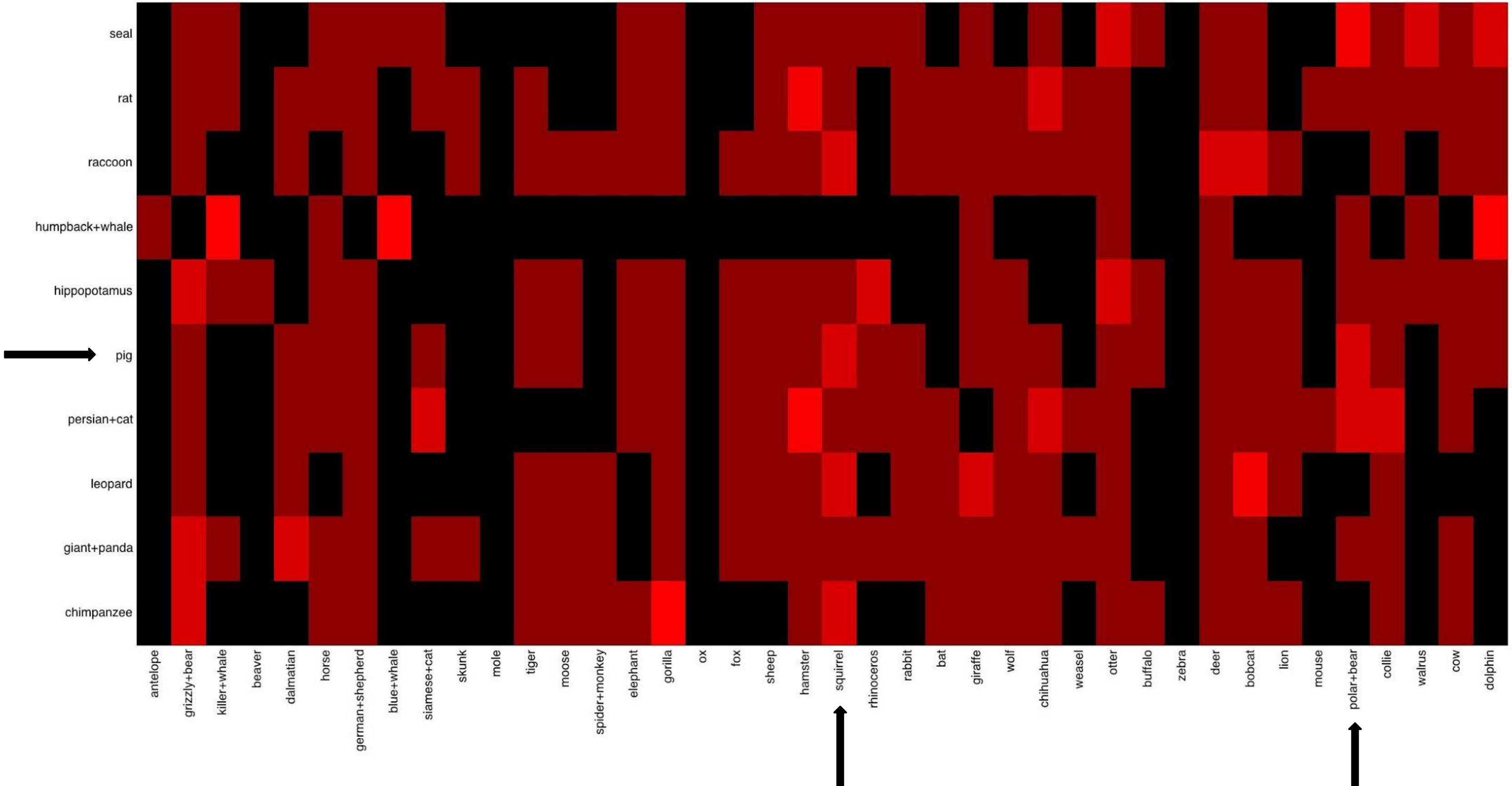
IAP Training Class Layer



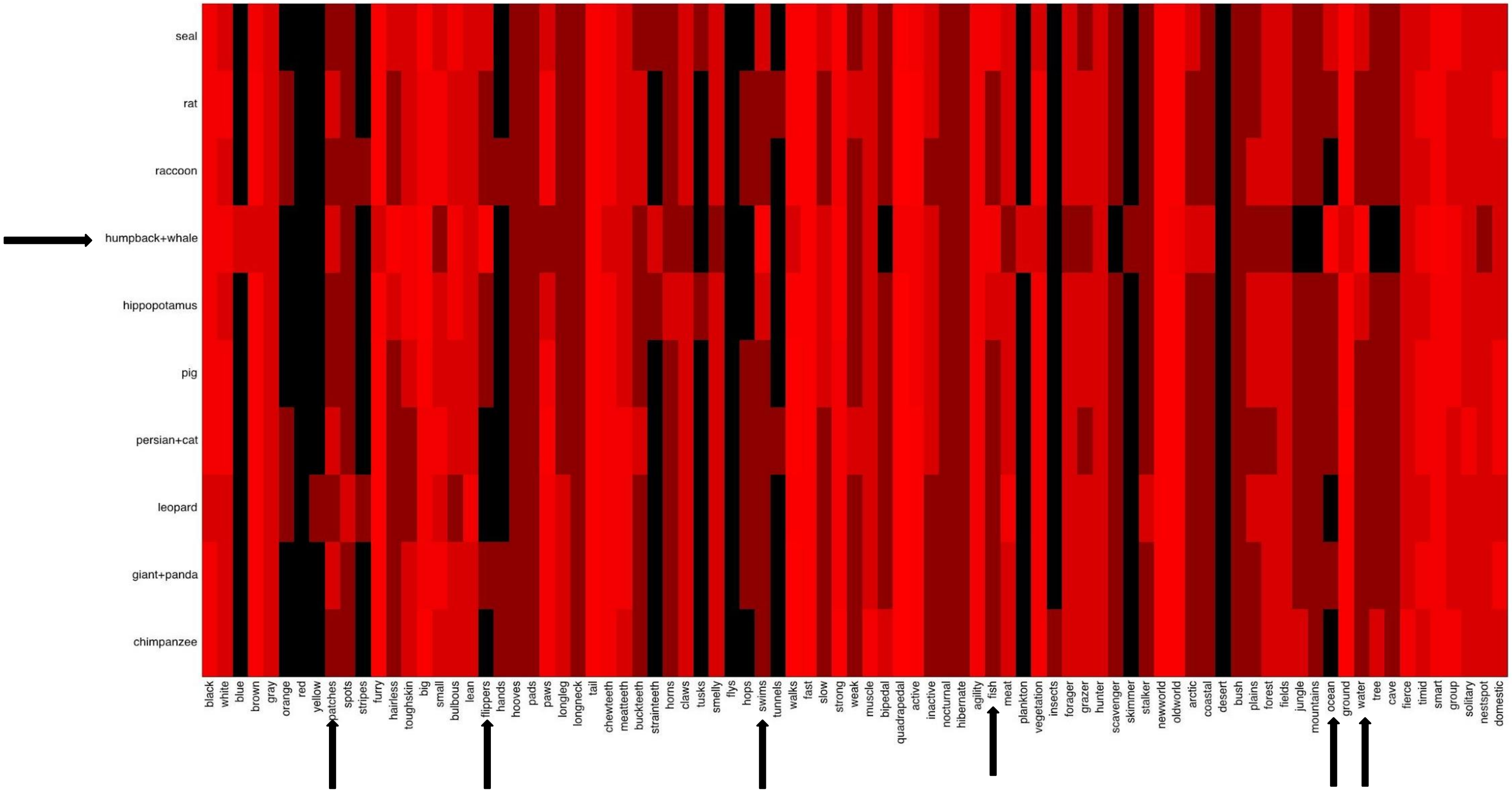
IAP Training Class Layer



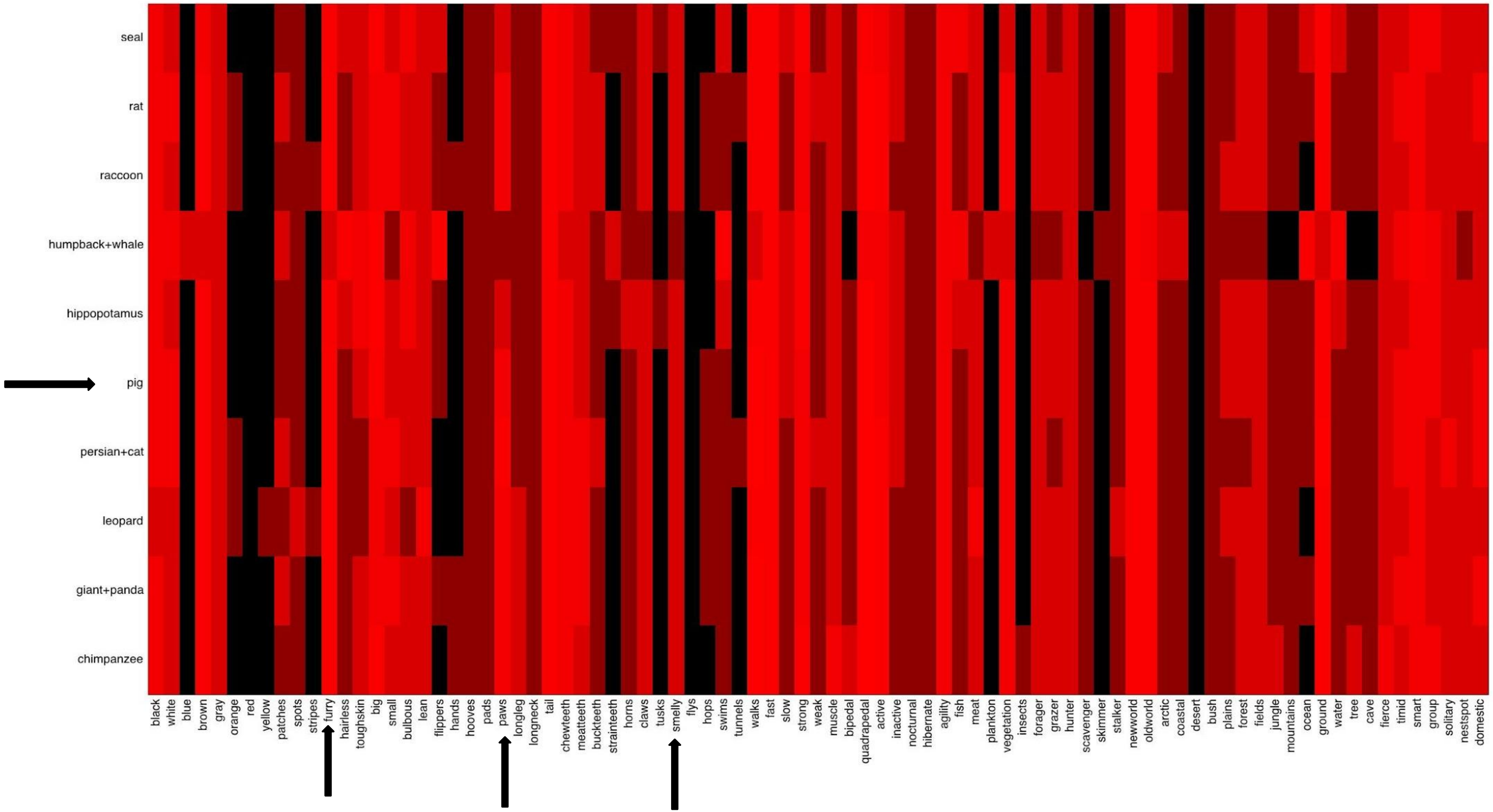
IAP Training Class Layer



IAP Attribute Layer



IAP Attribute Layer



Conclusions

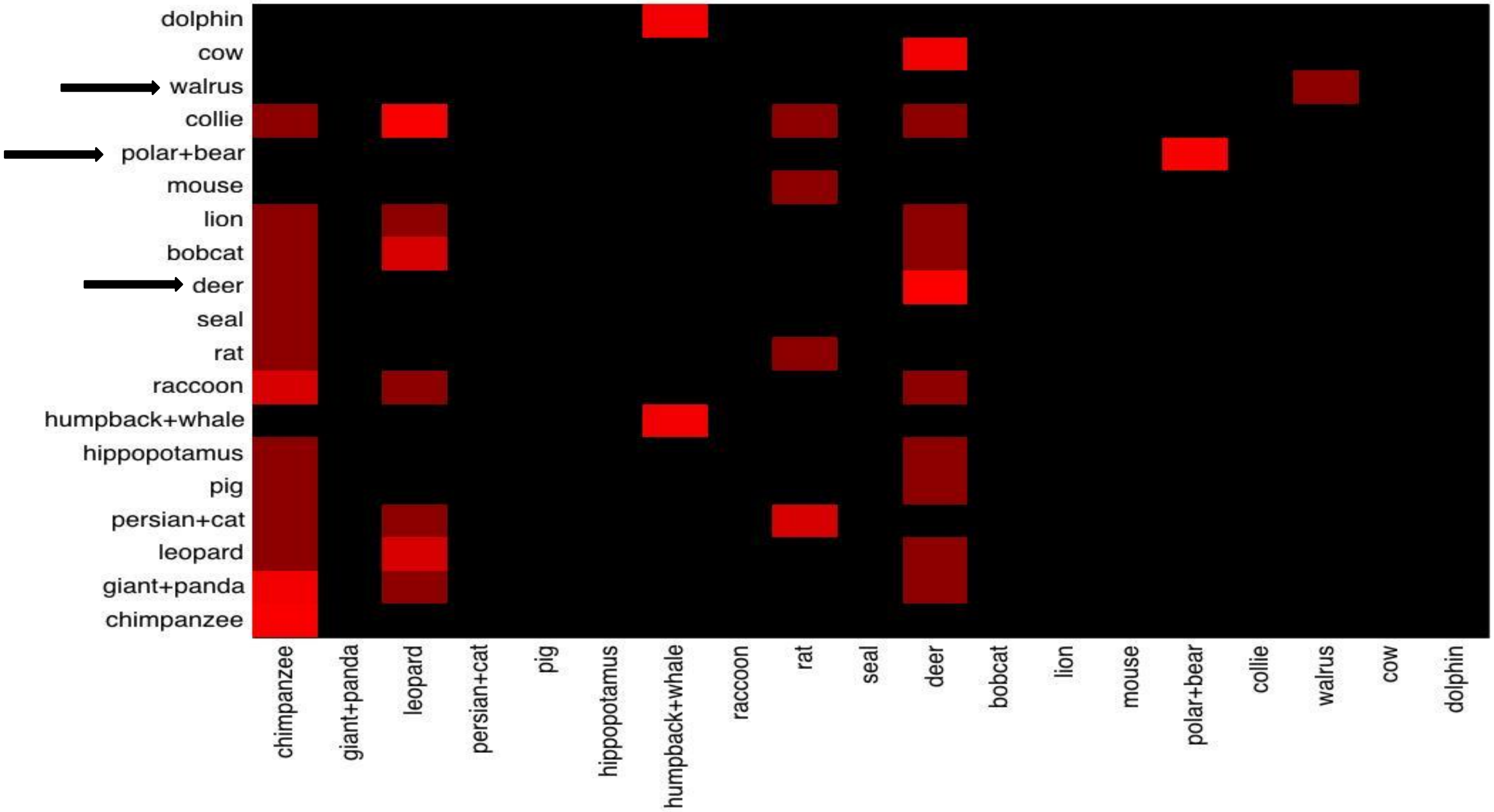
- Classes with high accuracy get mapped to similar training classes
- Classes with low accuracy do not get mapped to similar training classes
  - There aren’t similar enough classes
  - There are pretty similar classes but the algorithm doesn’t discover them
- Classes with high accuracy have good attribute representation
  - At least, one or a couple of attributes are discriminative enough and the class has a high score on it.
- Attributes with lower accuracy either have
  - low score for relevant discriminating attribute
  - poor attribute representation - all attributes with high score are too general.

Overlapping Test and Train Classes

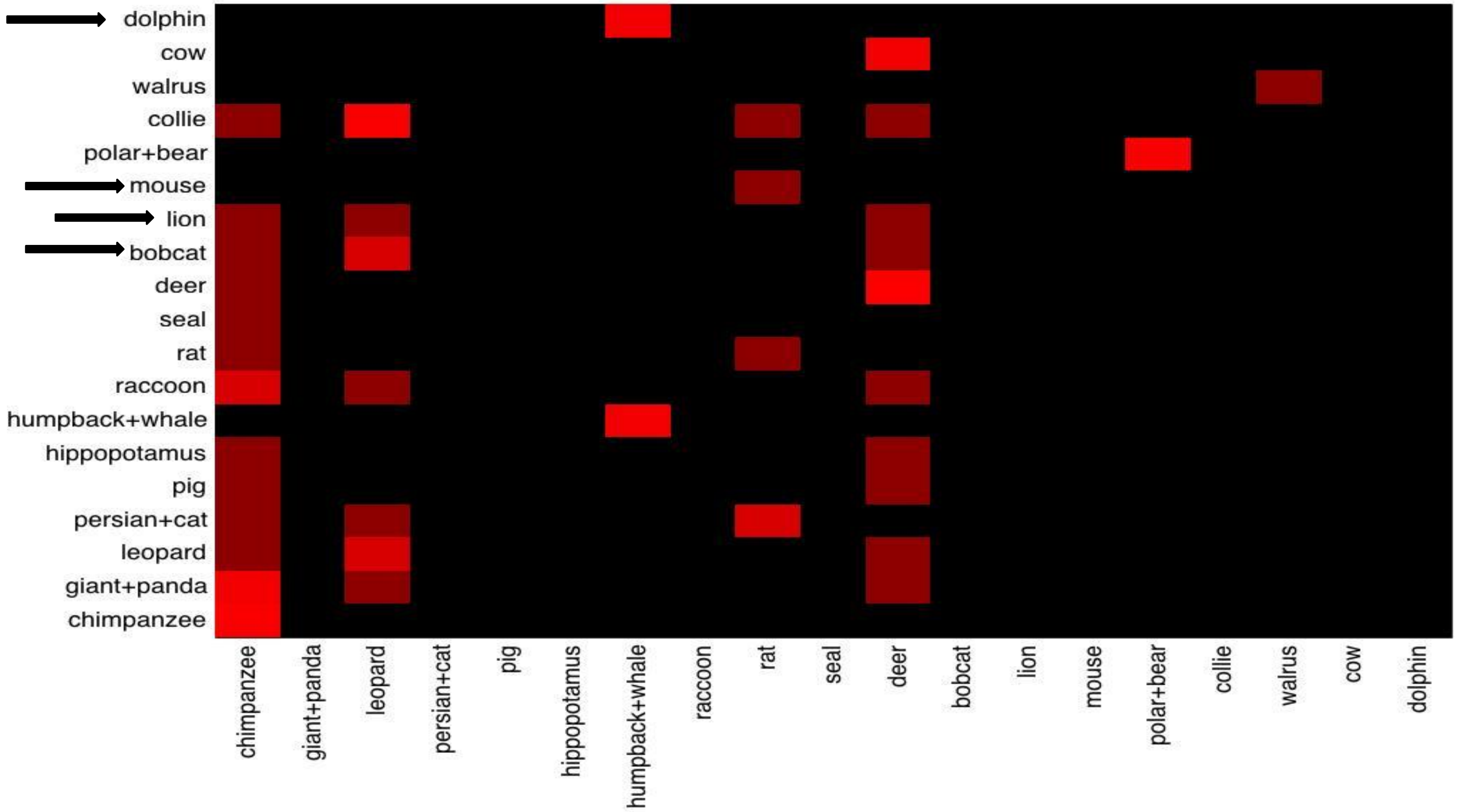
Setup

- Took 40 training and 19 test classes with 9 overlapping classes
    - deer, bobcat, lion, mouse, polar+bear, collie, walrus, cow, dolphin
- Used the same feature space as the paper
- Visualized the training class layer representation, attribute layer representation and confusion matrix
- Overall test class accuracy decreased from 27.4% to 26.5%

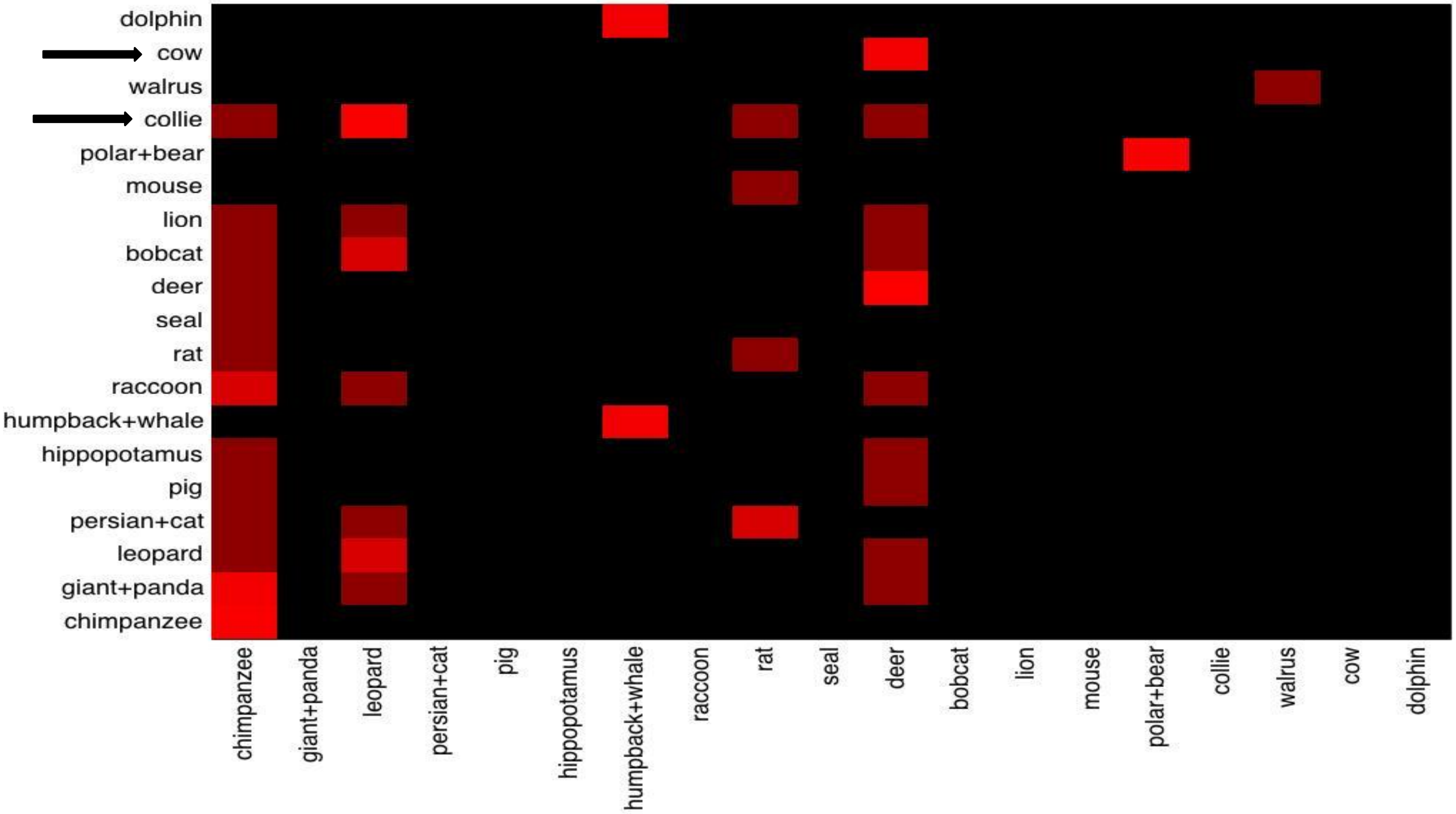
Final Confusion Matrix



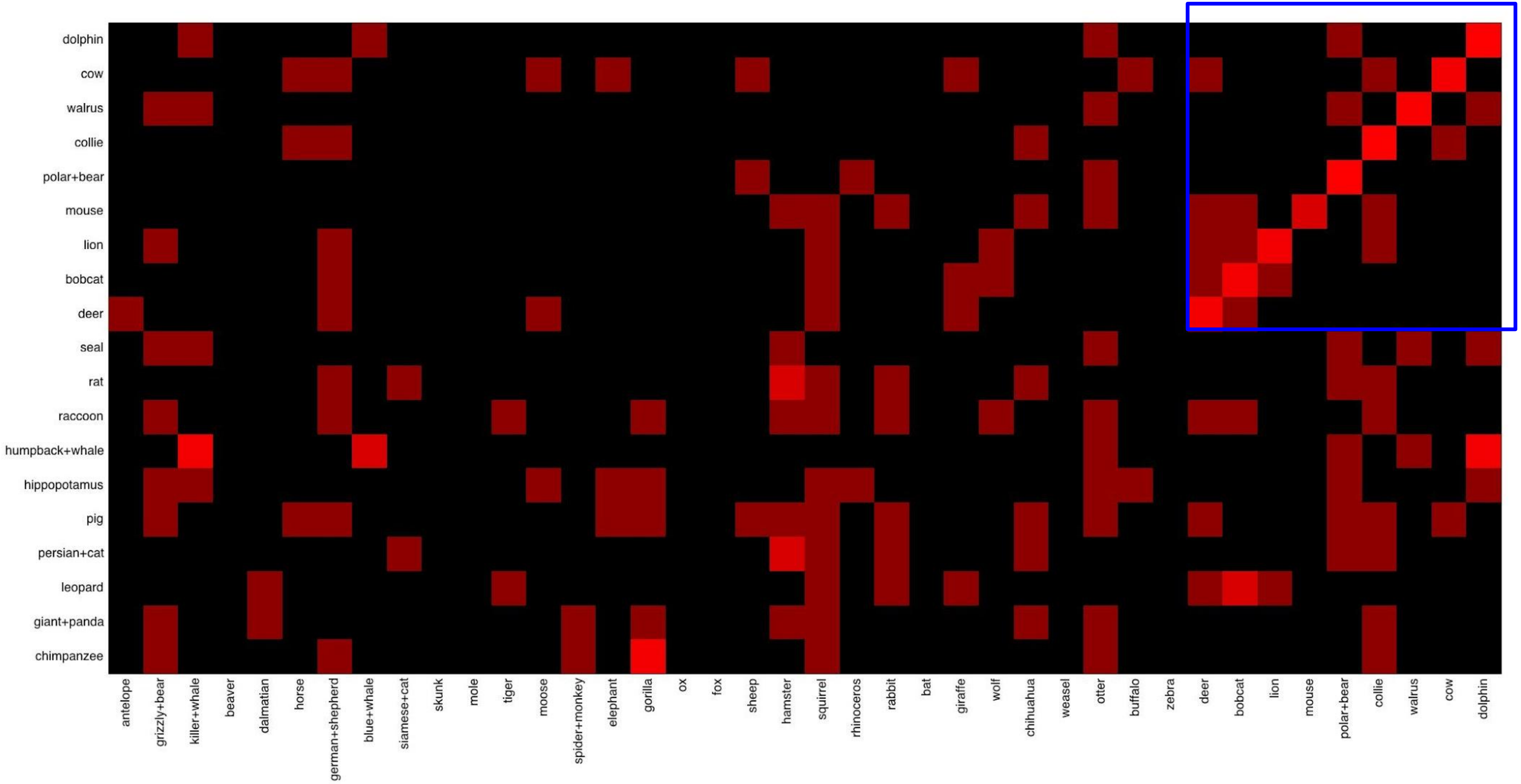
Final Confusion Matrix



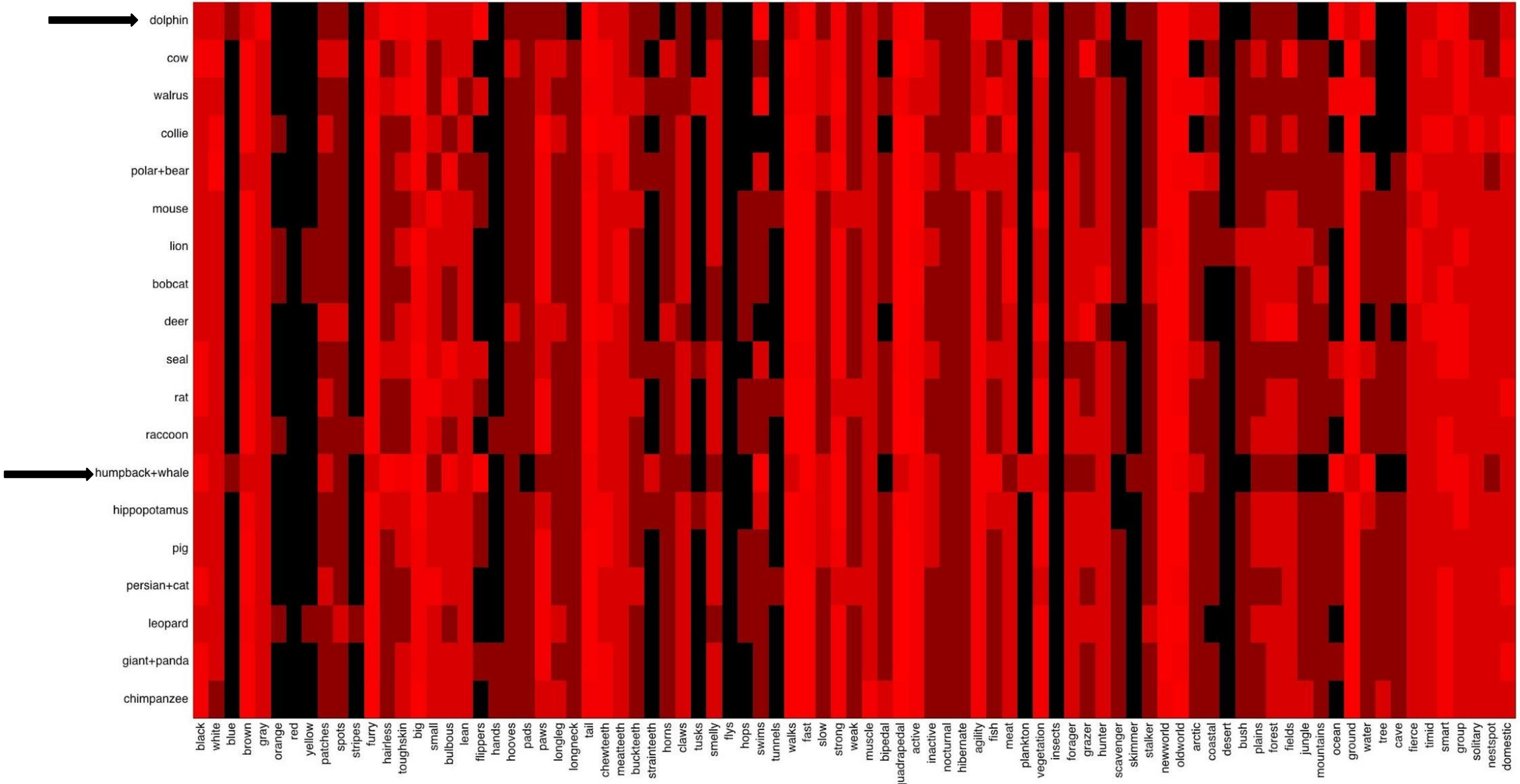
Final Confusion Matrix



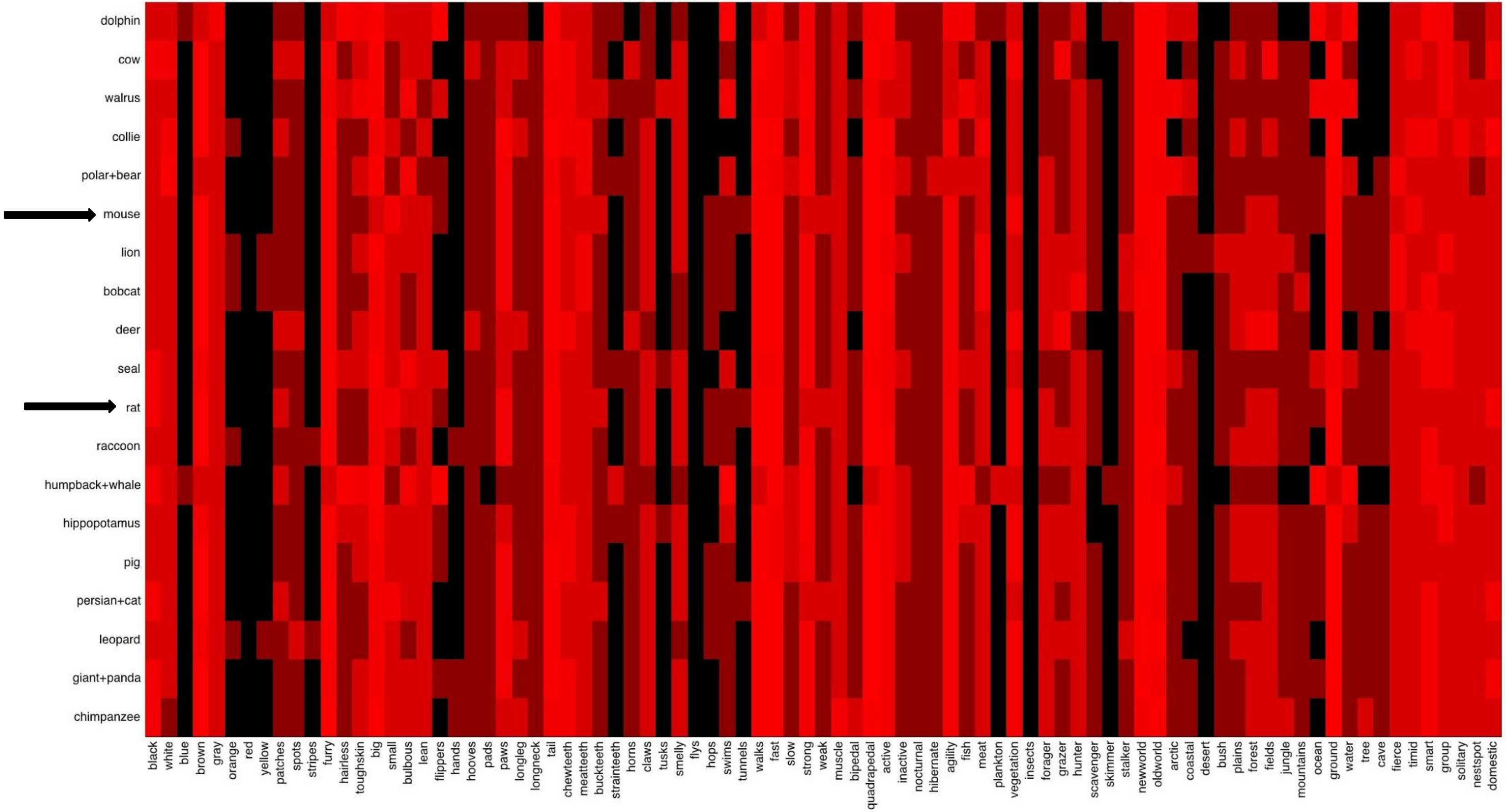
IAP Training Classes Layer



IAP Attribute Layer



IAP Attribute Layer



Conclusions

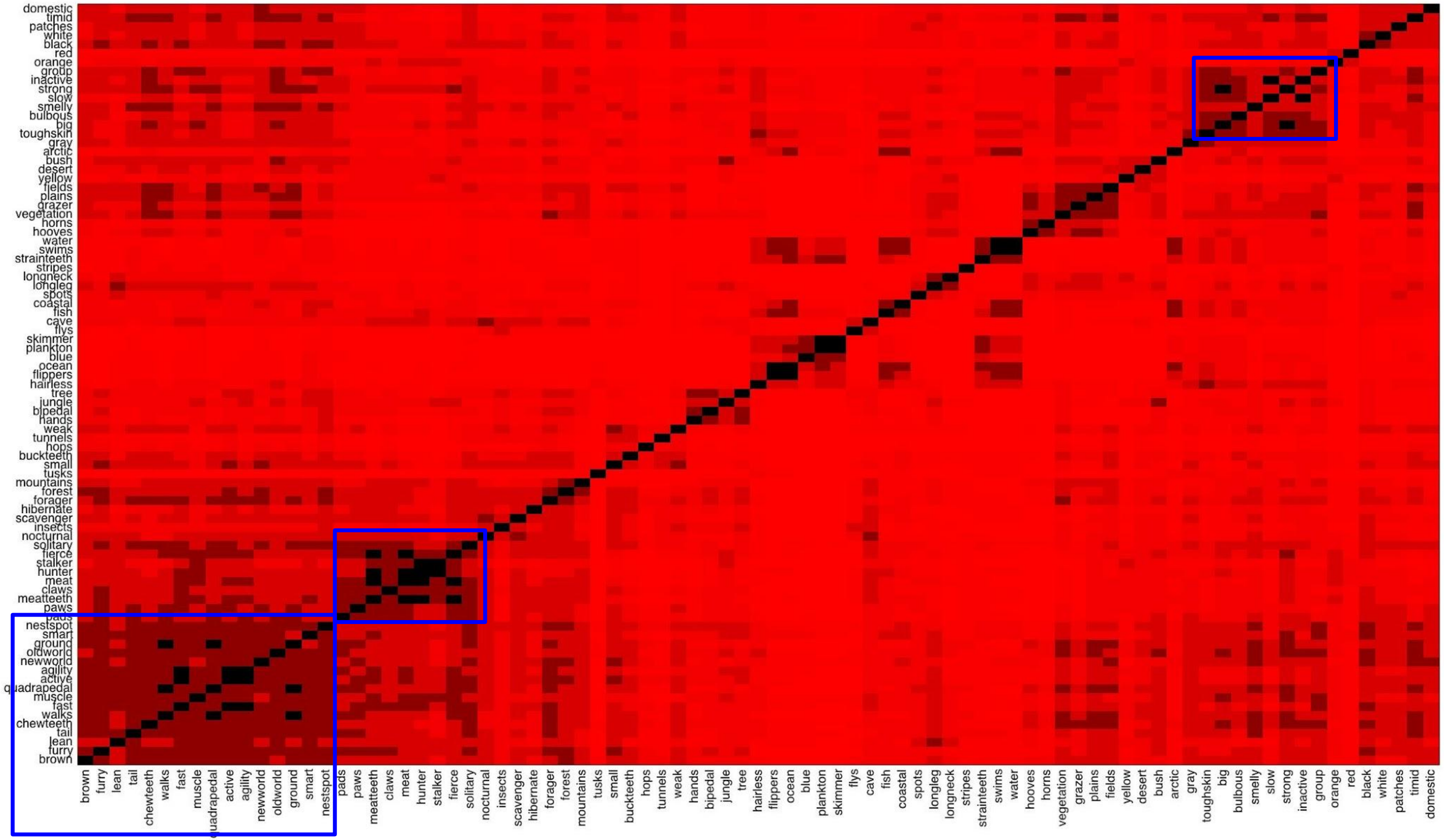
- Overlapping classes get correctly mapped at the training class layer
- But attribute representation in this case ambiguates the situation
  - Loss of Information
  - The final test class ends up being wrong
- Overlapping classes are not easy instances for IAP if there exist other similar test classes

Impact of Correlation

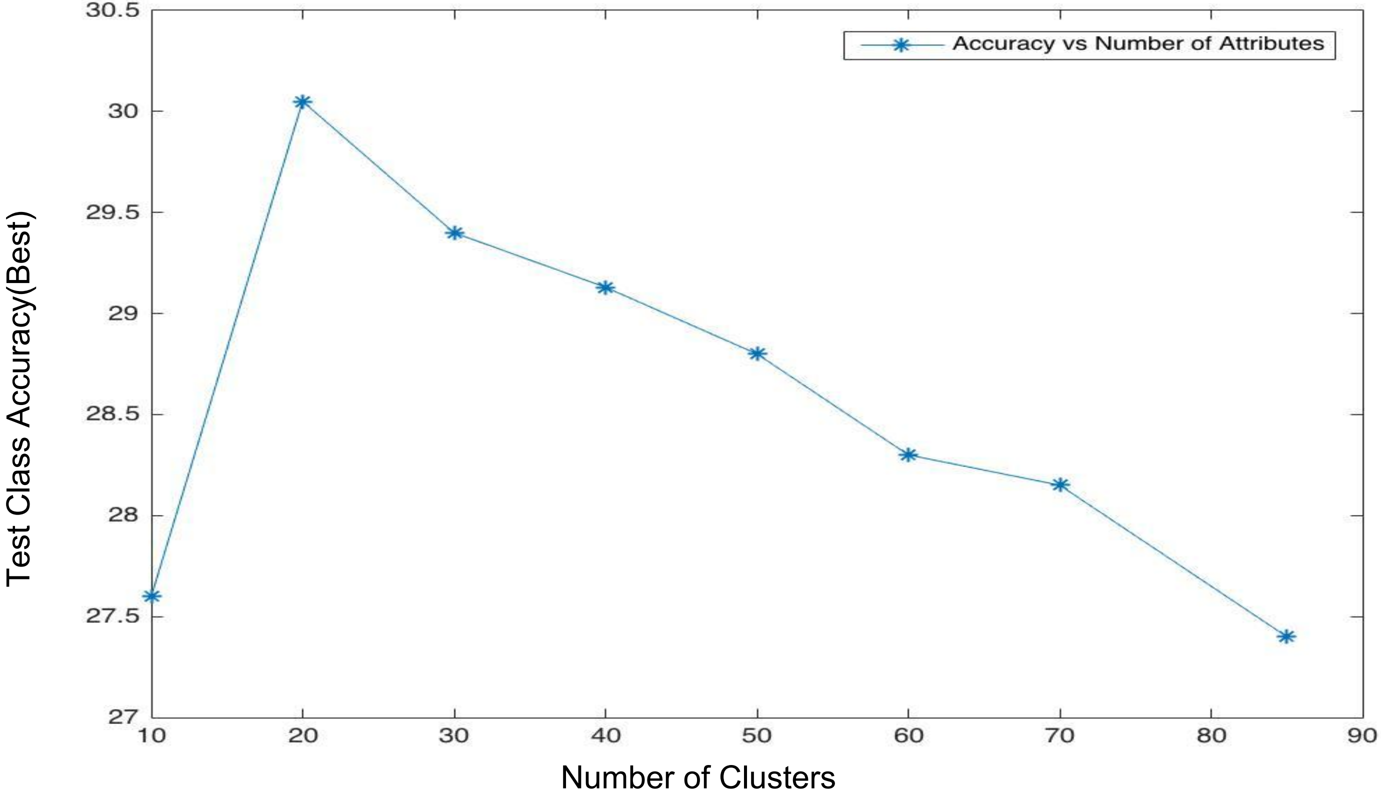
Setup

- First plotted the 85 x 85 distance matrix where each entry is the cosine distance between the corresponding attributes.
  - Attributes are represented as class vectors (containing a score for each class in the dataset).
- Clustered the attributes using the above cosine distance metric.
  - Each cluster can be looked at as a *Super Attribute*
- Computed the variation of final test class accuracy with number of clusters

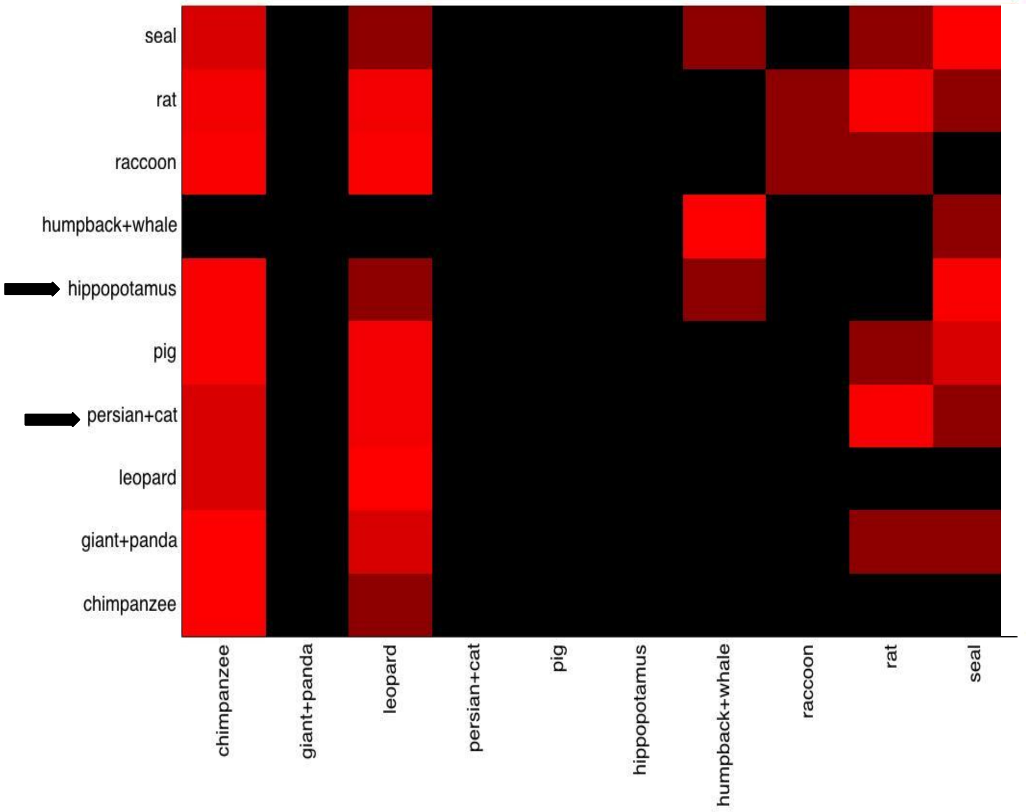
Correlation Among Attributes

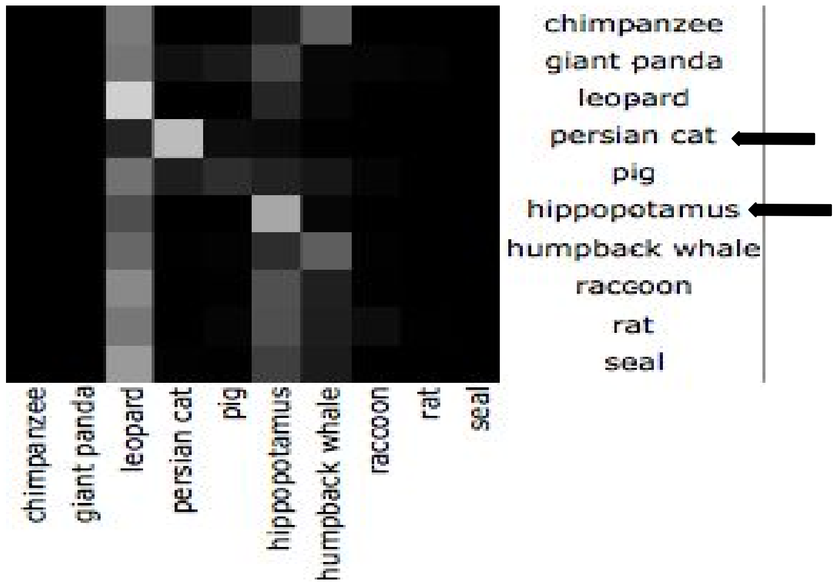


Accuracy vs Number of Clusters



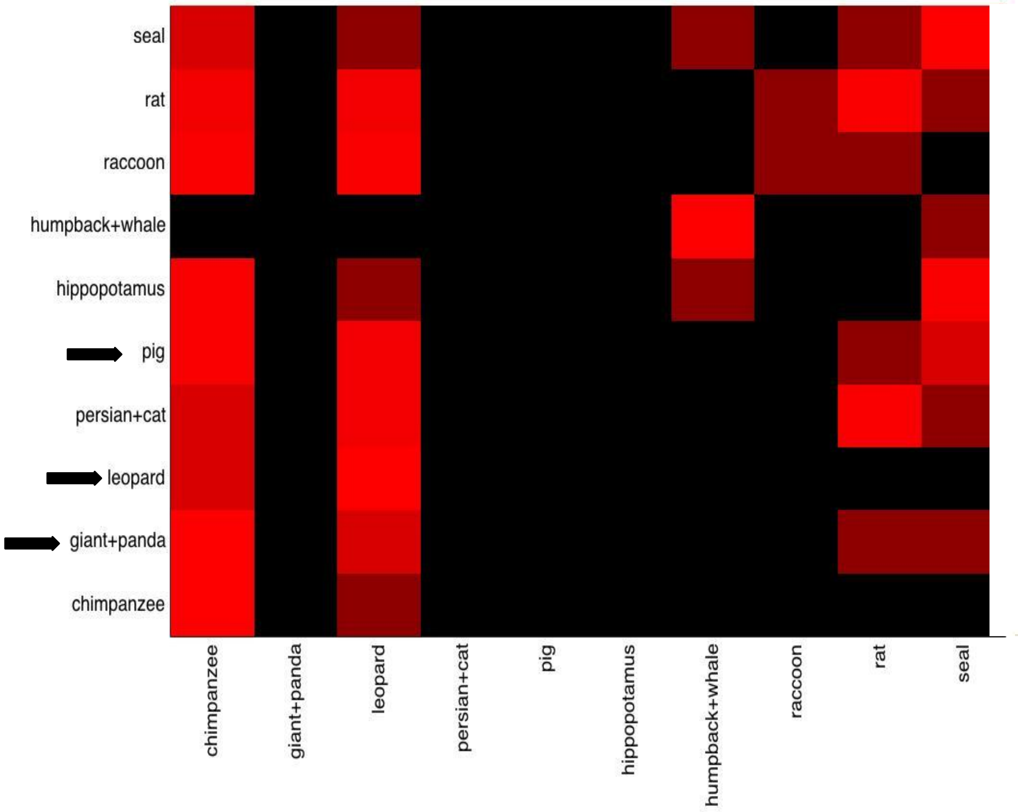
Confusion Matrix for Best Case - Worse Off Classes

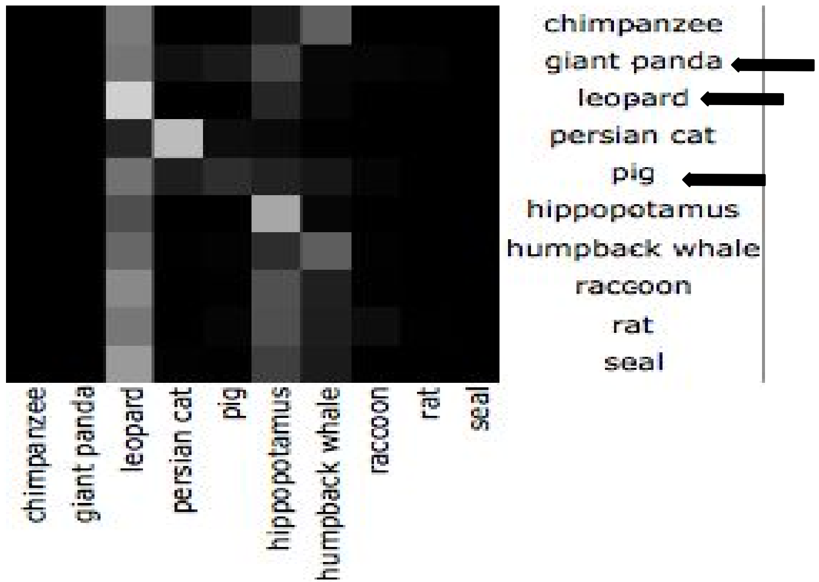




Lampert, Nickisch et. al.

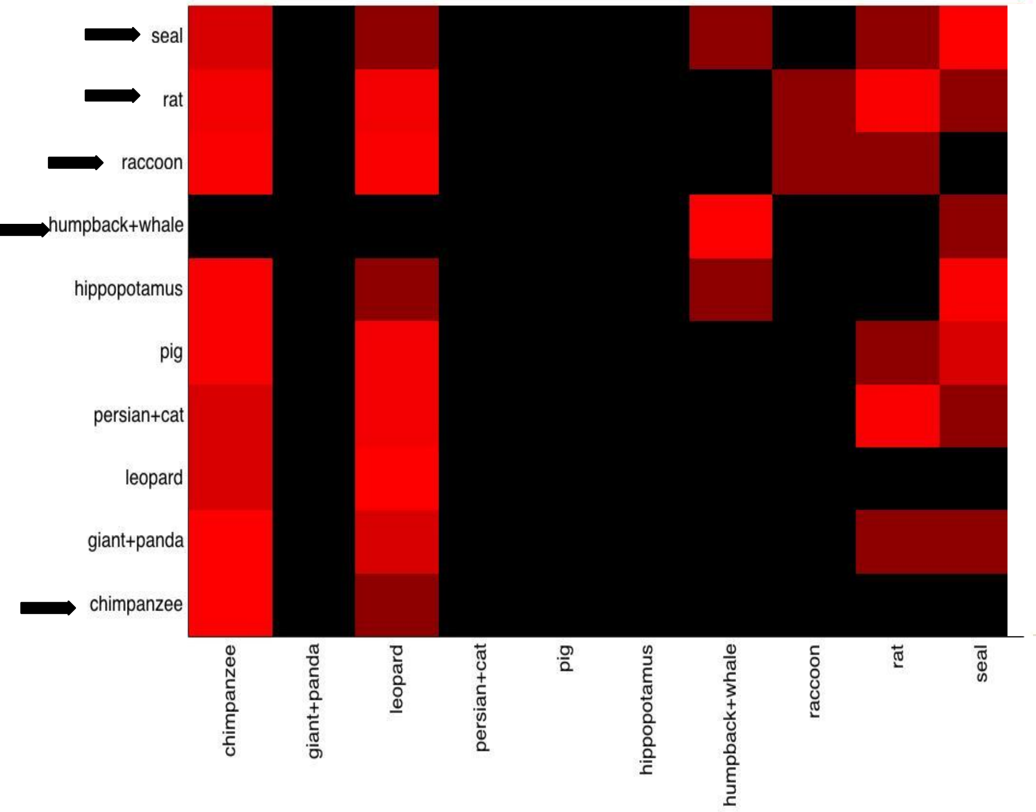
Confusion Matrix for Best Case - Same Classes

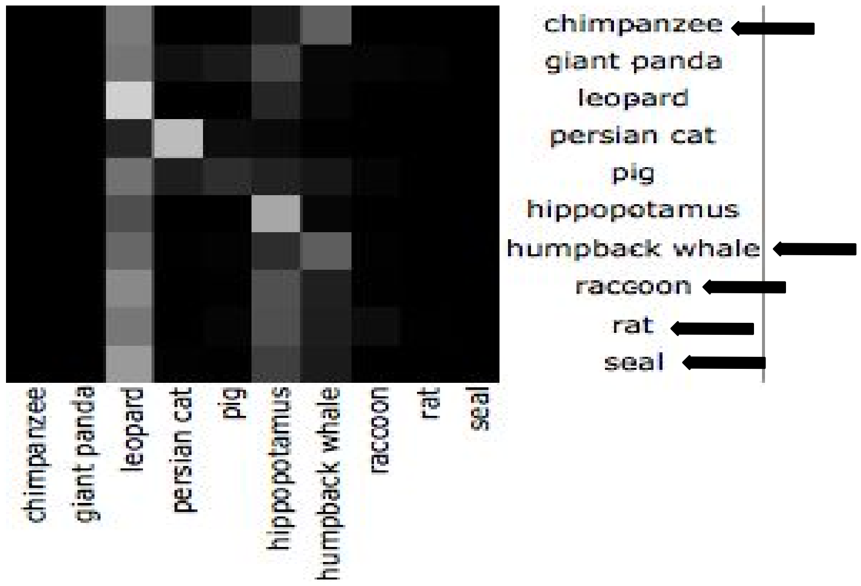




Lampert, Nickisch et. al.

Confusion Matrix for Best Case - Better Classes





Lampert, Nickisch et. al.

Examples of Super Attributes

'brown', 'furry', 'lean', 'tail', 'chewteeth', 'walks', 'fast', 'muscle', 'quadrapedal', 'active', 'agility', 'newworld', 'oldworld', 'ground', 'smart', 'nestspot'

wikipedia

wikipedia

Conclusion

- For classes that were pretty ‘close’, clustering actually leads to decrease in the accuracy.
    - e.g. Persian Cat and Leopard were earlier identified correctly but now both get mapped to leopard.
- For many other classes, clustering helps in removing noise and avoid accidental similarities.
    - e.g. Rat initially had high score along ‘paws’, ‘claws’ which was probably why it was getting mapped to leopard
    - After clustering, it will no longer get mapped to the super attribute containing [‘paws’,’claws’] since the super attribute also contains many other attributes not relevant to it.
    - More likely to get mapped to the super attribute containing [‘brown’, ‘furry’,’tail’,’chewteeth’,’ agility’] which makes it easier to identify.

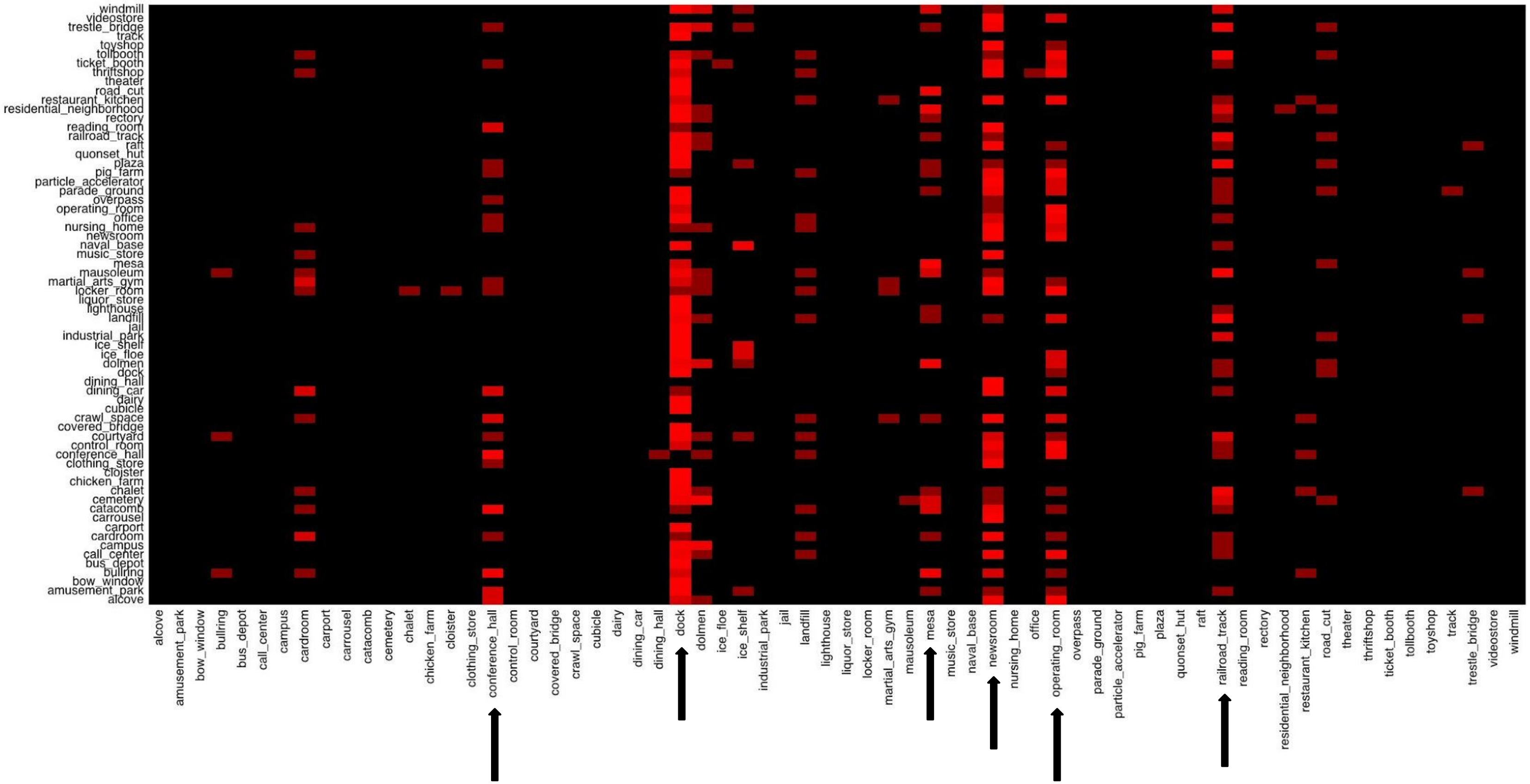
SUN Attribute Database

Description of Database[1] and Experiment

- Around 14000 images of 600 odd *scene* categories.
    - Categories such as airport, jail, kitchen, waterfall etc.
- 102 scene attributes
    - Attributes describe what objects those scenes contain as well as the activities performed
    - Attributes include biking, hiking, studying, trees etc.
- Split the 600 odd classes into 550 randomly chosen train classes and around 60 test classes
- Attained only 4.7% accuracy on the test classes

https://cs.brown.edu/~gen/sunattributes.html

Results


windmill
videostore
trestle_bridge
track
toyshop
tollbooth
ticket_booth
thriftshop
theater
road_cut
restaurant_kitchen
residential_neighborhood
rectory
reading_room
railroad_track
raft
quonset_hut
plaza
pig_farm
particle_accelerator
parade_ground
overpass
operating_room
office
nursing_home
newsroom
naval_base
music_store
mesa
mausoleum
martial_arts_gym
locker_room
liquor_store
lighthouse
landfill
jail
industrial_park
ice_shelf
ice_floe
dolmen
dock
dining_hall
dining_car
dairy
cubicle
crawl_space
covered_bridge
courtyard
control_room
conference_hall
clothing_store
cloister
chicken_farm
chalet
cemetery
catacomb
carrousel
carport
cardroom
campus
call_center
bus_depot
bullring
bow_window
amusement_park
alcove

Conclusion

- Results are much worse than on the *Animals with Attribute* dataset
- One of the reasons is number of training samples per class
  - Animals with Attributes - 30,000 images for 50 classes
  - SUN Attribute DB - 14000 images for around 600 classes
- Predicate Matrix is sparser for the SUN Attribute DB case
- Possibly easier to specify discriminating attributes for animals than scenes
- IAP has a tendency to output only a small percentage of all test classes
  - In the original paper, 5 of the 10 test classes have zero weight
  - This tendency might be getting magnified because of the sparseness in the data

Questions