

CSci 8980

**Edge-based Discovery of  
Training Data for Machine  
Learning**

CMU authors

# Deep Learning Recipe

- Collect a large amount of data and label it
- Select a model and train a DNN
- Deploy the DNN for inference

# Labelled Data

- Some data are easy to label ...



- Some require domain expertise

Valuable in ecology, military intelligence, medical diagnosis, etc.

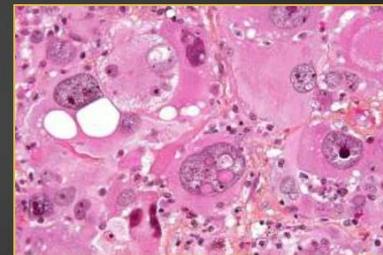
- Low **base rate** (prevalence) in the data
- Requires **expertise** to identify



Masked palm civet (*Paguma larvata*).  
Transmitter of SARS during its 2003  
outbreak in China.



BUK-M1. Believed to have shot down  
MH17 and killed 298, 2014.



Nuclear atypia in cancer.

# Building a test set is hard

- Non-expert crowd-sourcing won't work
- Data may have privacy or other restrictions
- Need  $10^x$  or more training samples for DNN
- Expert may need to shift through  $10^y$ ,  $y \gg x$  samples; experts are \$\$
- **Goal: make expert's life easier**
  - Optimize “human-in-the-loop” time

# Eureka Approach

- Focus on image labelling
- Assume images are widely distributed and come from different sources
  - Even live streams, e.g. IoT
  - Can turn on/off data sources
- Support the expert in the labelling process
  - Early discard => filter or classifier that says “NO WAY”
  - Iterative discovery workflow
  - Edge computing

# Stolen slides begin now

## Eureka's Architecture

Only a tiny fraction of data along with meta-data is transmitted and shown to user, consuming little Internet bandwidth.



Expert with domain-specific GUI

Internet

High-bandwidth, low-latency access



cloudlet

LAN



Archival Data Source



cloudlet

LAN



Archival Data Source



cloudlet

LAN

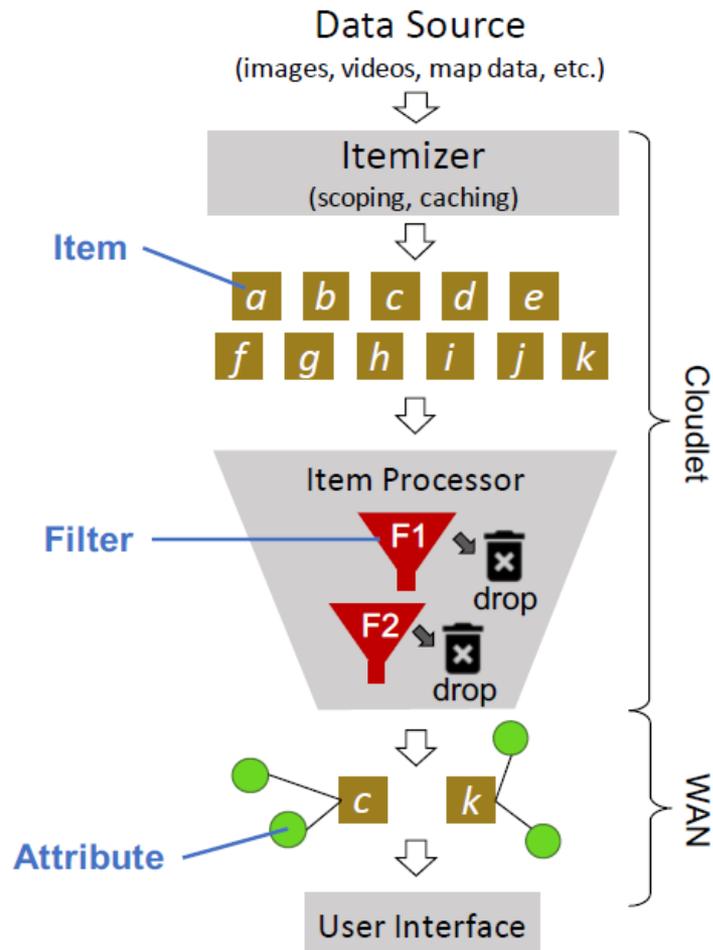


Live Video

Executes **early-discard** code to drop clearly irrelevant data

cloudlet = edge node near data source

# Edge node (cloudlets) run Filters



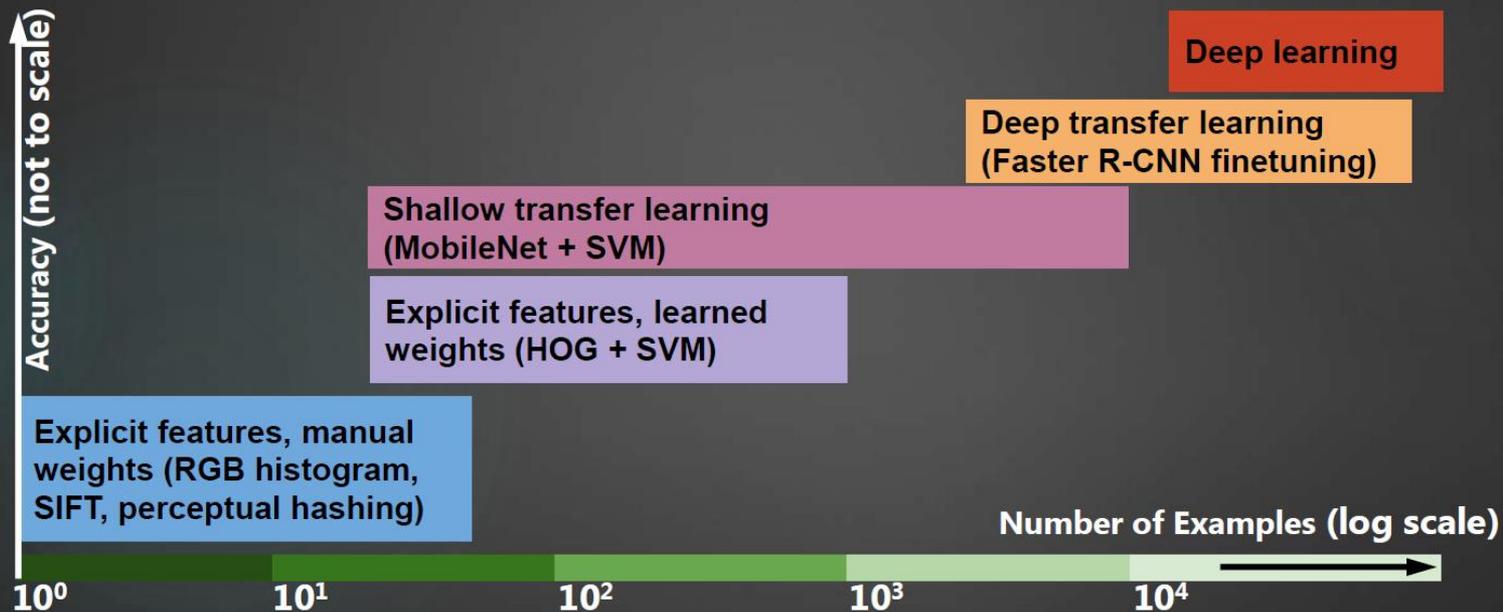
# Example GUI: Finding Deer

The screenshot shows the Eureka GUI interface. On the left, the 'Predicates' section is highlighted with a red box and contains the following filters:

- DOG Texture  
fur texture
- RGB Histogram  
fur color
- RGB Histogram  
grass color

Below the predicates is the text 'Early-discard filters'. At the bottom left, there are buttons for 'Define Scope', 'Start', 'Stop', 'Export', and 'Import'. The main area displays a grid of 9 images with green bounding boxes around objects. The bottom status bar shows: 'Get next 500 results' and 'Total 152276, Searched 152135, Dropped 151048 (99.29%), Passed 1087 (0.71%)'.

# Iterative Discovery Workflow



=> More data ... Better classifiers ... Control false positives!

# Finding Deer (after a few iterations)

The screenshot displays the Eureka software interface. On the left, the 'Predicates' panel is active, showing a checked box for 'DNN + JIT SVM svm5'. Below this are buttons for 'Define Scope', 'Start', 'Stop', 'Export', and 'Import'. The main window shows a 3x3 grid of nine images of deer in various settings. At the bottom, a status bar indicates 'Total 10531, Searched 10470, Dropped 9789 (93.50%), Passed 681 (6.50%)' and a button to 'Get next 500 results'.

Codec: Built-in Edit

Predicates

DNN + JIT SVM svm5 Edit X

Define Scope

Start Stop

Export Import

Get next 500 results

Total 10531, Searched 10470, Dropped 9789 (93.50%), Passed 681 (6.50%)

# Matching

- Optimize user time/attention
- Deliver data to expert at a rate they can handle
  - Human labelling time  $\gg$  Single filter time
- Too fast – overwhelmed with data
  - Fewer cloudlets (less data) or deeper filter
- Too slow – kept waiting
  - More cloudlets (Watch false positives)

# Evaluation: Case Studies



Deer



Taj Mahal



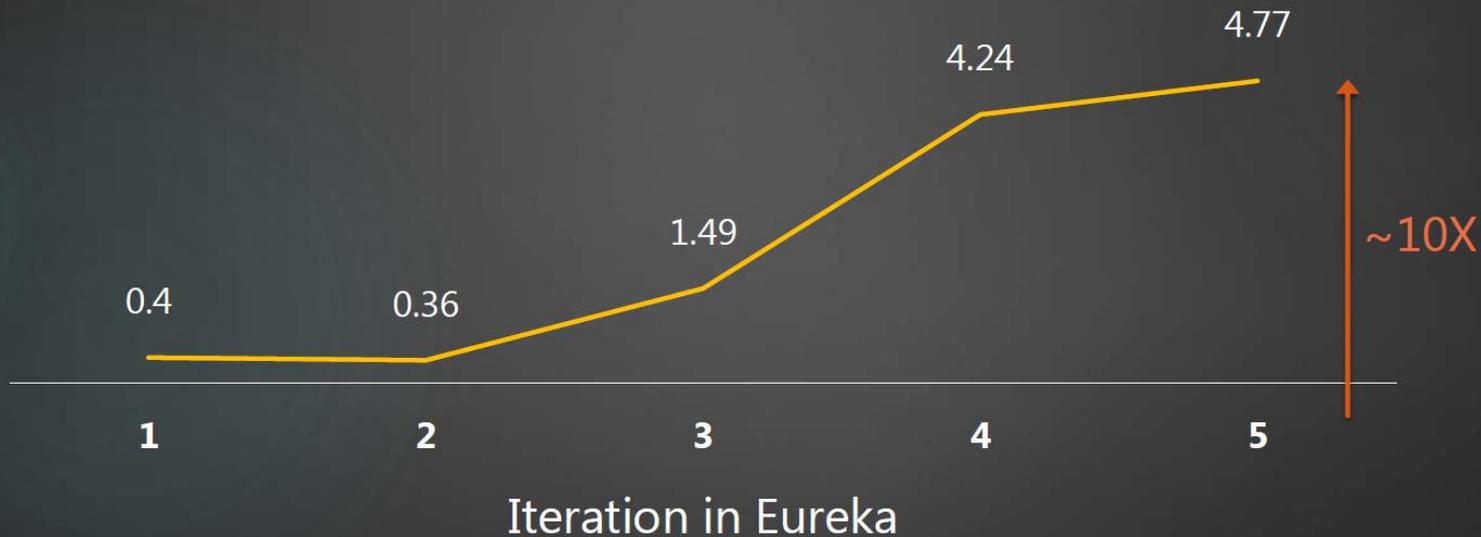
Fire hydrant

Estimated base rate	0.07%	0.02%	0.005%
Collected positives in evaluation	111	105	74
Images viewed by user	7,447	4,791	15,379
Images discarded by Eureka	2,104,076	2,542,889	2,734,070

# Iteratively Improving Productivity

The case of deer

Productivity (New true positives / minute)



# Discussion

- Creating data labels is time-consuming
- Discussion
  - Assumptions: data can come from anywhere
  - Expert data: is this true?