CSC 390
Topics in Artificial Intelligence

"Unsupervised Machine Learning"

Fall 2016
Prof. Sara Mathieson
Smith College

Outline: 11/10

- Today:
  - **Deepshikha** 10:30am
  - **Sarah** 10:45am
  - **Maria** 11:00am
  - **Amelia** 11:15am
  - **Ravinder** 11:30am

- Office Hours today: 4-5pm, Ford 355
- Homework 6 (Project Proposal) due Nov 17 (Thurs)

- Presenters:
  - Speak loudly
  - I will give you a 2 min warning after 10 minutes

- Audience:
  - Give presenters your full attention and ask questions
  - I will ask questions too

Overcoming Key Weaknesses of Distance-Based Neighbourhood Methods using a Data Dependent Dissimilarity Measure

By: Kai Ming Ting, Ye Zhu, Mark Carman, Yue Zhu, Zhi-Hua Zhou

Keywords:

KNN Anomaly Detection

Multi-Label kNN classifier

Mass-Based Dissimilarity (get back to you at the end)

Scientific Question

Overcoming Key weaknesses by replacing distance-based algorithm with the data-dependent dissimilarity measure

Algorithm	Key function	Key weakness
Density-based clustering	Identify core points which have high density using distance-based neighbourhood estimation	Inability to find all clusters of varying densities
kNN anomaly detector	Identify anomalies as points with the longest distance to the kth nearest neighbours	Inability to detect local anomalies
Multi-label kNN classifier (MLkNN)	Estimate class-conditioned likelihoods using a frequency estimate based on k nearest neighbours	Poor likelihood estimation in cases where the local neighbourhood covers regions of varied density

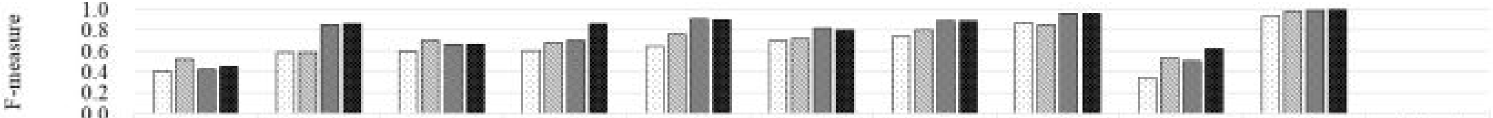
Proposed Solution

Replace the distance calculating function with mass based dissimilarity!!

Run all other part of algorithm based on ‘that’ dissimilarity matrix.

Findings on Clustering

- Used 10 datasets that has from 2 to 15 clusters
- Compare the F-measure between those two algorithms



	Libras	Thyroid	Segment	WDBC	Wine	Pendig	Seeds	Iris	S1	S2	#Top 1
DBSCAN	0.40	0.58	0.59	0.60	0.65	0.70	0.75	0.87	0.34	0.94	0/10
OPTICS	0.52	0.59	0.70	0.68	0.76	0.72	0.80	0.85	0.53	0.98	2/10
SNN	0.42	0.85	0.66	0.70	0.91	0.82	0.89	0.96	0.50	0.99	3/10
MBSCAN	0.45	0.86	0.67	0.86	0.90	0.80	0.889	0.963	0.62	0.993	5/10

Figure 6: Best F-measures of DBSCAN, OPTIC, SNN and MBSCAN on 10 datasets. The best performer on each dataset is underlined.

Findings on KNN Anomaly Detection

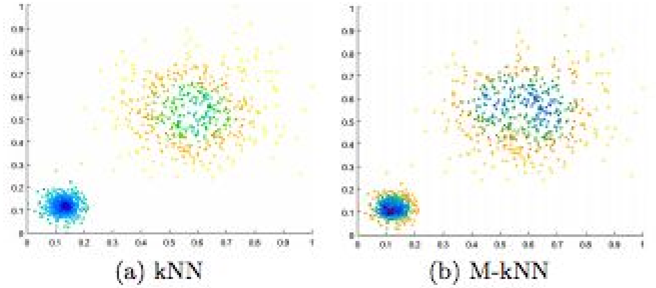
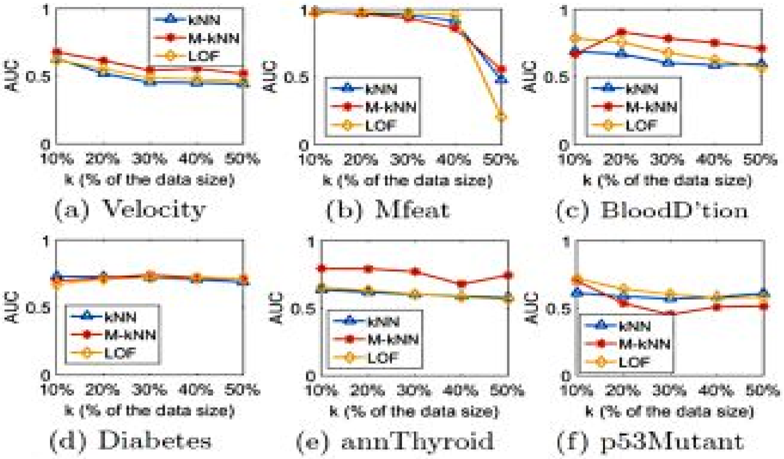


Figure 8: The ability to detect local anomalies in the dense cluster. Contour of the scores of k-nearest neigbour anomaly detectors using ℓ_p and m_e, with $k = 100$ on a synthetic dataset. The lighter the colour, the higher the anomaly score.

Table 6: Properties of benchmark datasets

Dataset	Size	Dimensions	% Anomaly
Velocity	229	20	34.06%
Mfeat	410	649	2.44%
BloodDonation	604	4	5.63%
Diabetes	768	8	34.9%
annThyroid	7200	6	7.42%
p53Mutant	10387	5408	0.51%



Findings Multi-Label Classification

Table 8: M-MLkNN versus MLkNN on hamming loss, ranking loss, coverage, one error and average precision. Each result is an average over 10 trials of 10-fold cross-validation and its standard error. For the first four performance measures, the lower the better; for the last one, the higher the better.

		HammingLoss	RankingLoss	Coverage	OneError	AveragePrecision
Birds	MLkNN	0.051 ± 0.001	0.298 ± 0.004	3.507 ± 0.084	0.713 ± 0.017	0.392 ± 0.010
	M-MLkNN	$\mathbf{0.046} \pm 0.001$	$\mathbf{0.167} \pm 0.007$	$\mathbf{2.008} \pm 0.067$	$\mathbf{0.473} \pm 0.006$	$\mathbf{0.600} \pm 0.005$
CAL500	MLkNN	0.139 ± 0.001	0.187 ± 0.002	132.467 ± 0.489	0.131 ± 0.005	0.489 ± 0.002
	M-MLkNN	0.139 ± 0.001	0.187 ± 0.001	131.731 ± 0.538	0.125 ± 0.005	0.489 ± 0.001
Emotions	MLkNN	0.269 ± 0.007	0.278 ± 0.011	2.358 ± 0.071	0.419 ± 0.013	0.692 ± 0.011
	M-MLkNN	$\mathbf{0.208} \pm 0.004$	$\mathbf{0.180} \pm 0.005$	$\mathbf{1.857} \pm 0.025$	$\mathbf{0.309} \pm 0.008$	$\mathbf{0.776} \pm 0.006$
Enron	MLkNN	0.053 ± 0.000	0.099 ± 0.001	13.850 ± 0.095	0.340 ± 0.013	0.604 ± 0.005
	M-MLkNN	$\mathbf{0.052} \pm 0.000$	$\mathbf{0.094} \pm 0.001$	$\mathbf{13.477} \pm 0.103$	$\mathbf{0.288} \pm 0.008$	$\mathbf{0.640} \pm 0.004$
Scene	MLkNN	0.175 ± 0.002	0.198 ± 0.003	1.065 ± 0.023	0.338 ± 0.008	0.774 ± 0.003
	M-MLkNN	$\mathbf{0.168} \pm 0.002$	$\mathbf{0.175} \pm 0.003$	$\mathbf{0.978} \pm 0.021$	$\mathbf{0.310} \pm 0.008$	$\mathbf{0.794} \pm 0.004$

Evaluation of Runtime

Table 10: Runtime of the dissimilarity matrix calculation for the three dissimilarities (in seconds).

Data set (Data size) (Dimension)	Segment (2310) (19)	annThyroid (7200) (6)	Pendig (10992) (16)	p53Mut (10387) (5408)
ℓ_p	5	42	110	8182
m_e	31	259	600	548
SNN	26	243	573	9141

For data with low dimension l(distance-based dissimilarity) does better than mass-based dissimilarity.

For data with higher dimension it is opposite.

Big Question

What is mass-based dissimilarity?

Definition 1. $R(x,y|H;D)$ is the smallest local region covering x and y wrt H and D is defined as:

$$R(x,y|H;D) = \underset{r \subset H \ s.t. \{x,y\} \in r}{\arg\min} \sum_{z \in D} \mathbb{1}(z \in r) \tag{1}$$

where $\mathbb{1}(\cdot)$ is an indicator function.

Definition 2. Mass-based dissimilarity of x and y wrt D and F is defined as the expected probability of $R(x,y|H;D)$:

$$m(x,y|D,F) = E_{\mathcal{H}(D)}[P_F(R(x,y|H;D))] \tag{2}$$

where $P_F(\cdot)$ is the probability wrt F; and the expectation is taken over all models in $\mathcal{H}(D)$.

In practice, the mass-based dissimilarity would be estimated from a finite number of models $H_i \in \mathcal{H}(D), i = 1, \ldots, t$ as follows:

$$m_e(x,y|D) = \frac{1}{t}\sum_{i=1}^{t} \tilde{P}(R(x,y|H_i;D)) \tag{3}$$

Conclusion

Based on the different measures, Mass-based dissimilarity is a better option.

Based on the run-time complexity, mass-based dissimilarity works better than the distance based dissimilarity

Persistence in regional voting patterns in Turkey during a period of major political realignment

Authors: Ali T. Akarca and Cem Başlevent

Sarah Sutto-Plunz
CSC 390

Motivation

- Explore regional differences in voting patterns in Turkey
- Present these differences
- Look into the socio-economic characteristics on which they are built

Methods

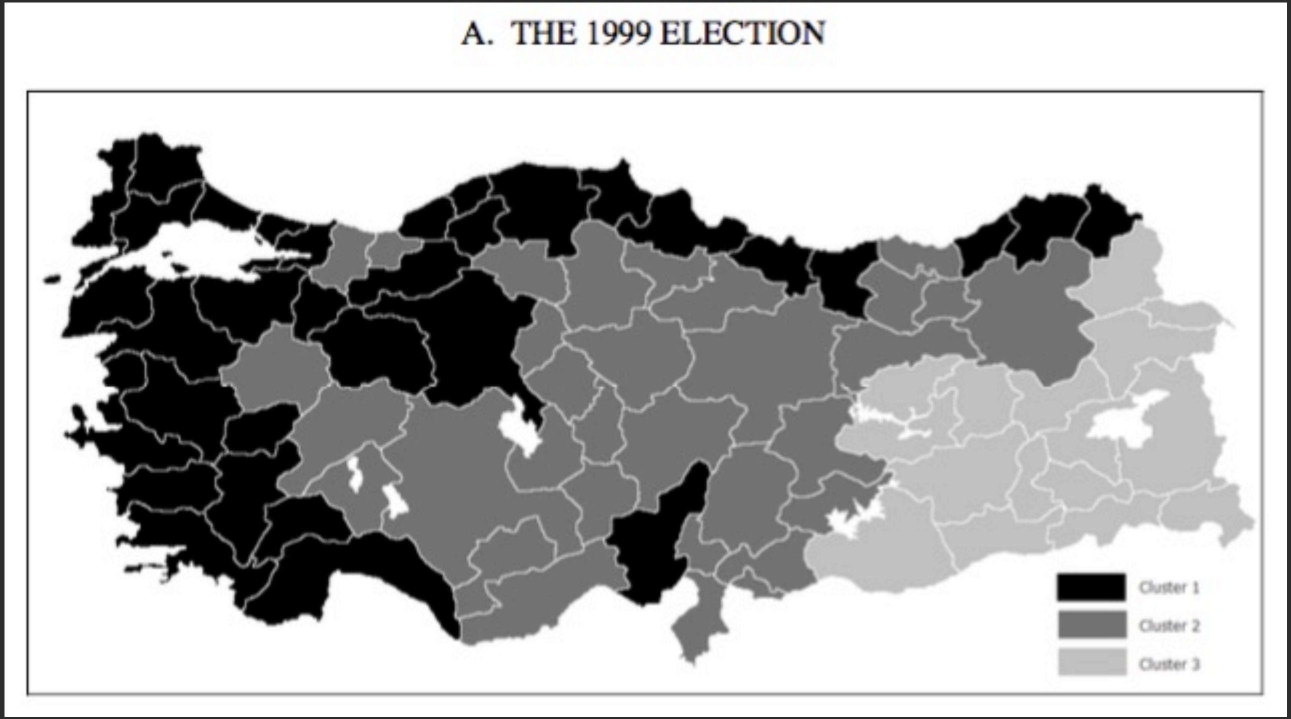
- K-means algorithm

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2$$

- 81 provinces classified according to vote shares of the major parties and independent candidates

- Done for each k for each election between 1999-2009

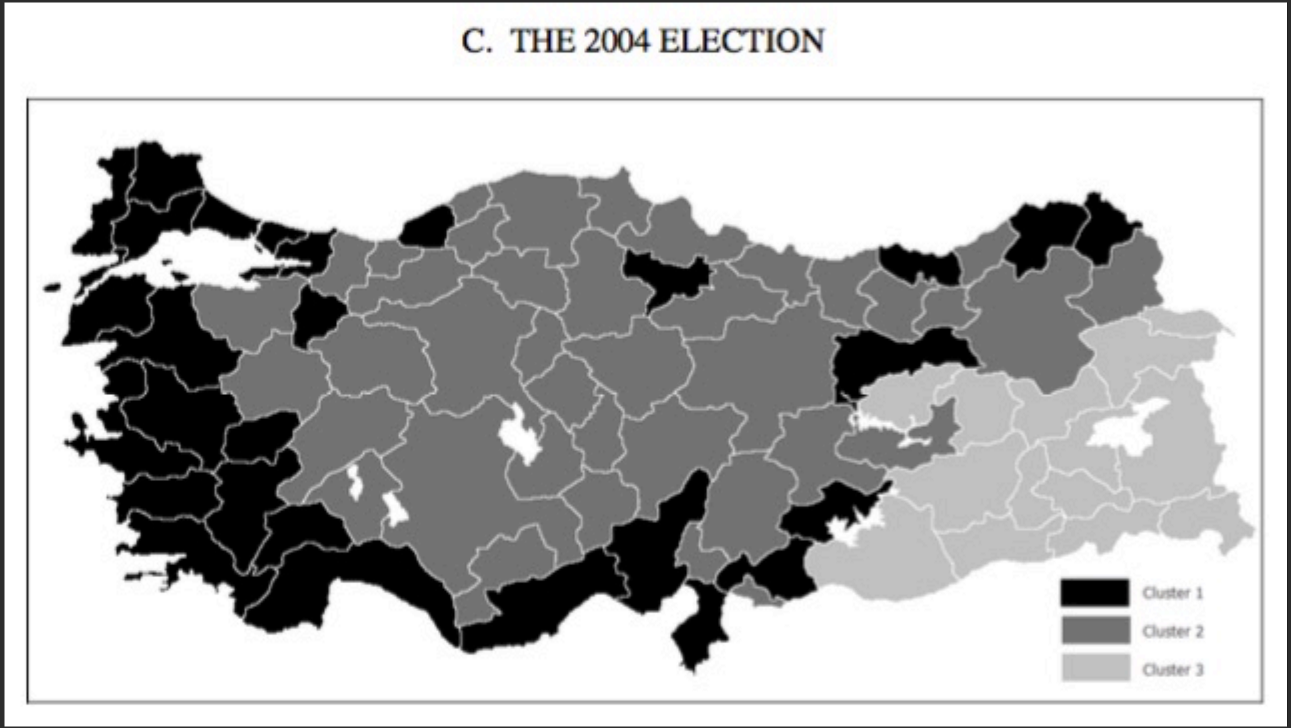
Results



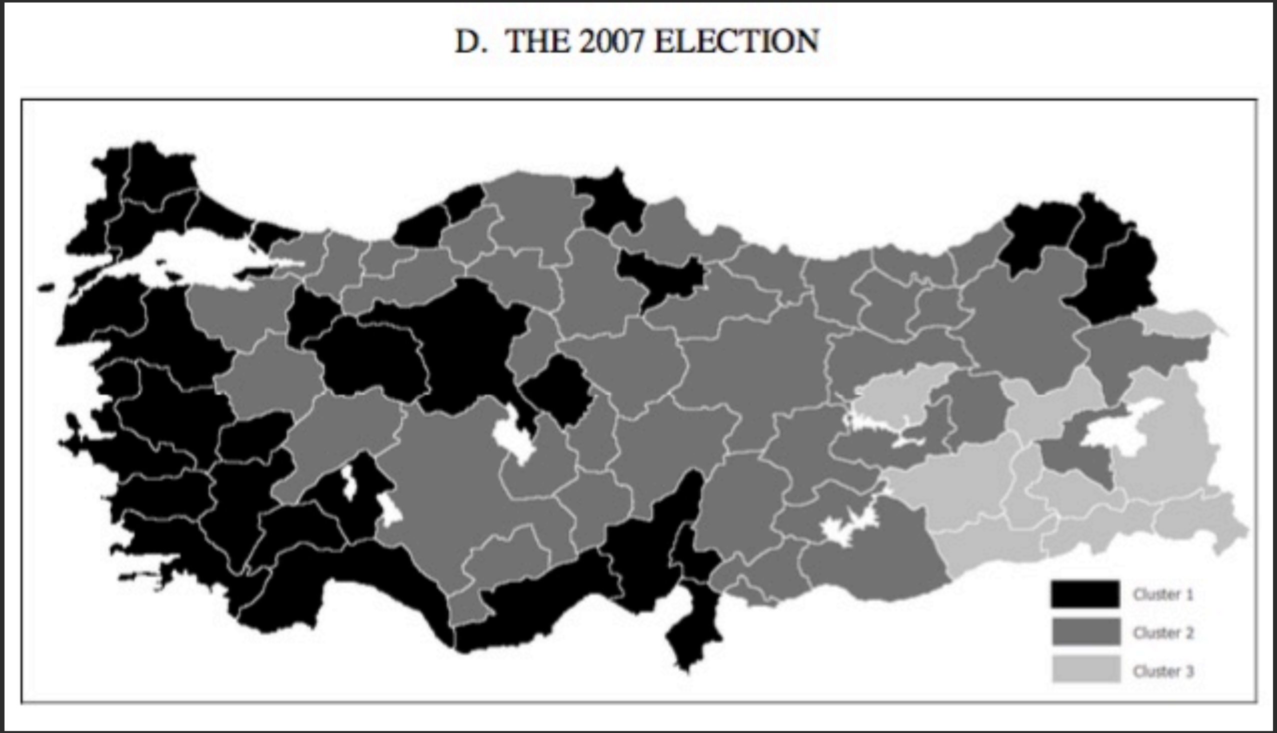
Results

B. THE 2002 ELECTION

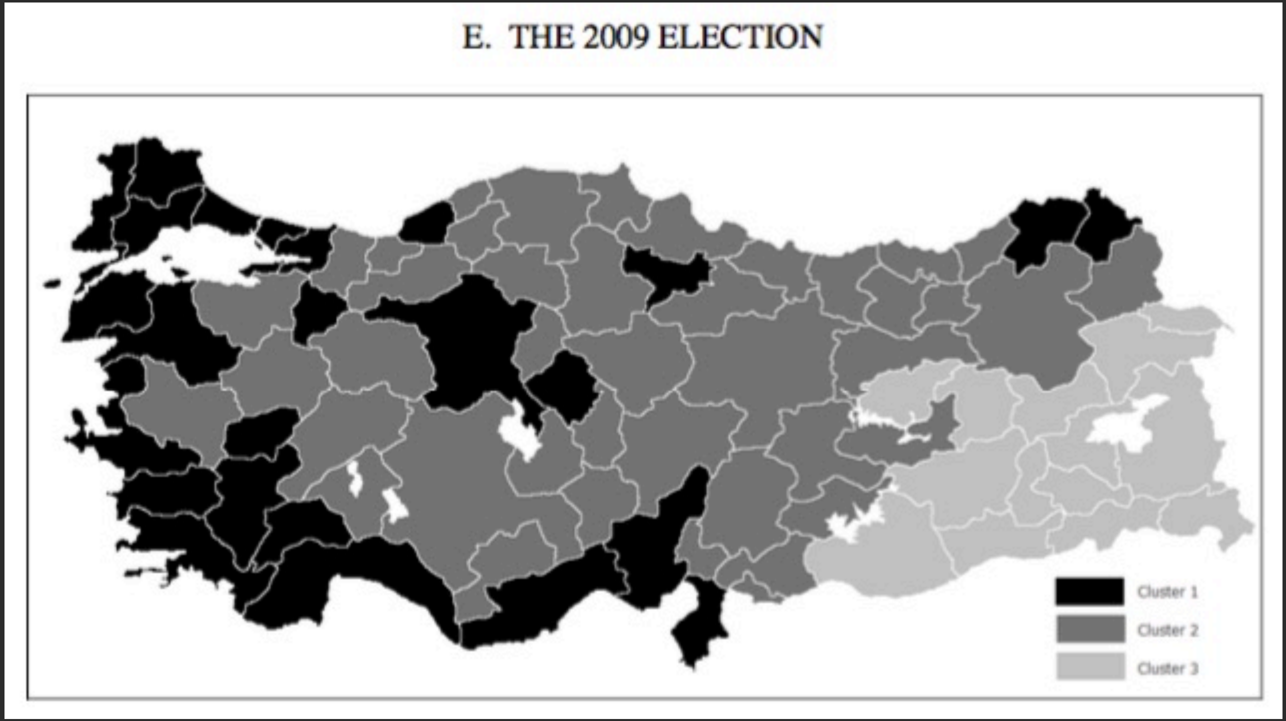
Results



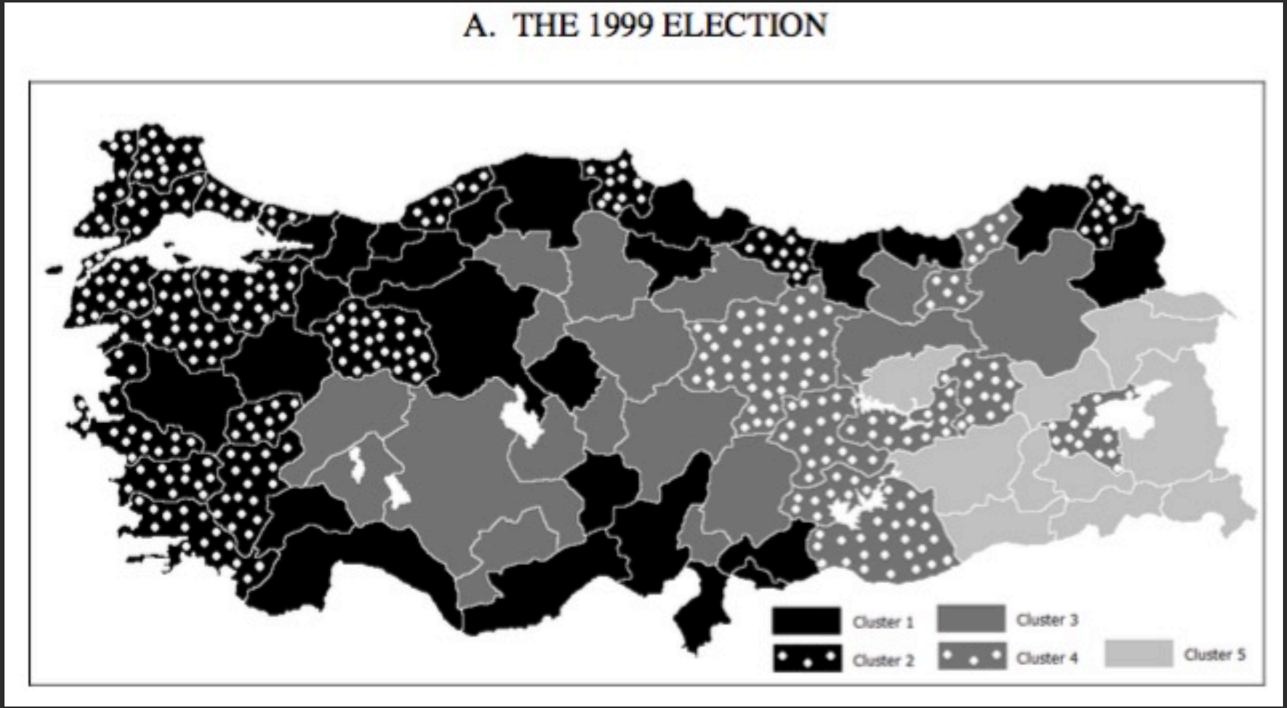
Results



Results

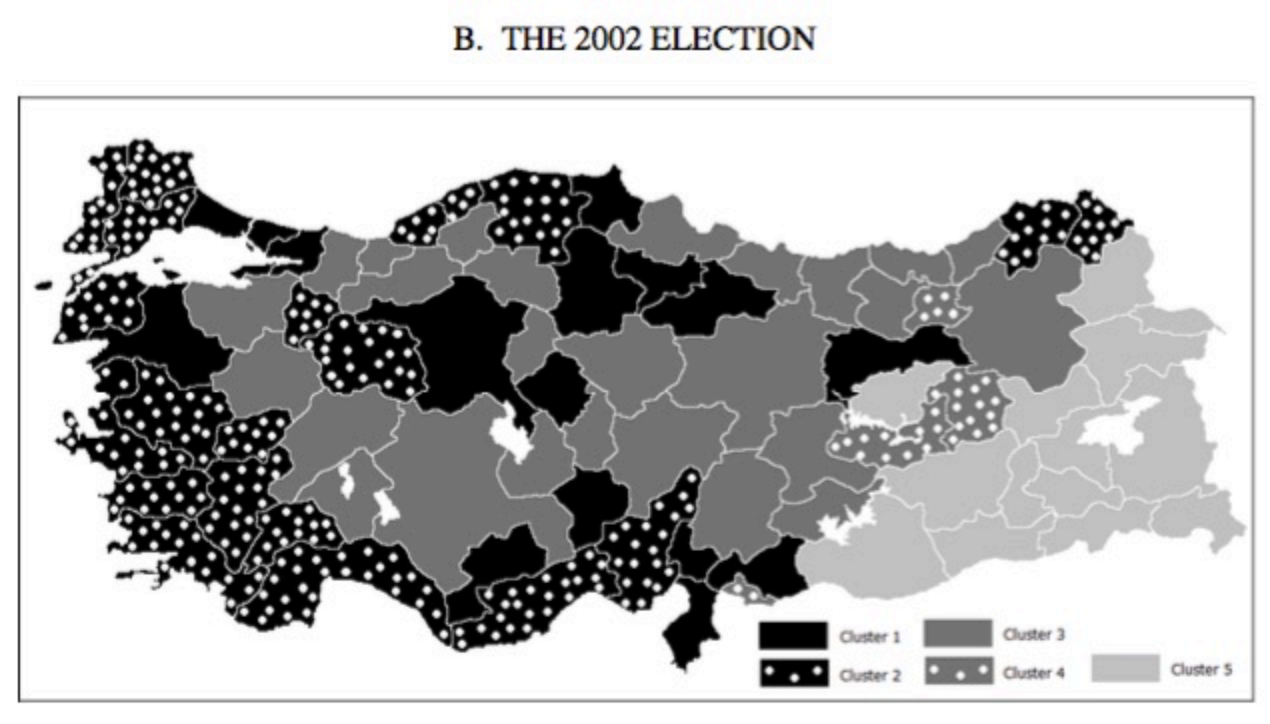


Results

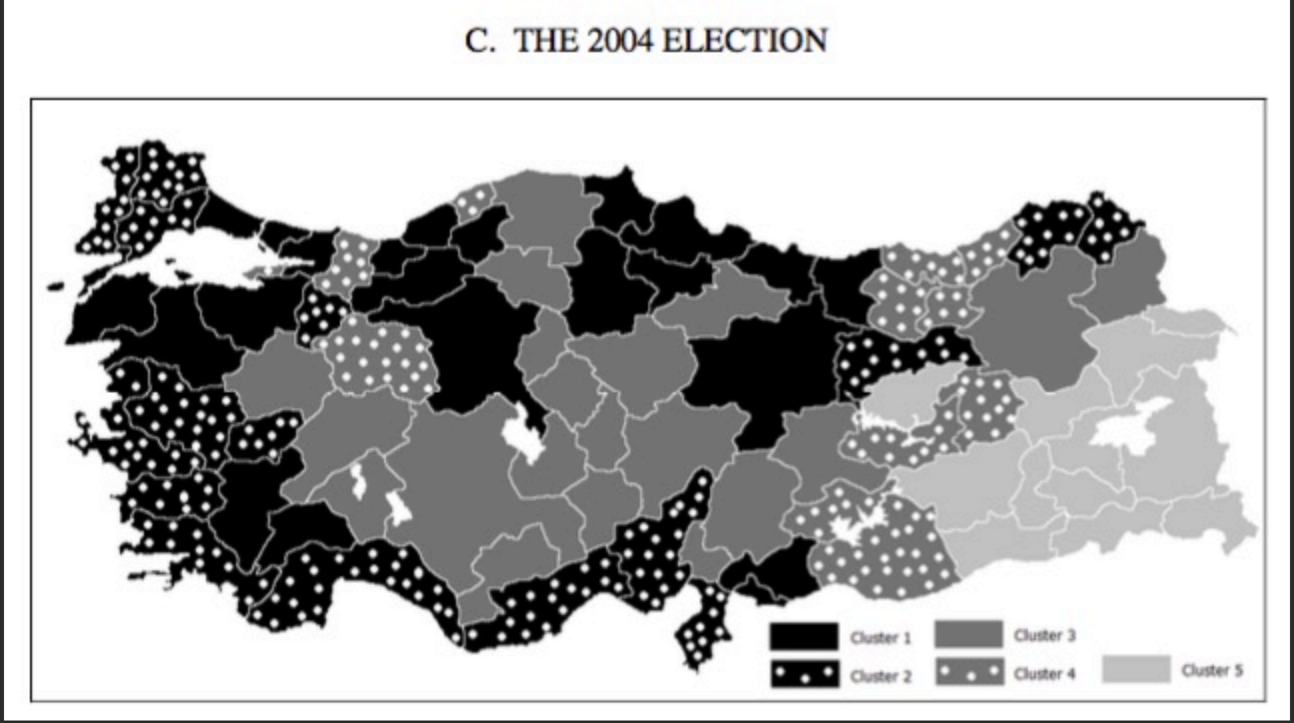


A. THE 1999 ELECTION
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5

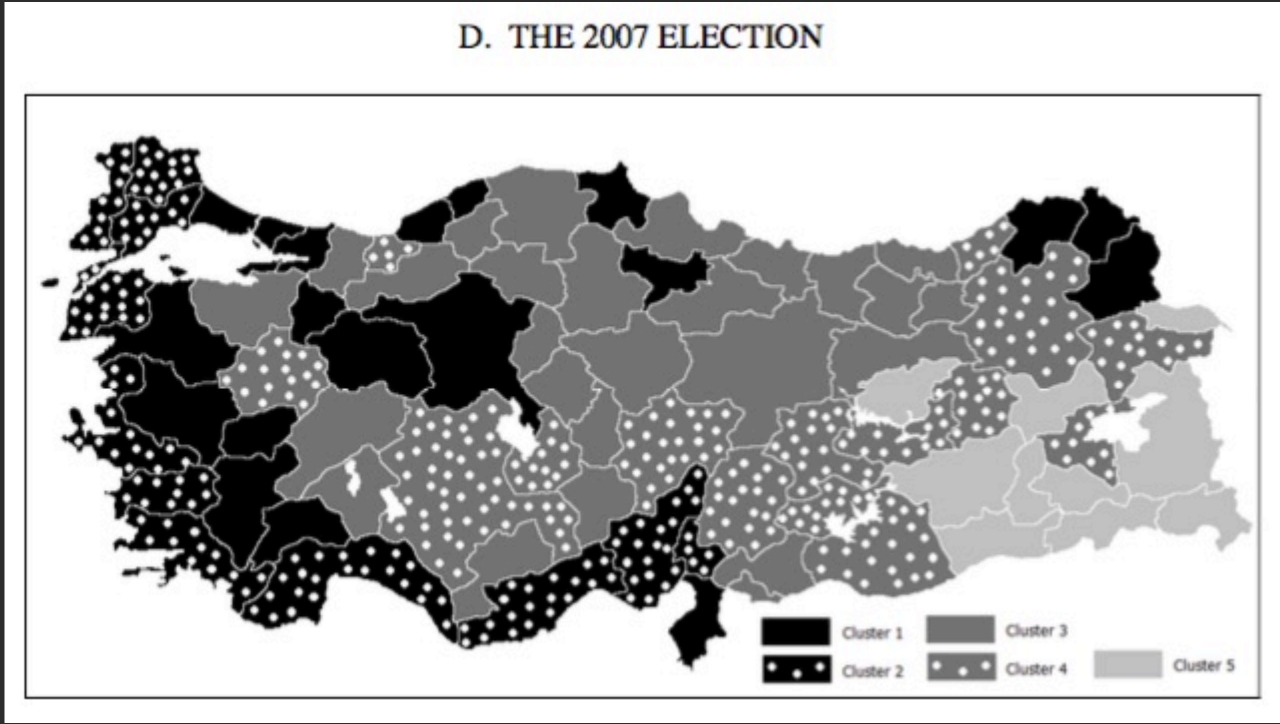
Results


B. THE 2002 ELECTION
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5

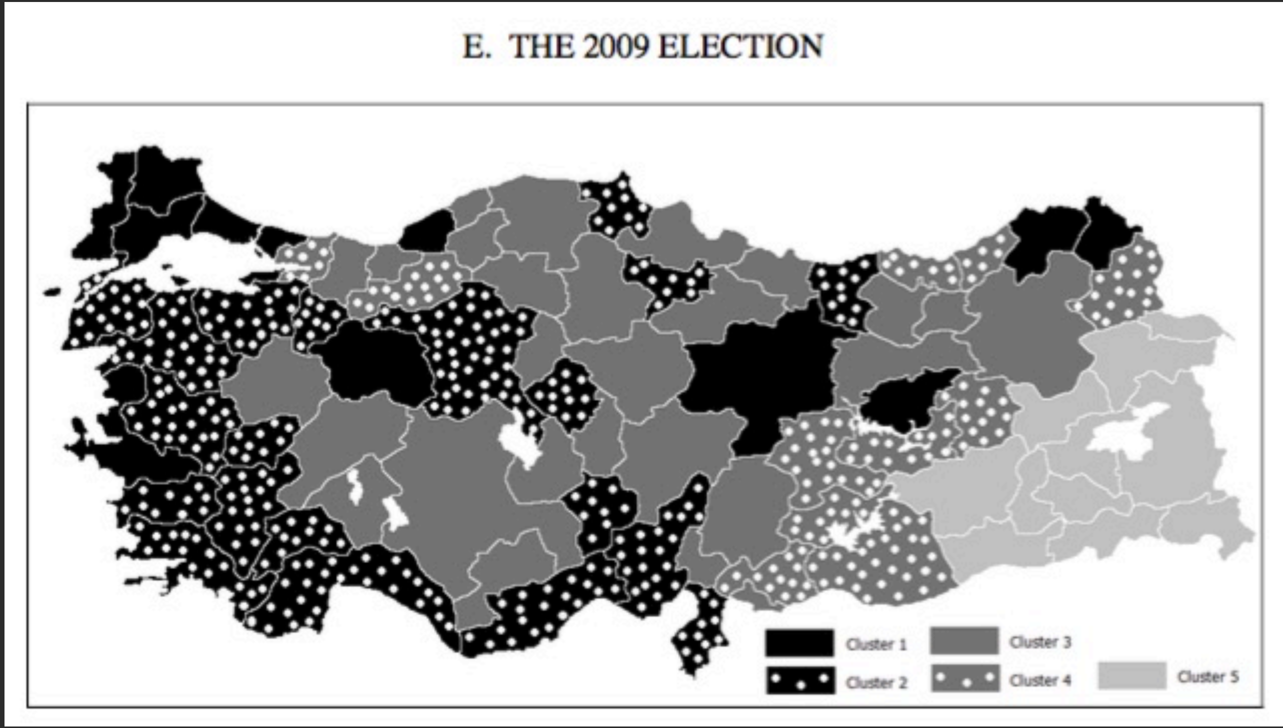
Results


C. THE 2004 ELECTION
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5

Results

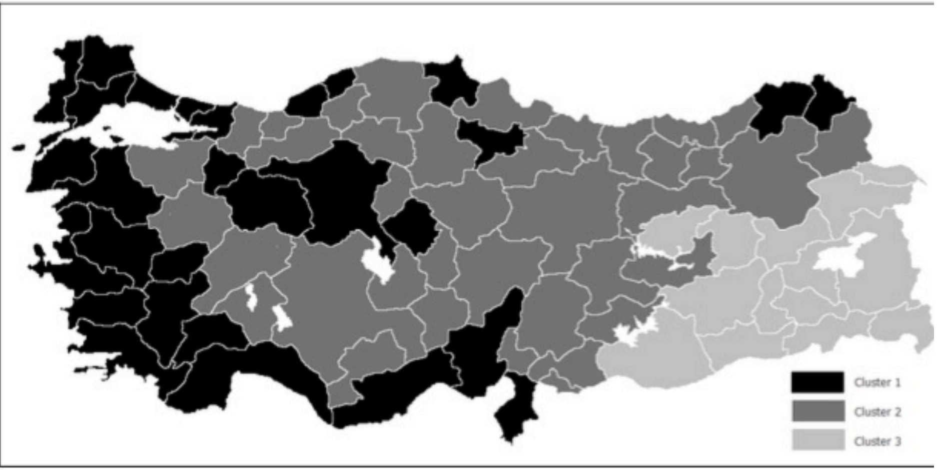

D. THE 2007 ELECTION
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5

Results

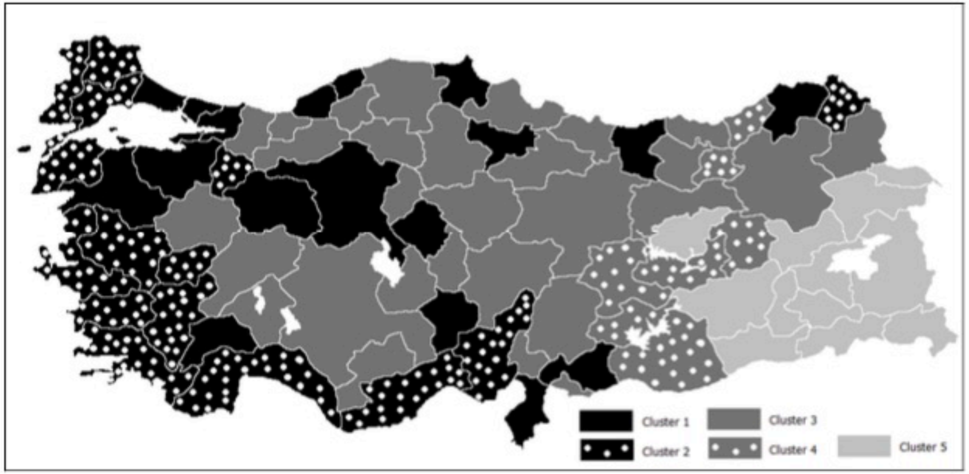

E. THE 2009 ELECTION
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5

Results

A. K=3



B. K=5



Conclusions

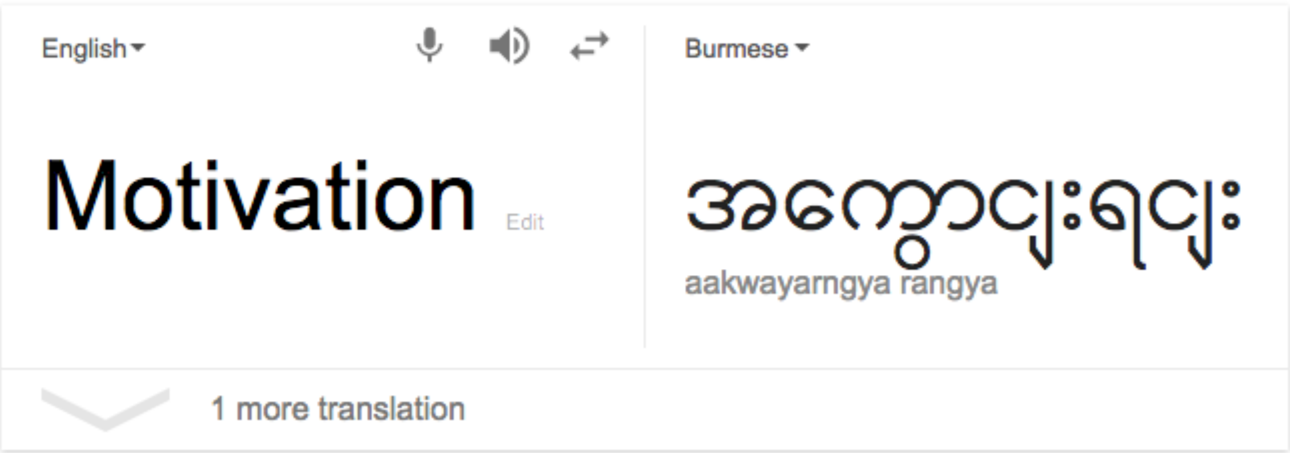
- Provincial make-up of the clusters remains essentially unchanged during a decade of political realignment
- Cultural, ethnic, socio-economic and historical ties have more lasting effects

- Future work: How does this relate to the US?

Google's Neural Machine Translation system: bridging the gap between human and machine translation

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, Jeffrey Dean

Presenter: Maria Xu
CSC 390
Nov 10



Google Neural Machine Translation (GNMT)

Key Terms

LSTM Network

Encoder

Decoder

Attention Mechanism

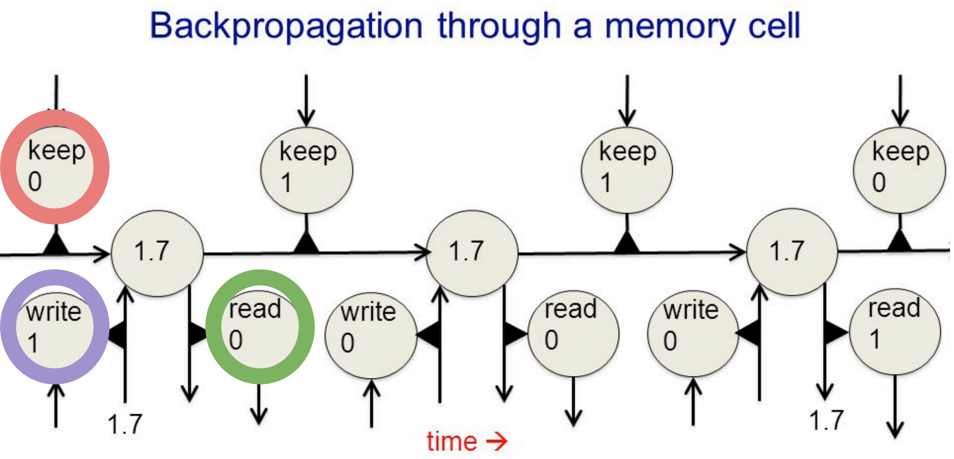
Parallelism

Residual Connections

BLEU score

Long Short-Term Memory Network

A recurrent neural network architecture



LSTM Network

Encoder

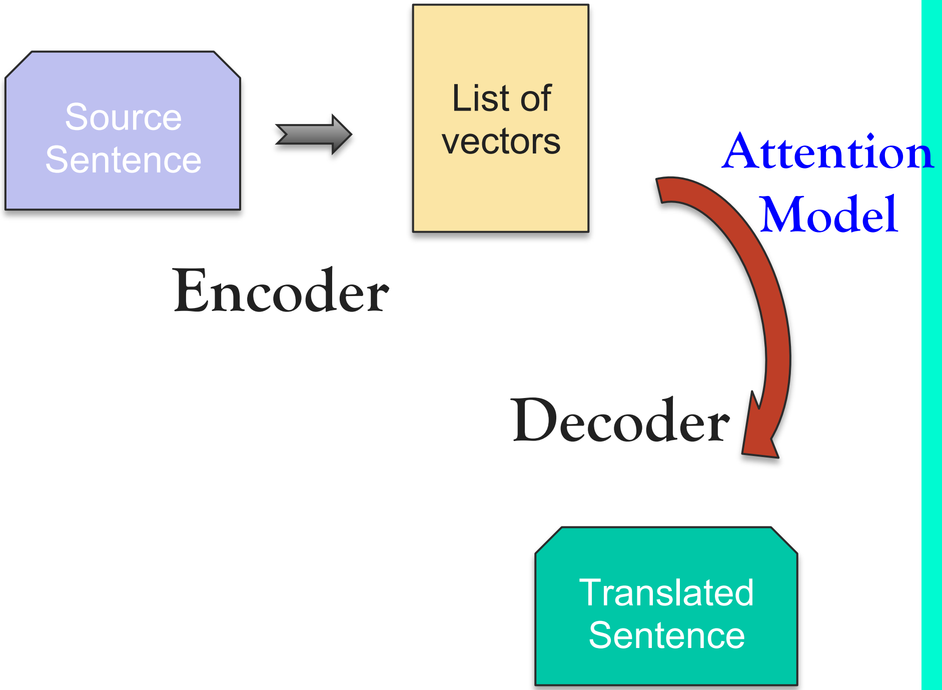
Decoder

Attention Mechanism

Parallelism

Residual Connections

BLEU score



LSTM Network

Encoder

Decoder

Attention Mechanism

Parallelism

Residual Connections

BLEU score

Parallelism

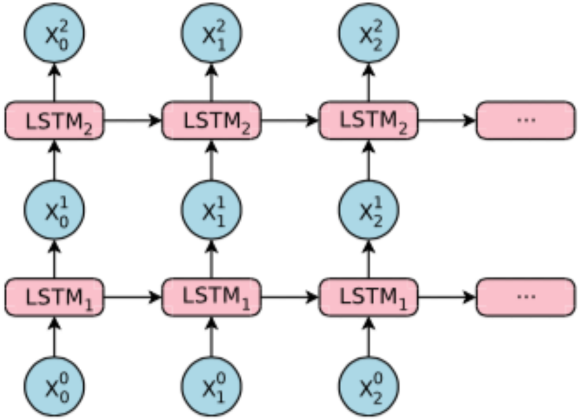
GOAL: decrease training time, improve efficiency

Data parallelism:

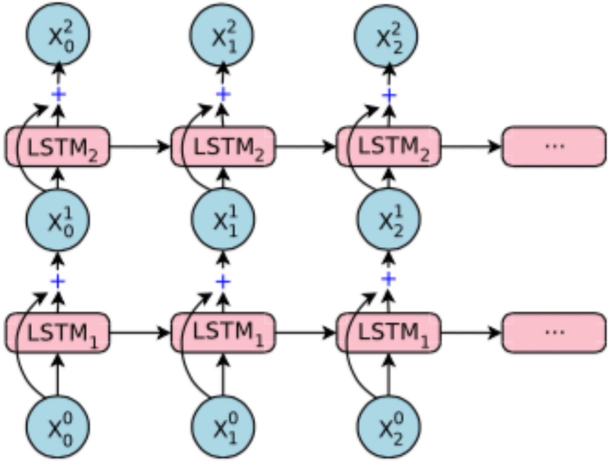
- train 10 model replicas concurrently
- each replica asynchronously updates parameters

Model parallelism:

- improve the speed of the computation on each replica
- partitioned and placed on multiple GPUs



Normal stacked LSTM



Stacked LSTM with residual connections

LSTM Network

Encoder

Decoder

Attention Mechanism

Parallelism

Residual Connections

BLEU score

BLEU score

- BilinguaL Evaluation Understudy score.

- It is an algorithm for evaluating the quality of text which has been machine-translated from one natural language to another.

LSTM Network

Encoder

Decoder

Attention Mechanism

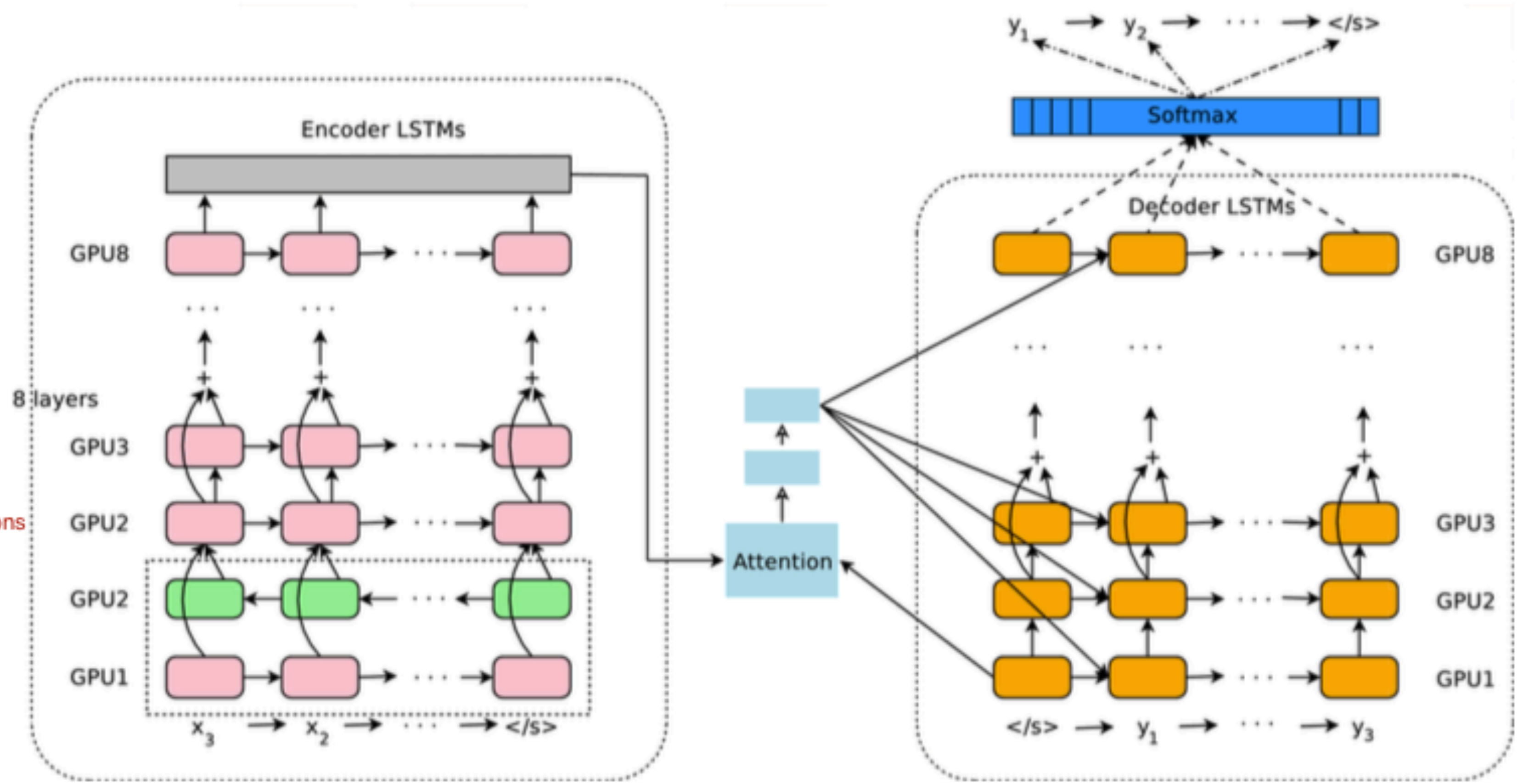
Parallelism

Residual Connections

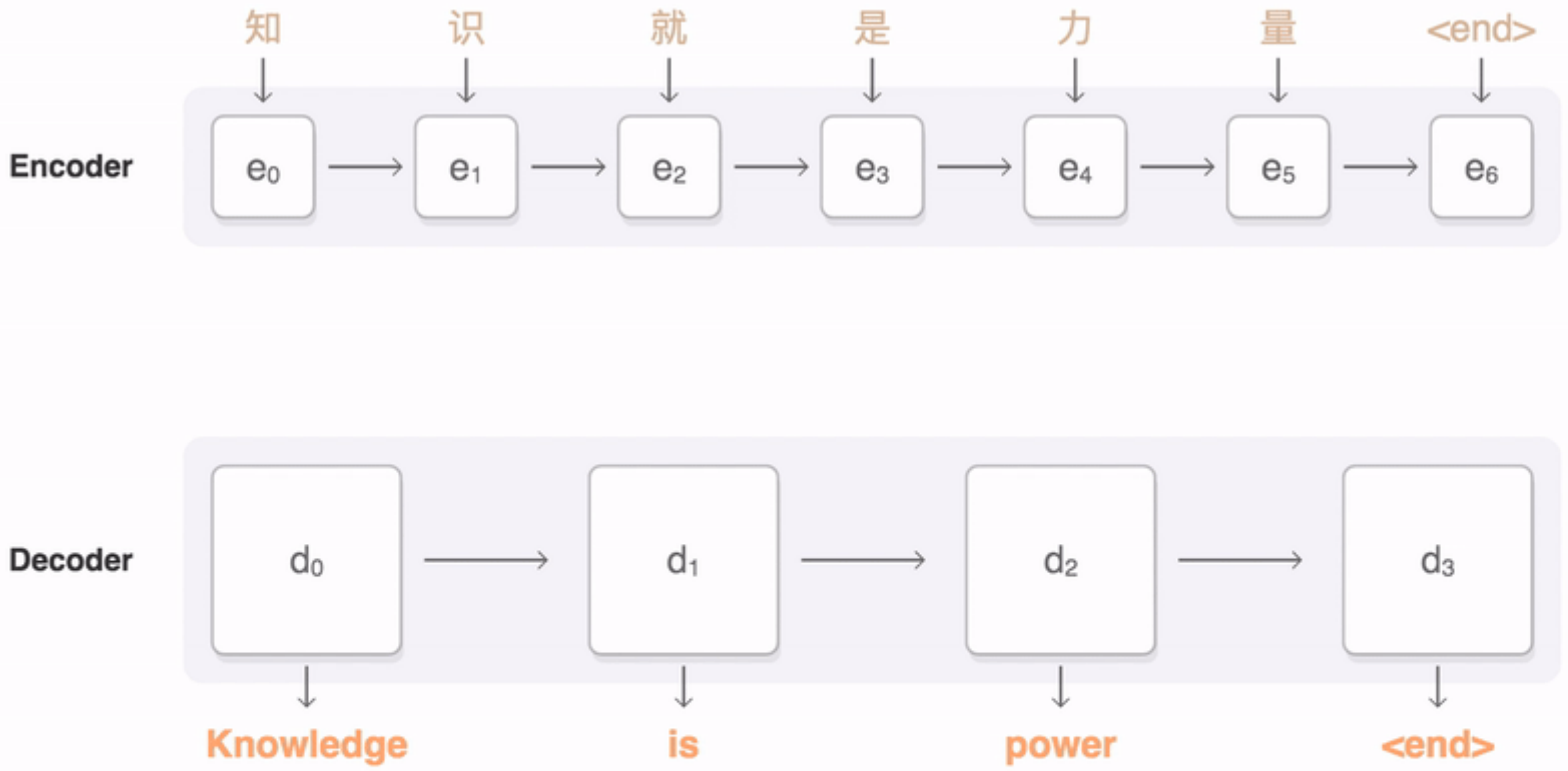
BLEU score

GNMT Model Architecture

softmax: non-linear variant of multinomial logistic regression



residual connections


知
识
就
是
力
量
<end>
Encoder
e_0
e_1
e_2
e_3
e_4
e_5
e_6
Decoder
d_0
d_1
d_2
d_3
Knowledge
is
power
<end>

Training the data

Wordpiece Model (WPM)

- **Word:** Jet makers feud over seat width with big orders at stake
- **wordpieces:** _J et _makers _fe ud _over _seat _width _with _big _orders _at _stake

~ Mixed word/ Character model

Training Criteria: Maximum –likelihood training

standard maximum-likelihood training (ML)

$$\mathcal{O}_{\mathrm{ML}}(\boldsymbol{\theta}) = \sum_{i=1}^{N} \log P_\theta(Y^{*(i)} \mid X^{(i)}) \; .$$

$$\mathcal{O}_{\mathrm{RL}}(\boldsymbol{\theta}) = \sum_{i=1}^{N} \sum_{Y \in \mathcal{Y}} P_\theta(Y \mid X^{(i)}) \; r(Y, Y^{*(i)}).$$

expected reward objective (RL)

$$\mathcal{O}_{\mathrm{Mixed}}(\boldsymbol{\theta}) = \alpha * \mathcal{O}_{\mathrm{ML}}(\boldsymbol{\theta}) + \mathcal{O}_{\mathrm{RL}}(\boldsymbol{\theta})$$

Decoder

Beam search

Find the sequence that maximize a score function.

Refinements

Length normalization (α)

Coverage penalty (β)

En -> French

BLEU		α					
		0.0	0.2	0.4	0.6	0.8	1.0
β	0.0	30.3	30.7	30.9	31.1	31.2	31.1
	0.2	31.4	31.4	31.4	31.3	30.8	30.3
	0.4	31.4	31.4	31.4	31.1	30.5	29.6
	0.6	31.4	31.4	31.3	30.9	30.1	28.9
	0.8	31.4	31.4	31.2	30.8	29.8	28.1
	1.0	31.4	31.3	31.2	30.6	29.4	27.2

Result

Evaluation after Maximum Likelihood Training

Table 4: Single model results on WMT En→Fr (newstest2014)

Model	BLEU	Decoding time per sentence (s)
Word	37.90	0.2226
Character	38.01	1.0530
WPM-8K	38.27	0.1919
WPM-16K	37.60	0.1874
WPM-32K	38.95	0.1146
Mixed Word/Character	38.39	0.2774
PBMT [15]	37.0	
LSTM (6 layers) [30]	31.5	
LSTM (6 layers + PosUnk) [30]	33.1	
Deep-Att [43]	37.7	
Deep-Att + PosUnk [43]	39.2	

Table 5: Single model results on WMT En→De (newstest2014)

Model	BLEU	Decoding time per sentence (s)
Word (512 nodes)	22.54	0.1829
Character (512 nodes)	22.62	0.8011
WPM-8K	23.50	0.5387
WPM-16K	24.36	0.4757
WPM-32K	24.61	0.4581
Mixed Word/Character	24.17	0.2959
PBMT [6]	20.7	
RNNSearch [36]	16.5	
RNNSearch-LV [36]	16.9	
RNNSearch-LV [36]	16.9	
Deep-Att [43]	20.6	

Result

On Production Data

Full score: 6

Table 10: Mean of side-by-side scores on production data

	PBMT	GNMT	Human	Relative Improvement
English → Spanish	4.885	5.428	5.550	87%
English → French	4.932	5.295	5.496	64%
English → Chinese	4.035	4.594	4.987	58%
Spanish → English	4.872	5.187	5.372	63%
French → English	5.046	5.343	5.404	83%
Chinese → English	3.694	4.263	4.636	60%

WOW!

Question?

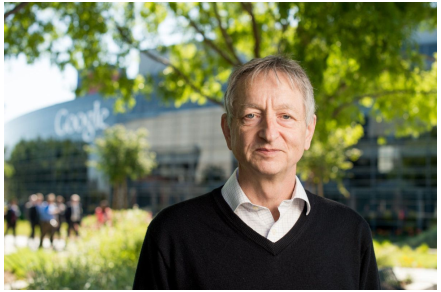
Thank you!

Visualizing and Understanding Convolution Neural Networks

Matthew D. Zeiler, Rob Fergus (2013)

Presenter: Amelia Yeoh

Background



1970s

Neural Networks

Hinton

year 1989

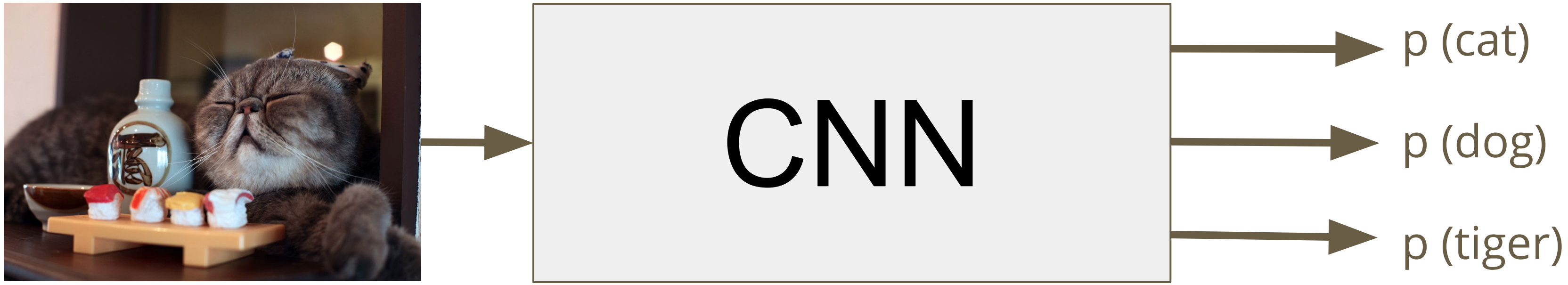
Convolutional Networks

Lecun et al. ‘89

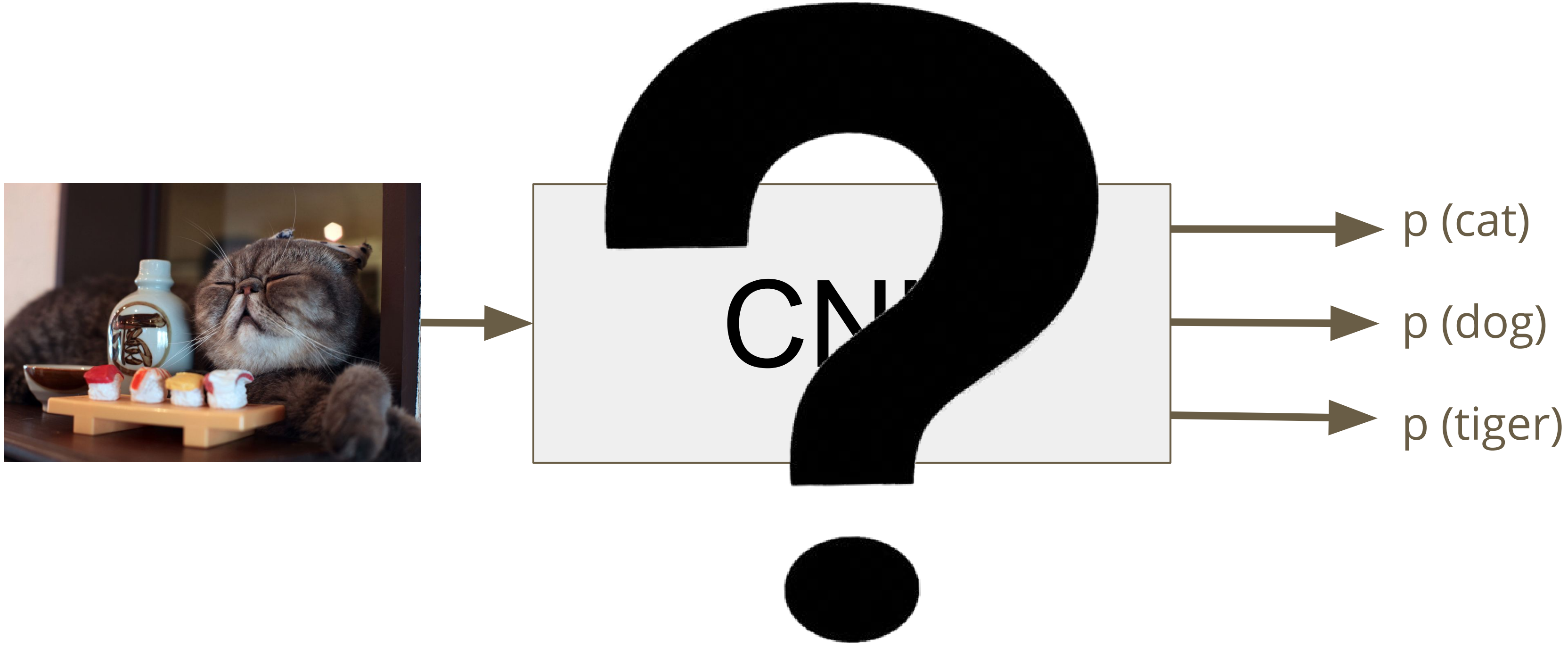
year 2009

GPU computing

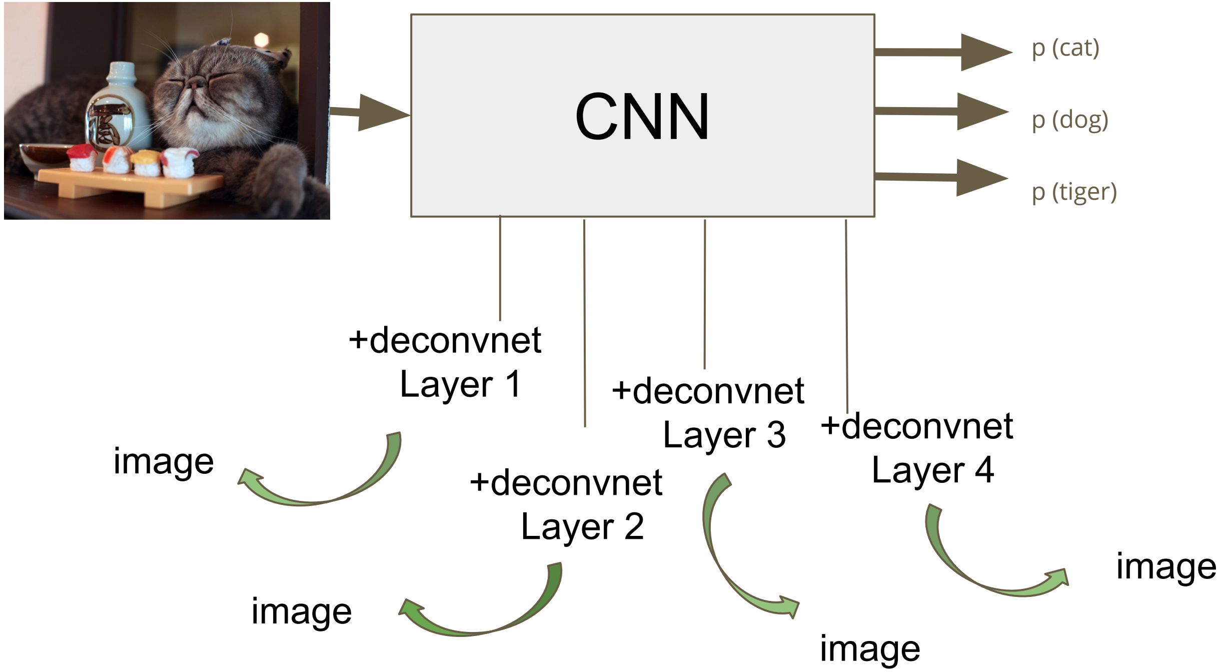
Motivation



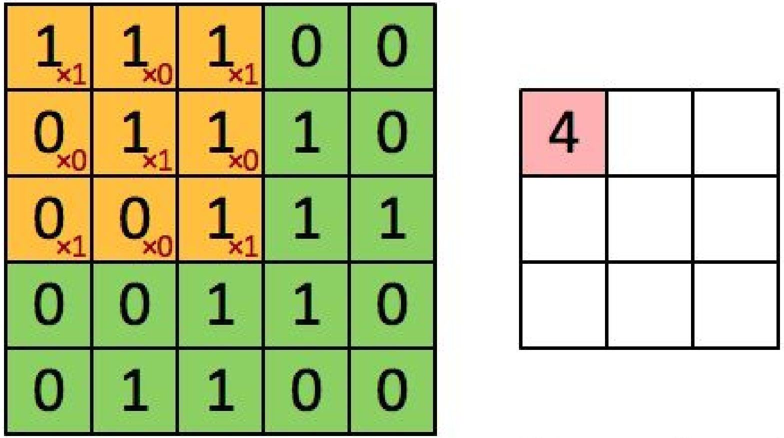
Motivation



Solution: Deconvolutional Network (deconvnet)



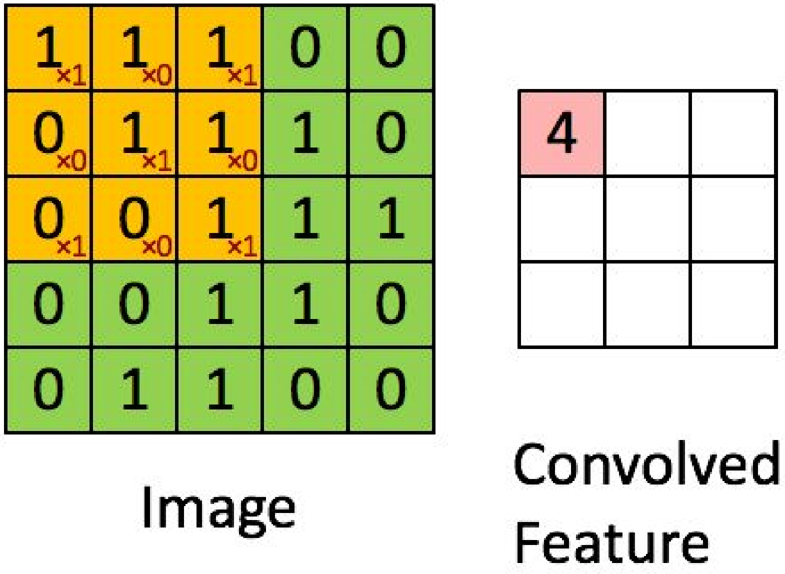
CNN Key Concepts: Convolution



Image

Convolved Feature / Feature map

CNN Key Concepts: Convolution



/ Feature map

Source

CNN Concept: Pooling

4	3	4
2	4	3
2	3	4

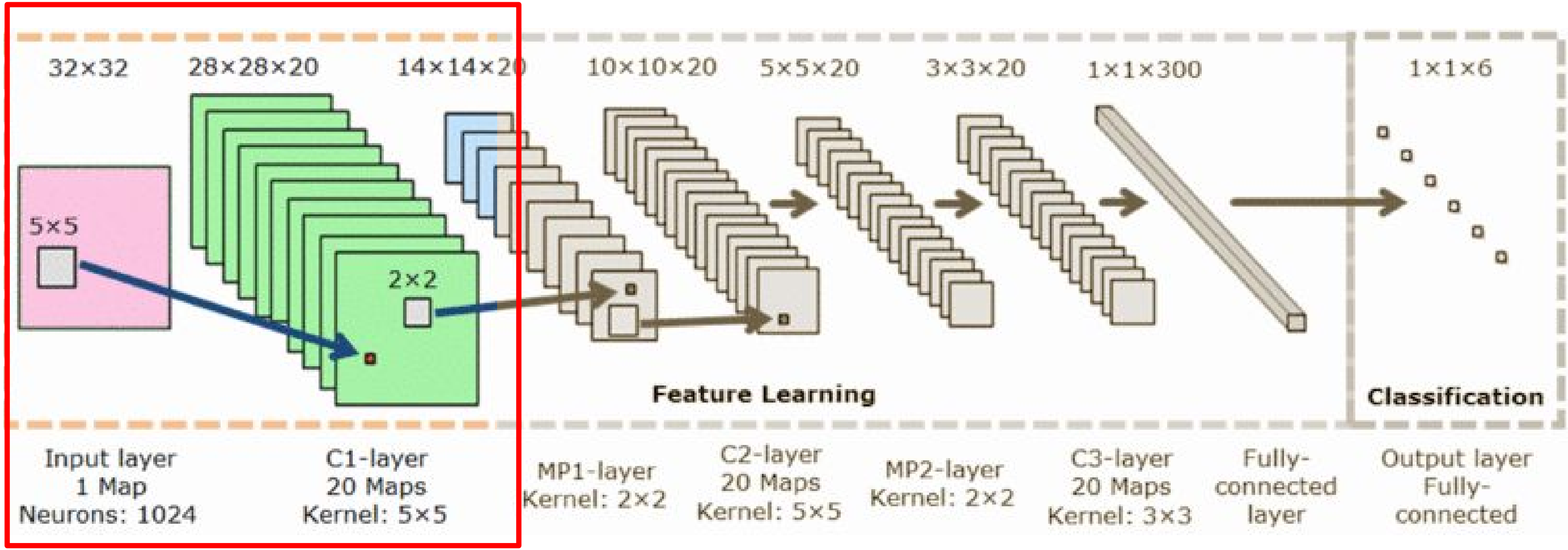
Convolved Feature / Feature map

4	4
4	4

Pooled layer

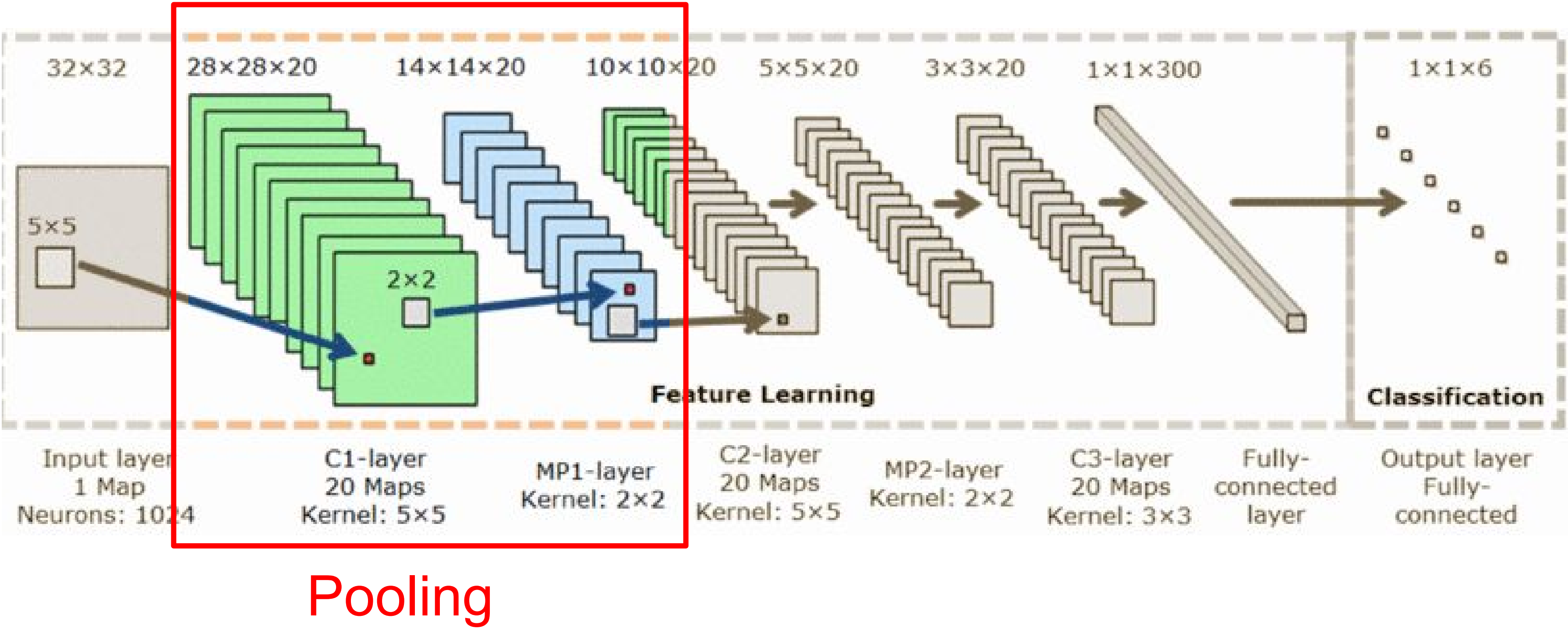
Source

CNN Overview

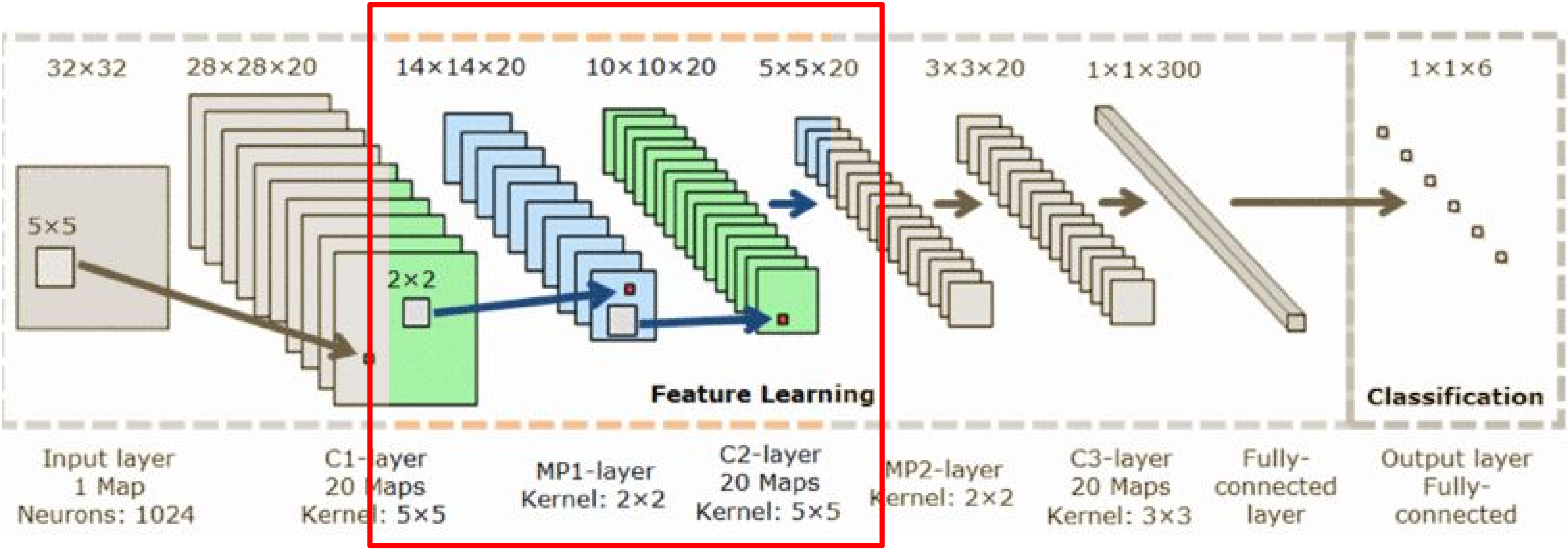


Convolution

CNN Overview

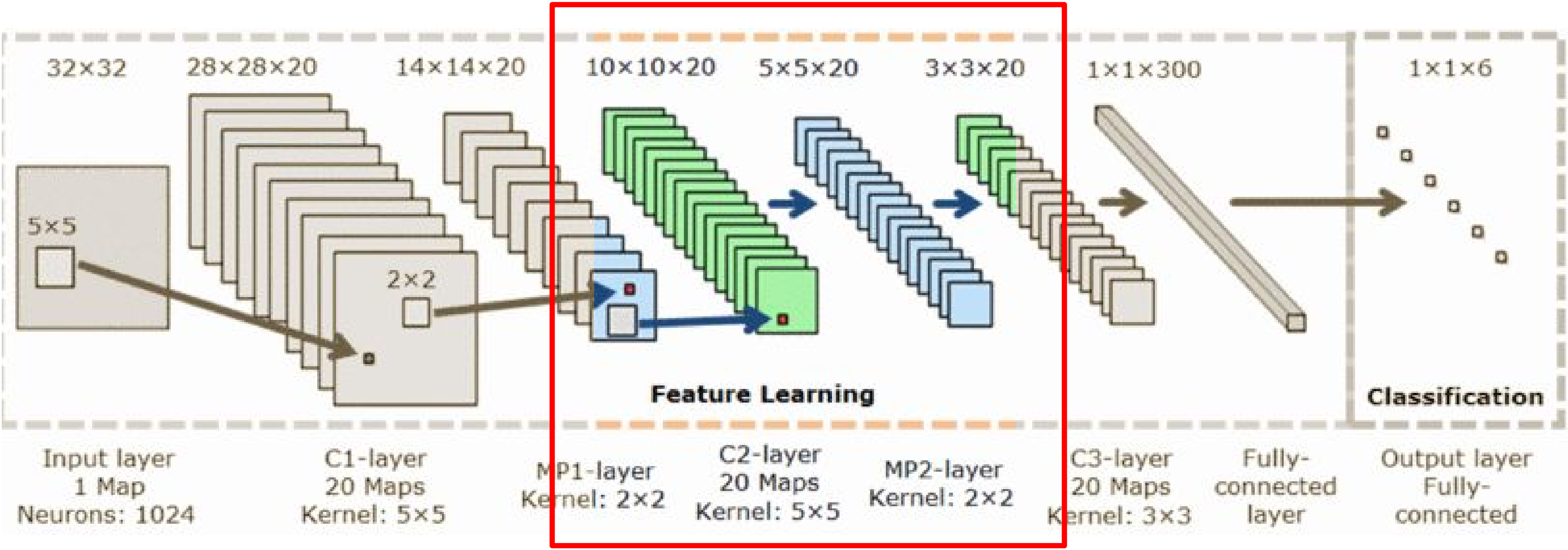


CNN Overview: Convolution



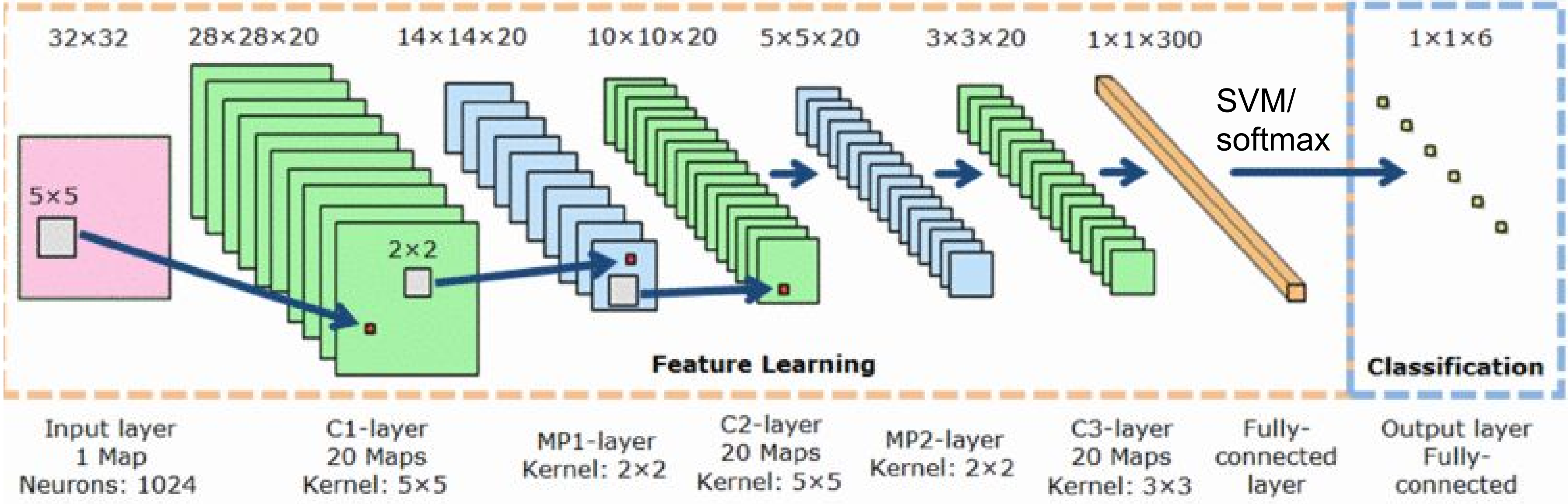
Convolution

CNN Overview: Pooling



Pooling

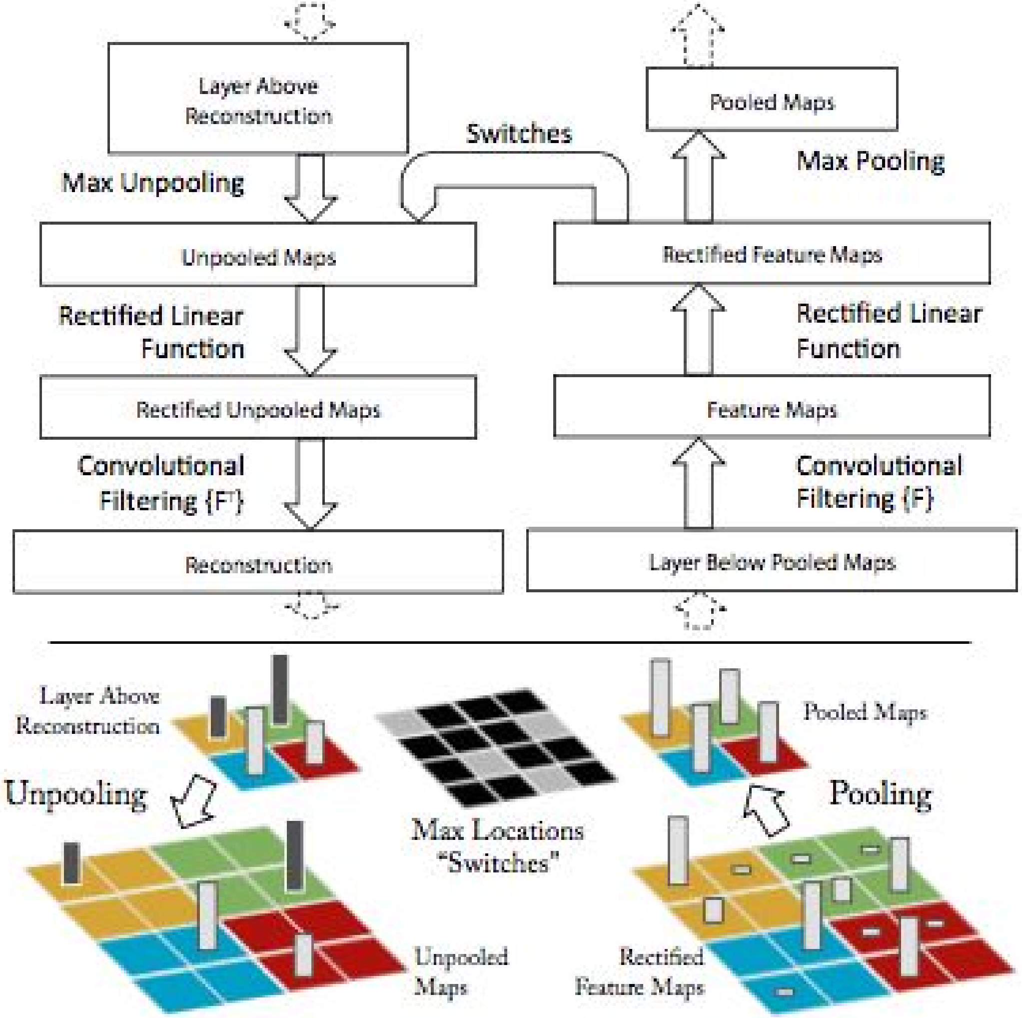
CNN Overview



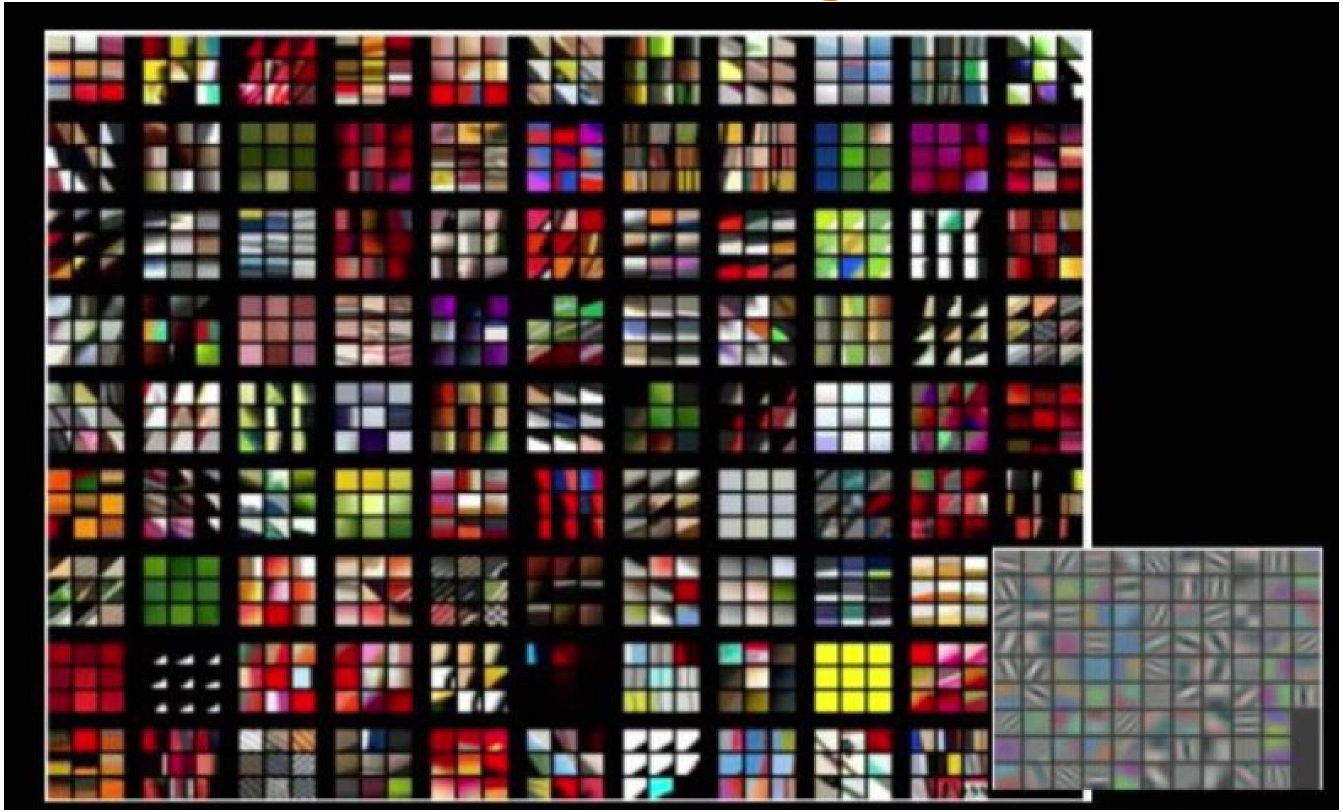
Deconvnet

- Same operations as CNN but reversed

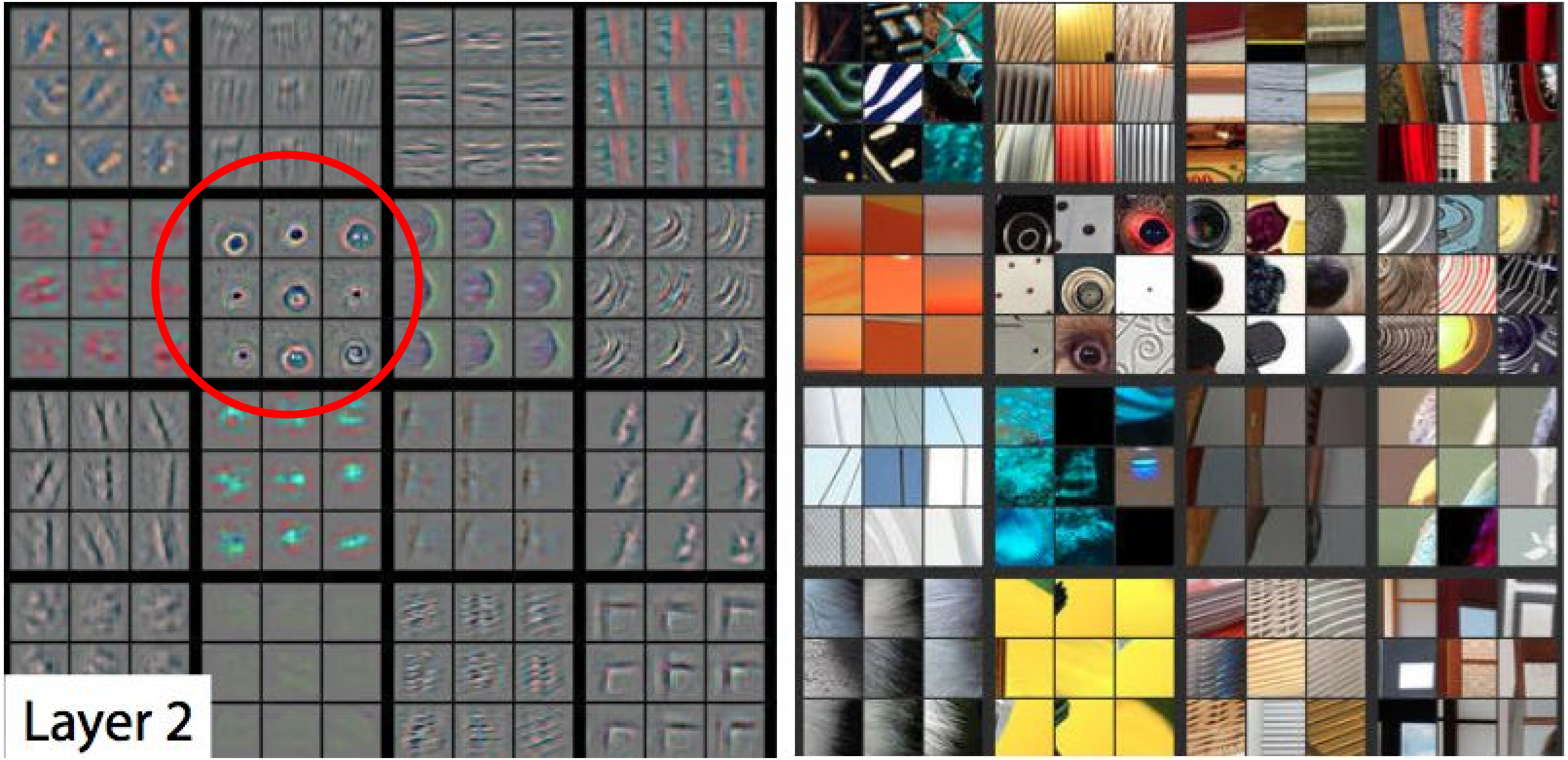
- ReLU: f(x) = max(0,x)



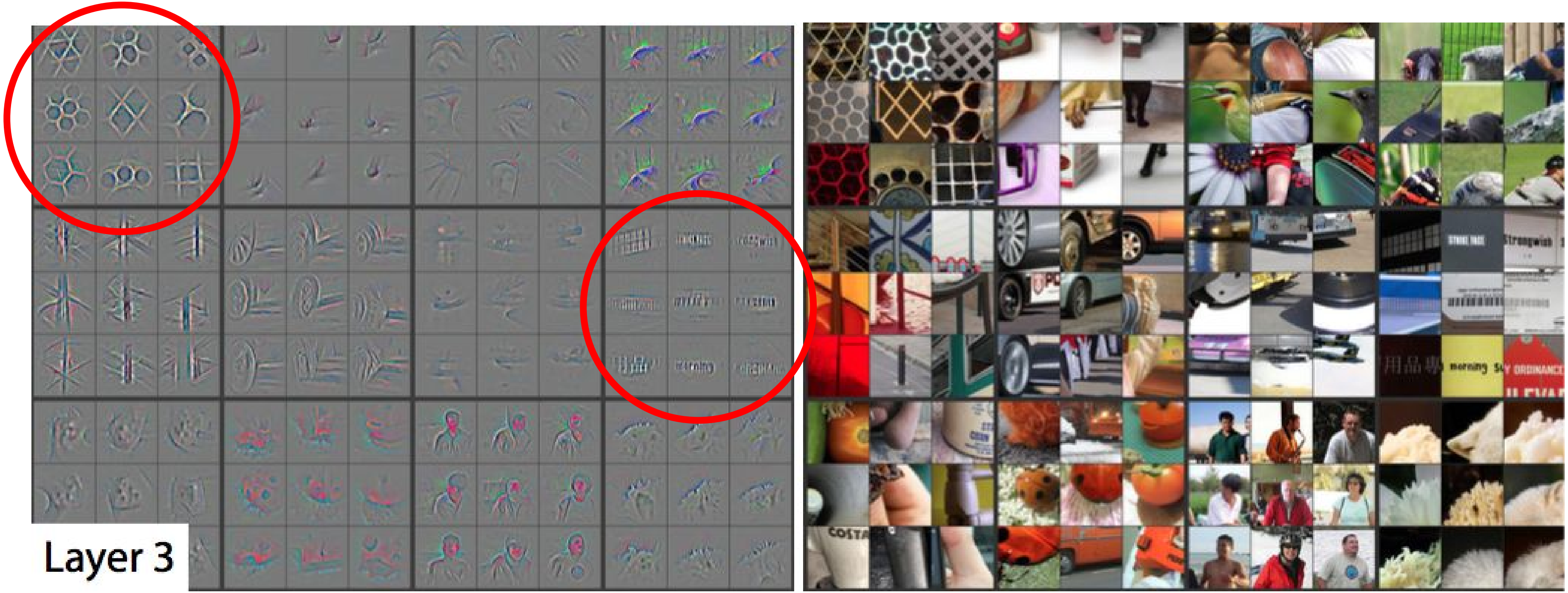
What are the models learning?



What are the models learning?

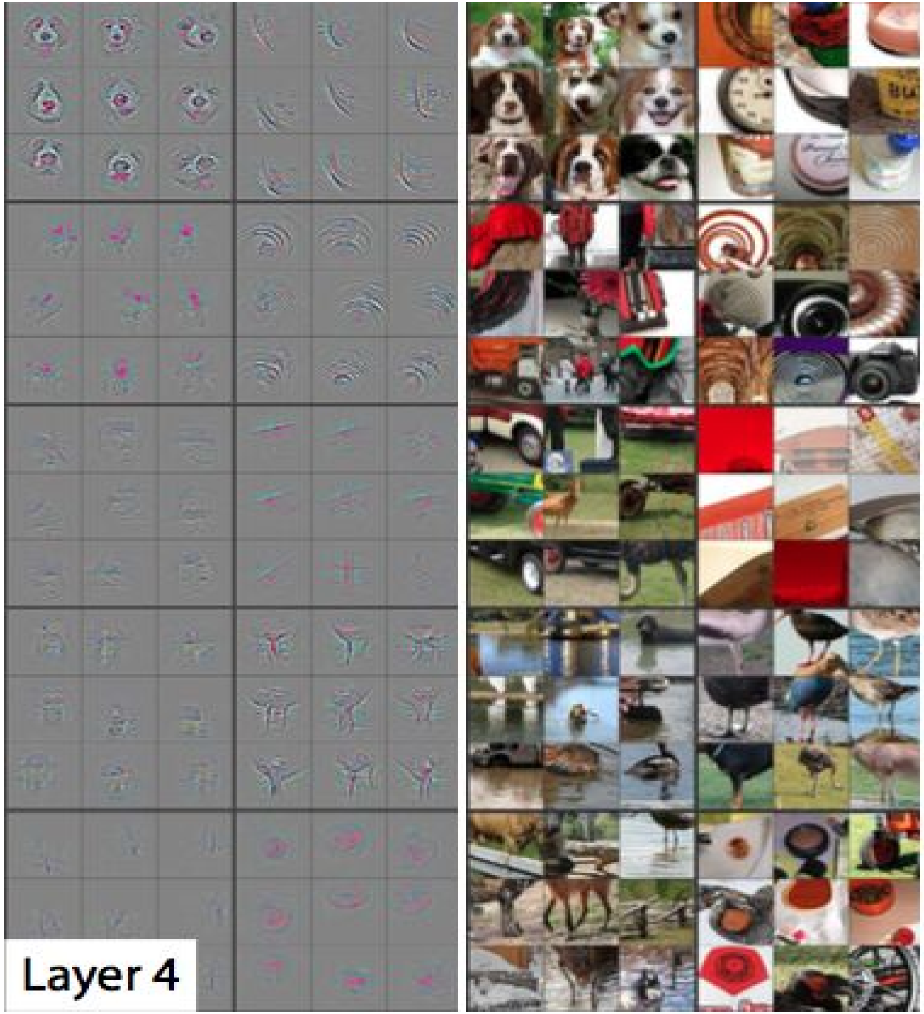


What are the models learning?



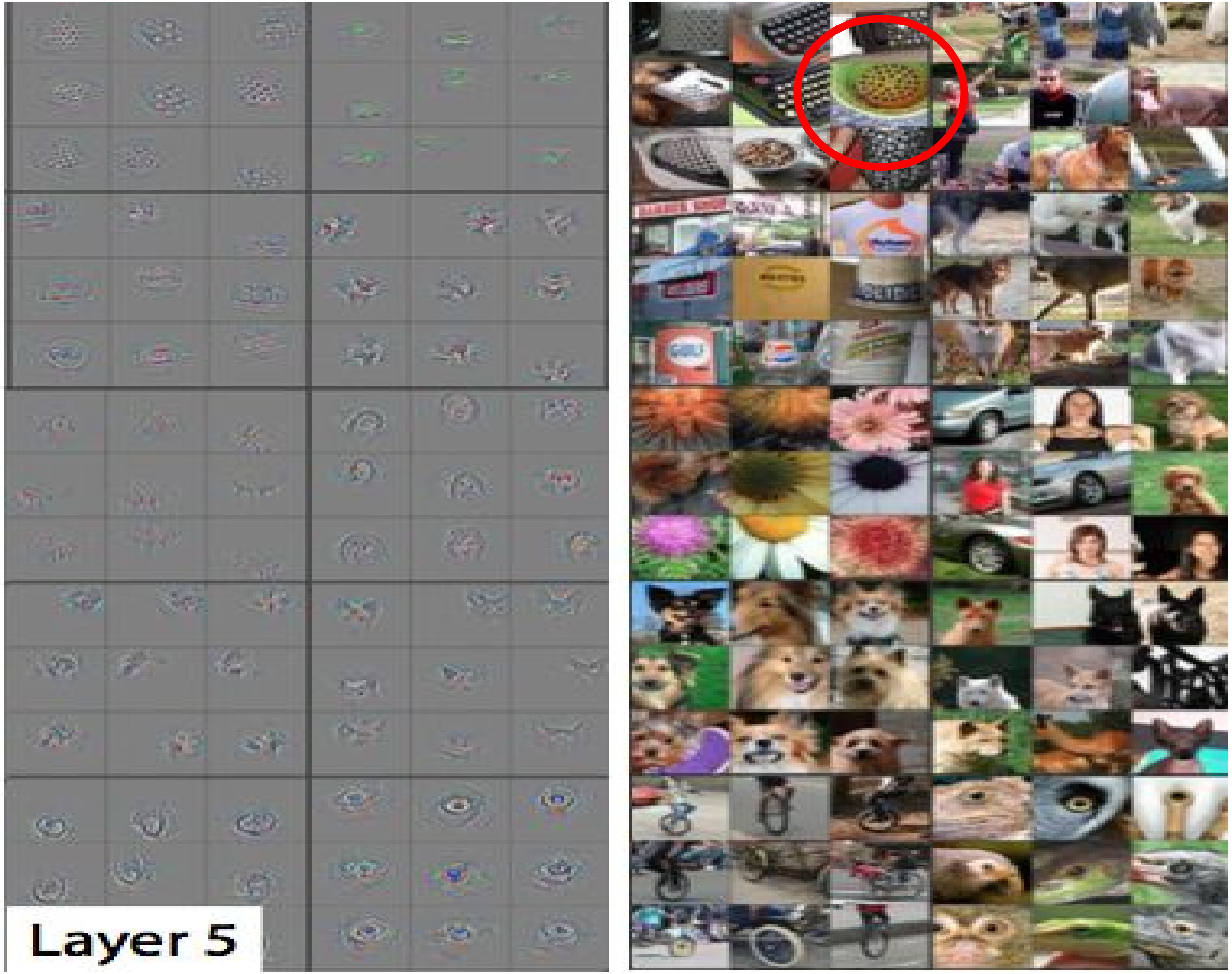
What are the models learning?

Discriminative



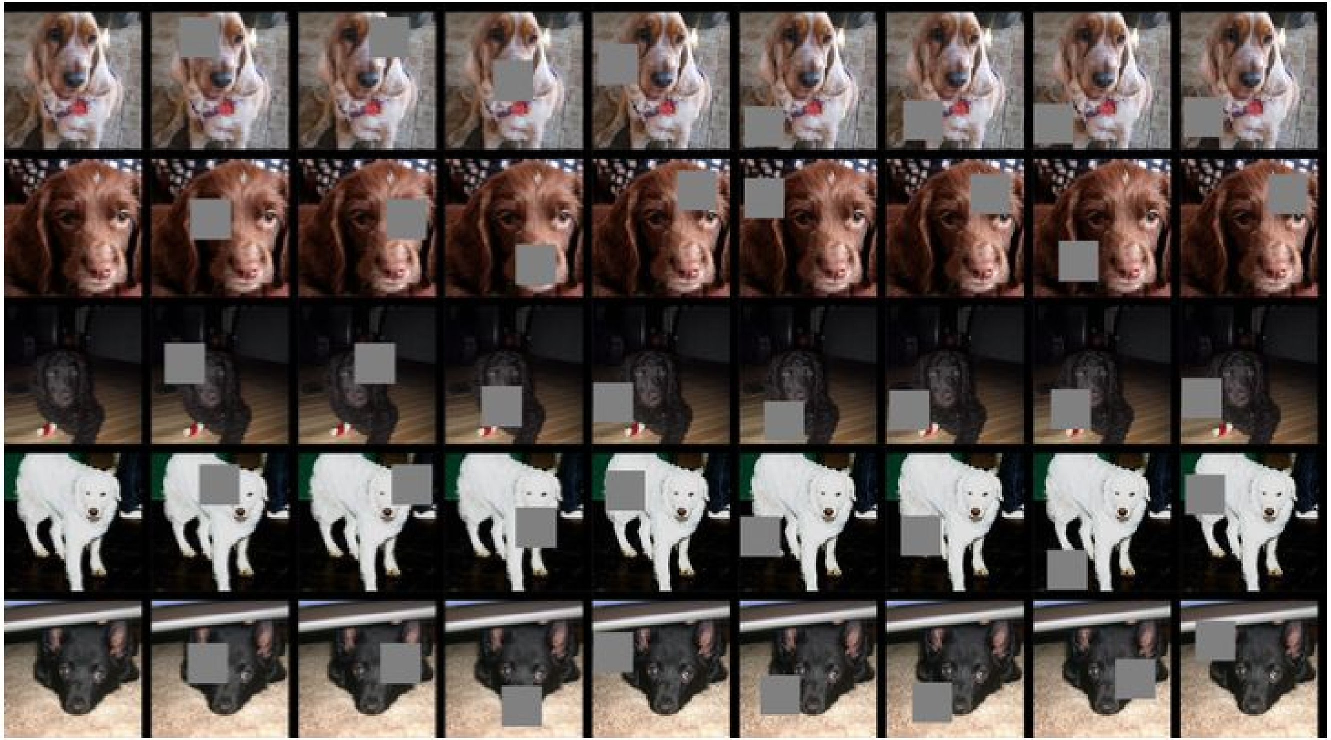
What are the models learning?

Color not washed out early on



Do objects correspond with one another?

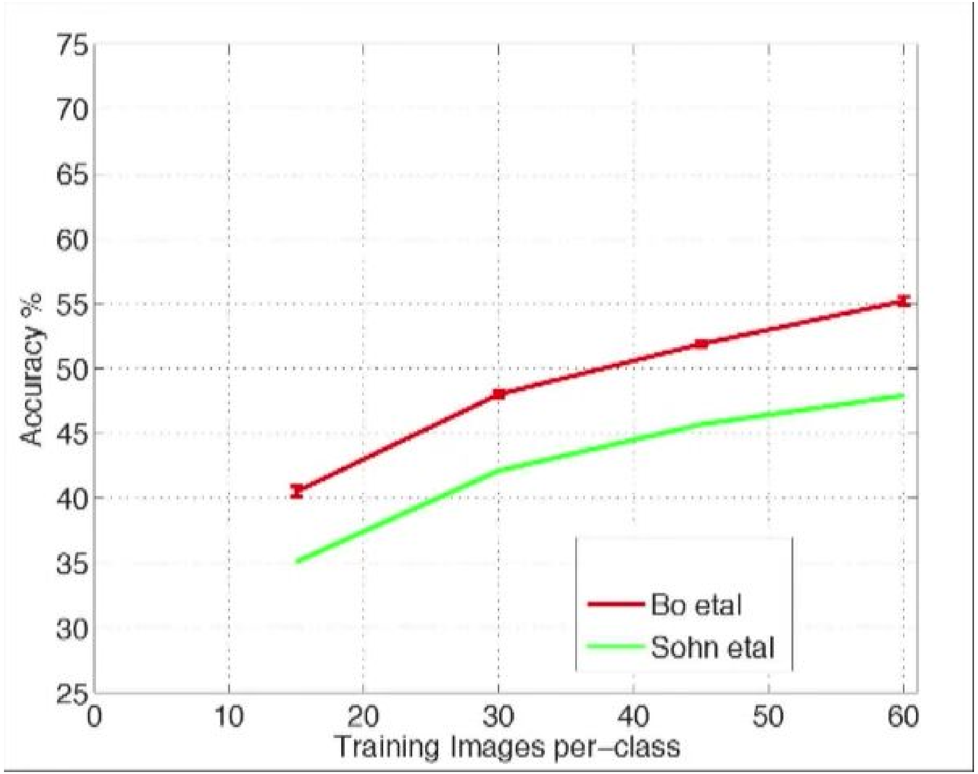
5 random dogs from ImageNet(2012)



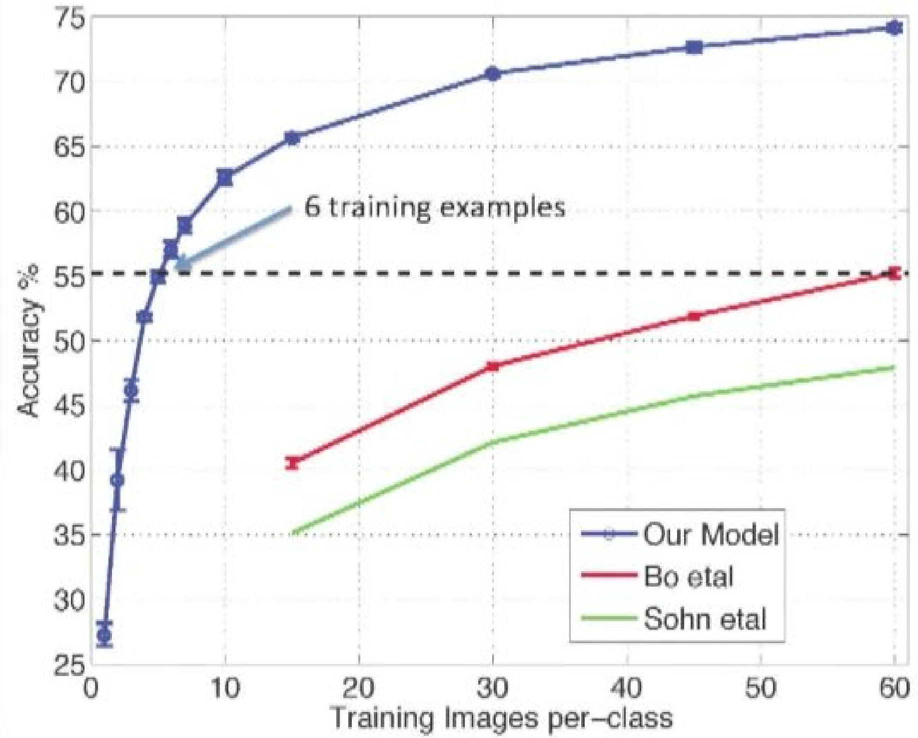
Do objects correspond with one another?

Occlusion Location	Mean Feature Sign Change Layer 5	Mean Feature Sign Change Layer 7
Right Eye	0.067 ± 0.007	0.069 ± 0.015
Left Eye	0.069 ± 0.007	0.068 ± 0.013
Nose	0.079 ± 0.017	0.069 ± 0.011
Random	0.107 ± 0.017	0.073 ± 0.014

Do features generalize well with other models?



Do features generalize well with other models?



Results: Feature Analysis

	Cal-101 (30/class)	Cal-256 (60/class)
SVM (1)	44.8 ± 0.7	24.6 ± 0.4
SVM (2)	66.2 ± 0.5	39.6 ± 0.3
SVM (3)	72.3 ± 0.4	46.0 ± 0.3
SVM (4)	76.6 ± 0.4	51.3 ± 0.1
SVM (5)	$\mathbf{86.2 \pm 0.8}$	65.6 ± 0.3
SVM (7)	$\mathbf{85.5 \pm 0.4}$	$\mathbf{71.7 \pm 0.2}$
Softmax (5)	82.9 ± 0.4	65.7 ± 0.5
Softmax (7)	$\mathbf{85.4 \pm 0.4}$	$\mathbf{72.6 \pm 0.1}$

Table 7. Analysis of the discriminative information contained in each layer of feature maps within our ImageNet-pretrained convnet. We train either a linear SVM or softmax on features from different layers (as indicated in brackets) from the convnet. Higher layers generally produce more discriminative features.

Recap

1. Not random and uninterpretable
    - Greater invariance and class discrimination at higher layers

2. Network shows a hierarchical nature of features in the network

3. Occlusion experiments are highly sensitive to local structure in image

4. ImageNet trained model can generalize well in other datasets

Application - Art and Inceptionism

Conceptual Imitation Learning: An Application to Human-Robot Interaction

Ravinder Dhesi

Motivation

- Most previous works focus on forming concepts based on similarity in perceptual characteristics
- There is not enough work to find abstract concepts, which can obtain skills that can’t be obtained from the perceptual alone.
- The goal is to teach robots abstract concepts through imitation learning

Background

- HMMs have been used for the development of most imitation models
- Most past works have been using perceptual space similarity, which can’t work for relational concepts
- closest work to this was proposed by Mobahi et al. (2007, 2005) however it only worked for single observations and not sequences

Terminology

- imitation: paradigm to teach complicated tasks to complex robots, such as humanoids
- Perceptual Concepts: based on similarity of instances in perceptual space
- Relational Concepts: uses both perceptual space and external information
- Associative Concepts: uses shared functional characteristics

Concept Illustration

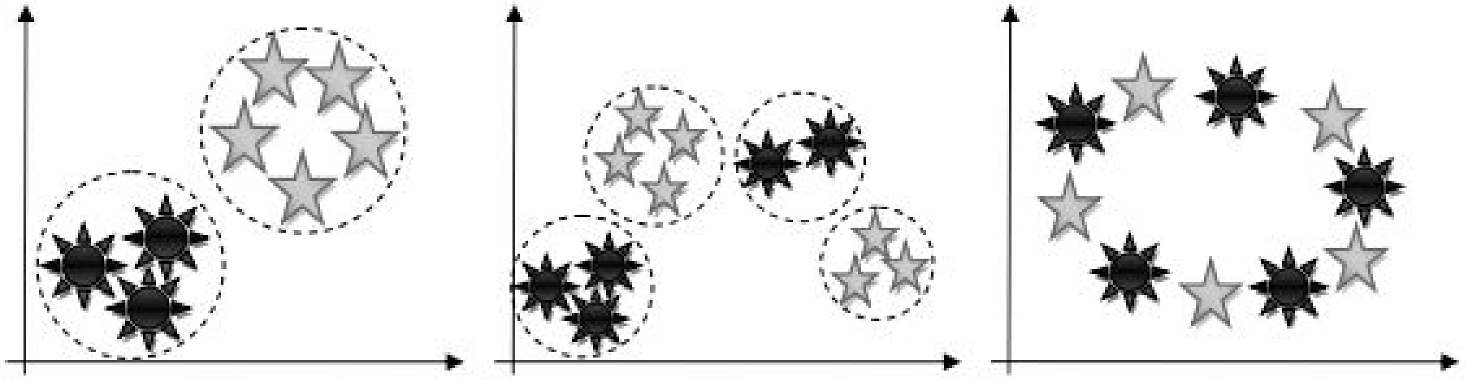


Figure 1: Three types of concepts (from left to right): Perceptual, Relational, and Associative.

Methods - Learning Phase

- The teacher will give the robot a signal and, using it’s existing information, it will try to find a prototype in the right concept
- If the reinforcement is positive, it’ll group the gesture with that concept
- If negative, the robot will keep trying
- If no concepts work a new one is formed

Methods - Learning Phase

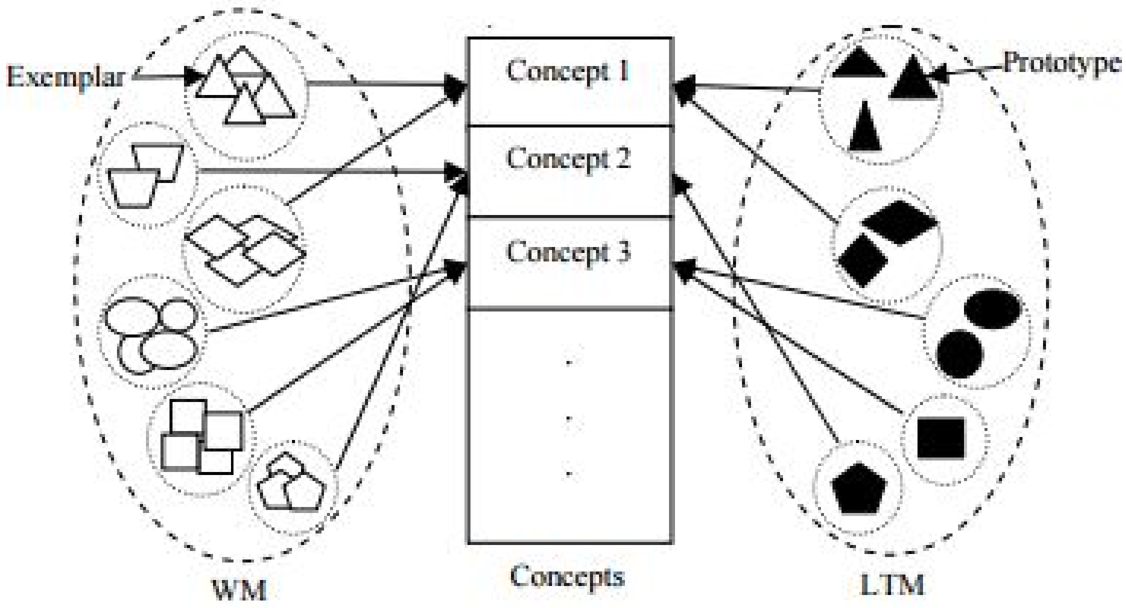


Figure 2: Associative memory of exemplars, prototypes, and concepts.

Methods - Motor Babbling

- This is how the coordination of the robot’s skills is developed
- First, temporary goals are determined
- Robot starts with an initial joint set up and makes small motions in it’s joint variables to gradually clear the temporary goals
- The information at the goals is obtained and, through it’s, feedforward neural network, it is mapped from sensory space to motor space to learn

Methods - Recall Phase

- At this point, the teacher stops giving the robot information and it has to learn on it’s own from this point
- It will try to use it’s existing knowledge to find the ght concept
- Once the right command is found it will try to use motor babbling to make the gesture

Experiment and Results

five people were asked to draw six signs (“Heart”, “Rectangle”, “Infinity”, “Tick”, “Arc”, and “Eight”) by moving their hands in the air for a robotic marionette controlled by 8 servo motors that pull the attached strings

Concept	Heart	Rectangle	Infinity	Tick	Arc	Eight
Heart	97.50	0.00	0.00	0.00	0.00	2.50
Rectangle	5.00	87.00	4.00	0.00	0.00	4.00
Infinity	15.00	0.00	80.00	0.00	0.00	5.00
Tick	2.22	0.00	0.00	97.78	0.00	0.00
Arc	6.94	6.67	11.67	0.00	72.50	2.22
Eight	5.00	0.00	0.00	0.00	0.00	95.00

able 3: Average Confusion Matrix for the experiment with 5-fold cross validation

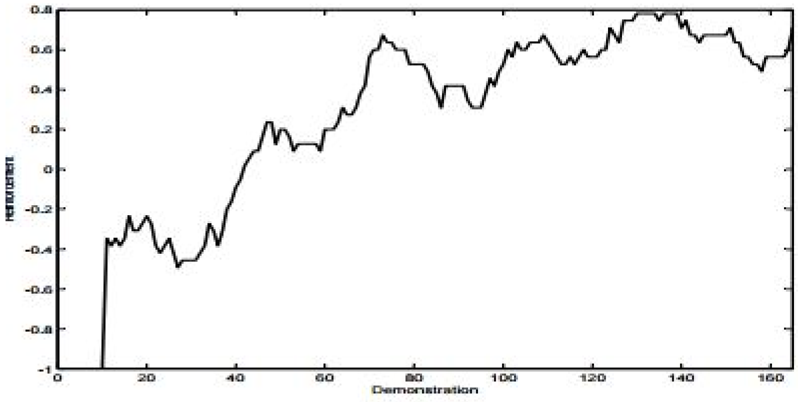


Figure 5: Reinforcement over demonstrations.

Experiment and Results

Figure 8: An example of hand-eye coordination with the robot.

Conclusion

- relational concepts form as a collection of irregularly scattered HMMs that are combined because of their functional properties
- Using this abstraction helps with efficient memory management, generalization, knowledge transfer, and flexibility of choice between different alternatives of gestures
- The experiment shows that this algorithm is good for getting concepts, creating and organizing prototypes, and recognizing and recreating abstract behaviors

Any Questions?

Credits

Hossein Hajimirsadeghi hajimirsadeghi@ut.ac.ir

Majid Nili Ahmadabadi* mnili@ut.ac.ir

Mostafa Ajallooeian ajal@ut.ac.ir

Babak Nadjar Araabi* araabi@ut.ac.ir

Hadi Moradi moradih@ut.ac.ir

School of Electrical and Computer Engineering University of Tehran Tehran, Iran

Editor: Masashi Sugiyama and Qiang Yang

JMLR: Workshop and Conference Proceedings 13: 331-346 2nd Asian Conference on Machine Learning (ACML2010), Tokyo, Japan, Nov. 8–10, 2010.