Zero-shot learning

Soichiro Tanaka, Inspection室

2019/12/26

Slide Title

- What is zero-shot learning?
- Why do we need zero-shot learning?
- How does it work?
- Experiment
- Conclusion

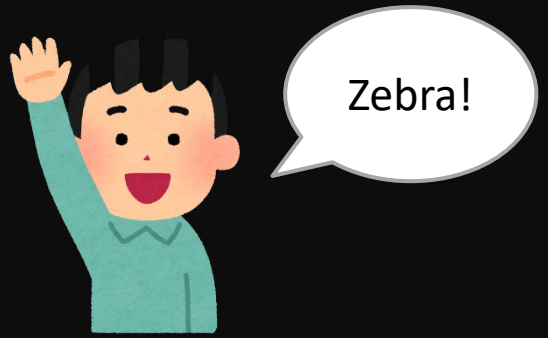
What is zero-shot learning?

He has seen a horse before.

He has never seen a zebra.

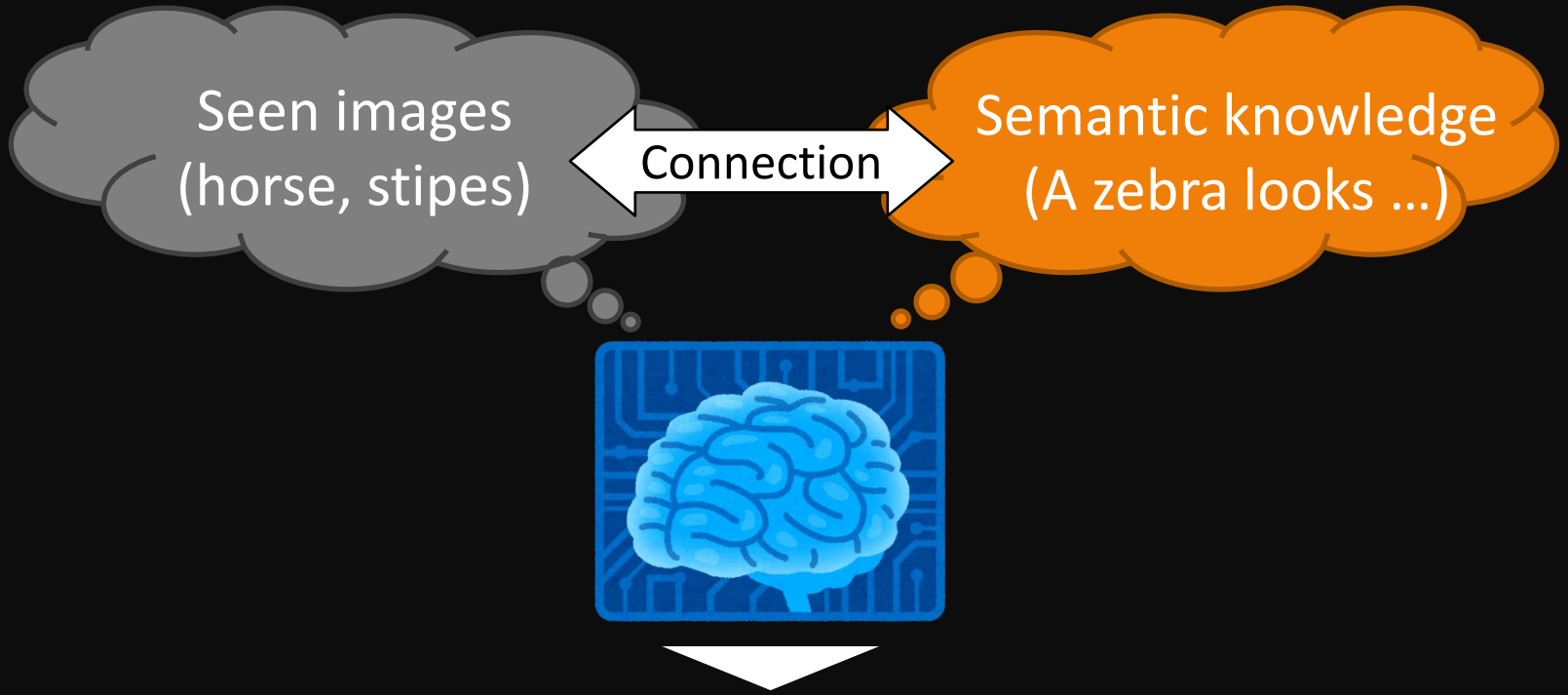
What is zero-shot learning?

But he has read somewhere
“A zebra looks similar to a horse,
but has black-and-white stripes.”



What is zero-shot learning?

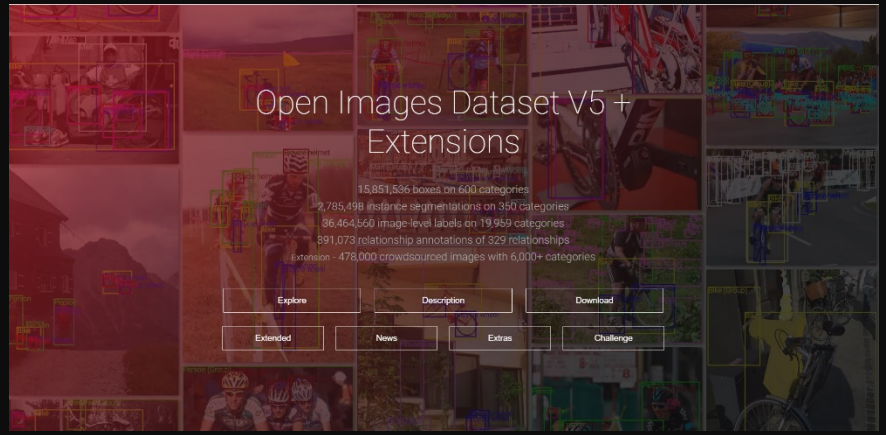
Motivation



Predict unseen classes without additional training

Why do we need zero-shot learning?

Because of vast and growing number of categories.



15,851,536 boxes on 600 categories
2,785,498 instance segmentations on 350 categories
36,464,560 image-level labels on **19,959 categories**
391,073 relationship annotations of 329 relationships

Adding new category to DL model requires a lot of labor.

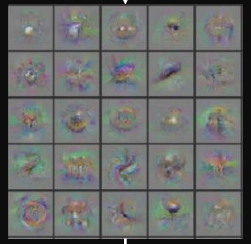
- ✓ Collecting **enough** data for each class
- ✓ **Annotation** cost is high
- ✓ Change the output layer and **train agein**
- ✓ More **time** or **computational power** with more data

Zero-shot learning can solve these problems maybe.

How does it work?

Learning phase

feature extraction



word vector

Word embedding space

Learn how to map

"horse"

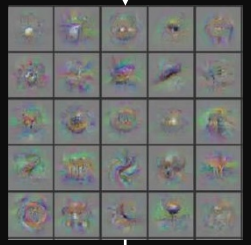
word2vec

word vector

- eat grasses
- run fast
- hunted
- brown
- …

How does it work?

Prediction phase



Word embedding space

Search nearest vector

Generate word from vector

word2vec

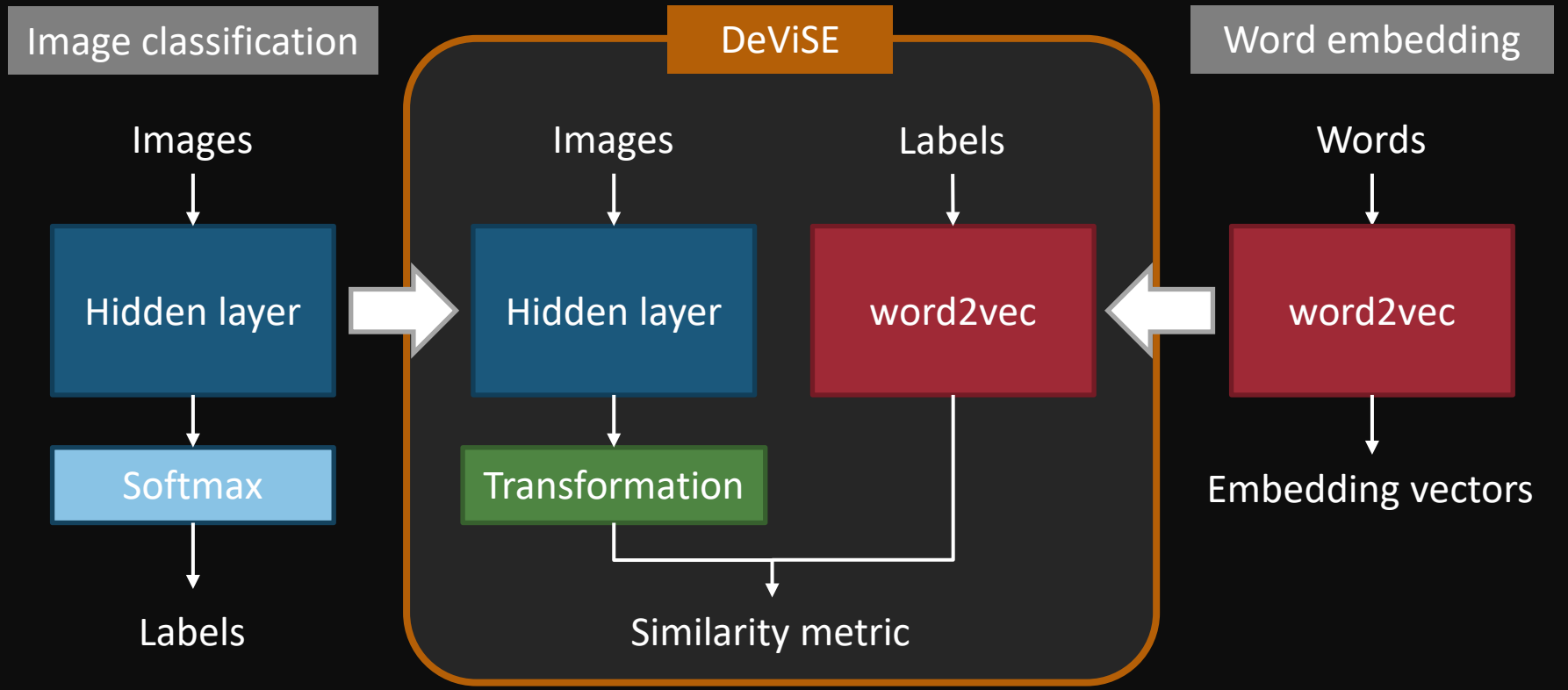
"zebra"

word vector

- horse-like
- black-and-white
- stripes
- …

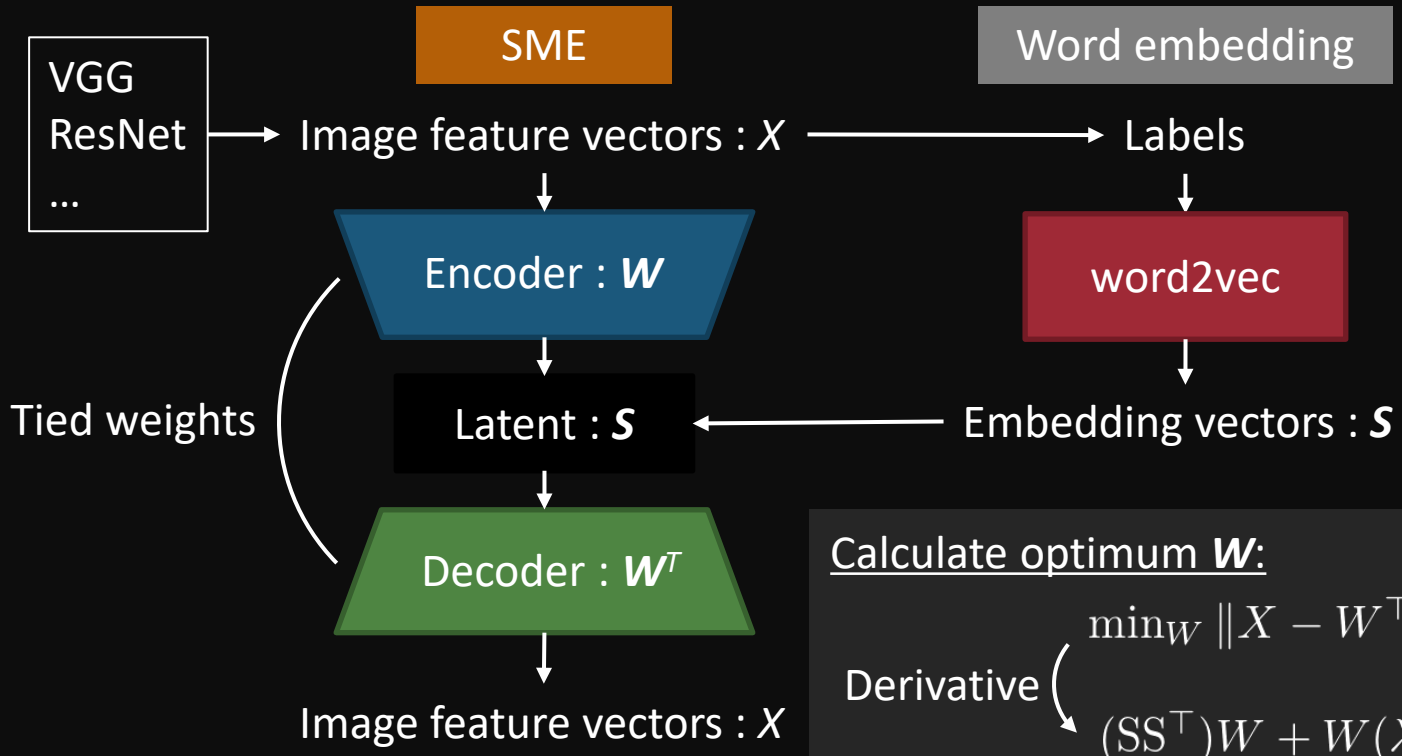
How does it work?

Deep Visual-Semantic Embedding(DeViSE) NIPS2013



How does it work?

Semantic AutoEncoder(SAE) CVPR2017



Calculate optimum ***W***:

$$\min_W \|X - W^\top S\|_F^2 + \lambda \|WX - S\|_F^2$$

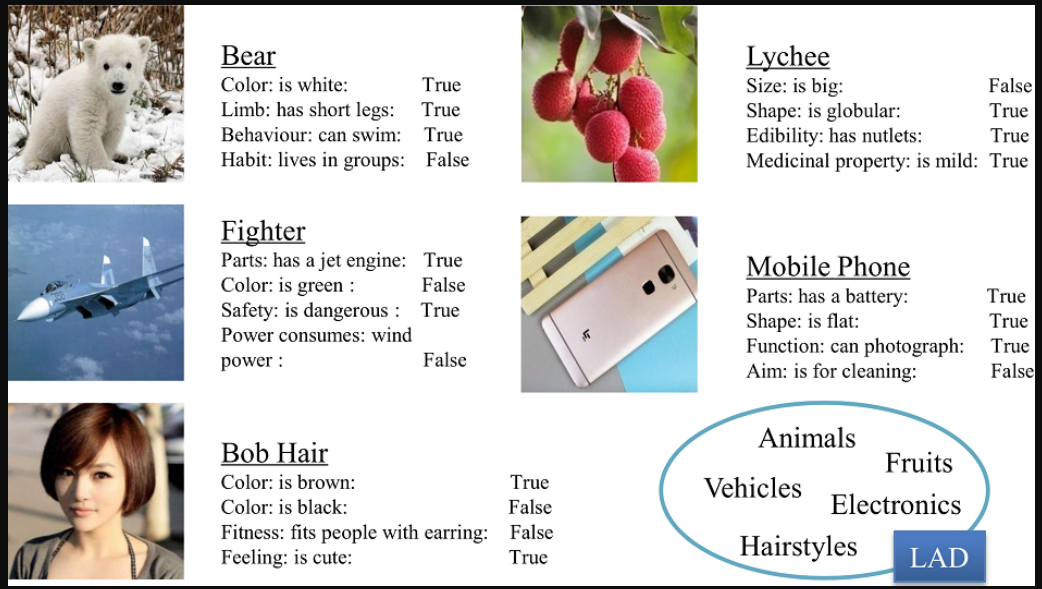
Derivative

$$\underbrace{(SS^\top)}_{const} W + W \underbrace{(\lambda X X^\top)}_{const} = \underbrace{(1+\lambda) S X^\top}_{const}$$

Experiment

Animal classification

Dataset : A Large-scale Attribute Dataset for Zero-shot Learning



Only animals

Training: 9048 images
Validation: 3878 images
Unseen classes : zebra, chimpanzee

ID	class	ID	class
0	ant	25	jellyfish
1	bear	26	koala
2	butterfly	27	leopard
3	carp	28	lion
4	cat	29	monkey
5	catfish	30	ostrich
6	cattle	31	panda
7	chicken	32	parrot
8	coral	33	platypus
9	deer	34	rabbit
10	dog	35	rat
11	dolphin	36	rhinoceros
12	donkey	37	shark
13	dragonfly	38	snail
14	duck	39	sparrow
15	eagle	40	starfish
16	elephant	41	sturgeon
17	fox	42	swallow
18	giraffe	43	swan
19	goat	44	tiger
20	goose	45	turtle
21	gorilla	46	whale
22	hedgehog	47	wolf
23	honeybee	48	woodpecker
24	horse		

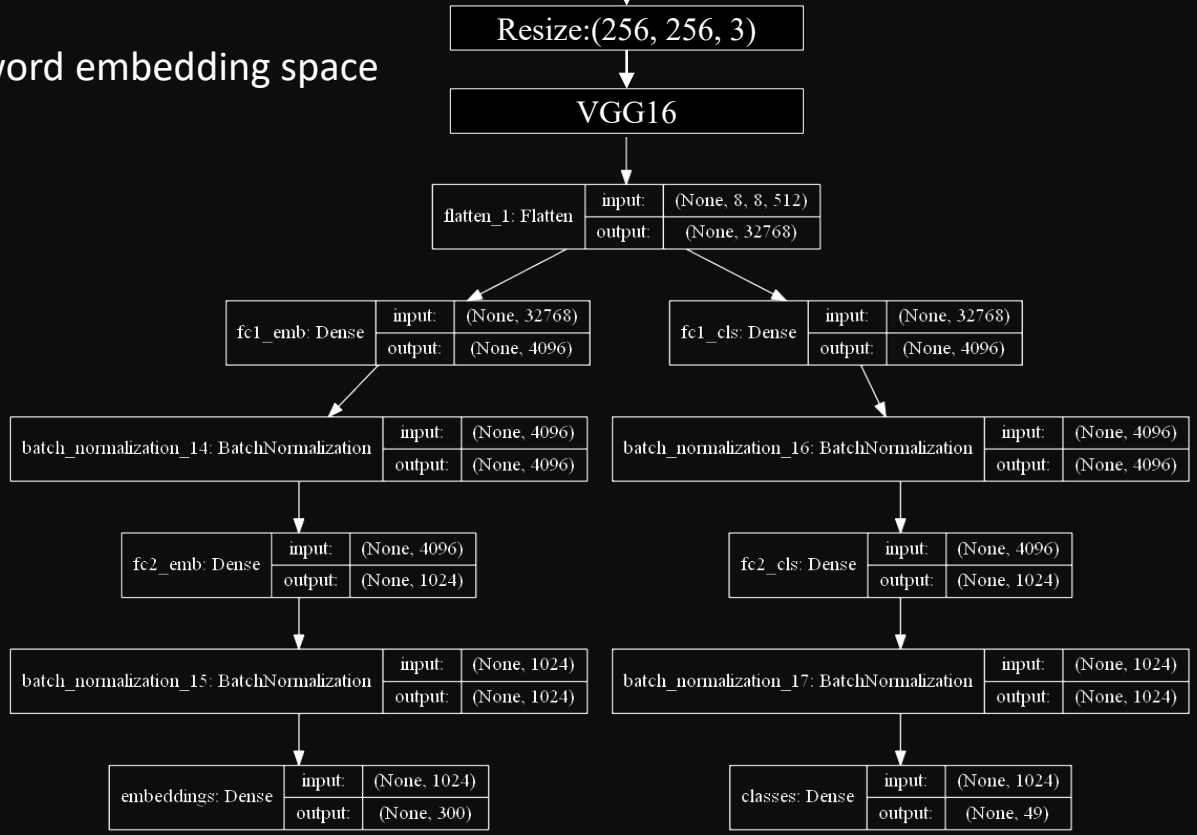
Experiment

Network architecture

VGG16 with two branches
- Mapping from image features to word embedding space (300 dim)
- Classification (49 categories)

Settings

Optimizer: adam
Learning rate: 1e-3 → 1e-6
Epochs: 1000
Batch size: 32
Input shape: (256, 256, 3)
Loss1: cosine proximity
Loss2: categorical crossentropy
Loss weights: ‘loss1’: 0.7, ‘loss2’: 0.3



Experiment

Prediction of unseen classes

Input	Output	Reliability
	woodpecker	0.857601
	squirrel	**0.707087**
	spotbill	0.704446
	liventer	0.698276
	gymnogene	0.696299
	turtle	0.886721
	alligator	**0.754621**
	tortoise	0.736731
	turtles	0.719899
	crocodile	**0.716109**
	goat	0.818994
	cow	**0.737375**
	dog	0.719159
	pig	0.716362
	rabbit	0.707818
	rhinoceros	0.985257
	rhinoceroses	0.719658
	dicerorhinus	0.695453
	hippopotamus	**0.692935**
	chousingha	0.683939
	**chicken**	**0.847994**
	goat	0.815152
	cow	0.764047
	pig	0.760932
	chickens	0.7592
	butterfly	0.778441
	parrot	0.687232
	turtle	0.647621
	frog	**0.636198**
	spotbill	0.632529
	tiger	0.896356
	leopard	0.765723
	hyena	**0.69587**
	rhinoceros	0.686366
	elephant	0.67506
	starfish	0.945515
	coral	0.776223
	urchins	**0.732707**
	urchin	**0.7239**
	corals	0.722444
	gorilla	0.993781
	ape	0.655898
	chimpanzee	**0.655245**
	monkey	0.645724
	chimp	**0.639019**

Experiment

Zebra's stripes are overwhelming!

leopard	0.954025
profelis	0.627406
panthera	0.612081
chousingha	0.597279
chowsingha	0.590462

leopard	0.902383
giraffe	0.72913
profelis	0.713392
hyena	0.705404
rhinoceros	0.696642

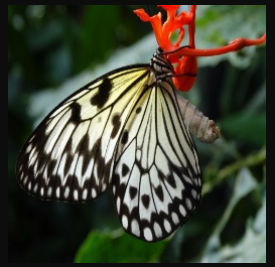
giraffe	0.794636
leopard	0.784001
animazia	0.716992
hyena	0.715736
profelis	0.706123

leopard	0.945774
tiger	0.781351
panthera	0.675595
hyena	0.647742
giraffe	0.64211

butterfly	0.773512
leopard	0.771586
tiger	0.649144
animazia	0.593183
profelis	0.576424

butterfly	0.868839
tiger	0.685825
leopard	0.618752
medley	0.594179
swallowtail	0.572934

Zebra images are classified into around leopard, giraffe, butterfly because of **weak distributed representation** power **except for stripes**.



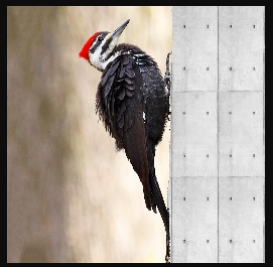
Experiment

Woodpecker vs Wood

woodpecker	**0.992235**
dendrocopos	0.71069
dryocopus	0.691981
leucotos	0.684678
campethera	0.683521

woodpecker	**0.954602**
bellied	0.693659
parrot	0.691737
dendrocopos	0.687798
pileated	0.678356

woodpecker	**0.991888**
dendrocopos	0.718132
dryocopus	0.696118
campethera	0.685811
leucotos	0.682005



squirrel	0.745027
profelis	0.722475
animazia	0.718295
parrot	0.714114
turtle	0.710388

Misrecognize tree as woodpecker because woodpecker is always with tree.
→ ZL maps predominantly the feature of tree into word embedding space.

Solutions

- ✓ Add images of other animals with tree to train data.
- ✓ Attract attention to animals（remove background, attention network, etc.）

Conclusion

Result

Succeeded in predicting unseen categories very rarely.

Problems

Feature extraction fails to cover all important features because…

- texture is overwhelming
- background attracts much attention

Feature work

- ✓ Study attention networks.
- ✓ Knowledge Graph greatly enhances the performance.
  - Zero-shot Recognition via Semantic Embeddings and Knowledge Graphs (CVPR 2018)
- ✓ Try object detection and semantic segmentation with ZL.
- ✓ Apply to actual projects in the distant future.

Unsuitable problem

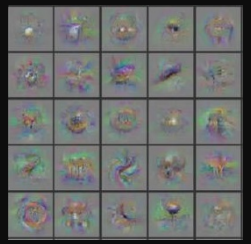


Image feature
- eight limbs
- suction
- ocean
- red
- …

word vector
- soft-bodied
- eight limbs
- suction
- highly intelligent
- …

word2vec ← “octopus”

mapping

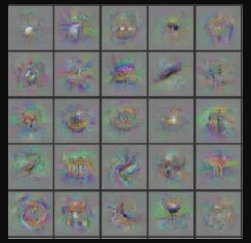


Image feature
- red circle
- white

word vector
- island
- east
- 47 prefecture
- Aging population
- …

word2vec ← “Japan”

mapping