

# Taking a closer look at the I in AI

Erik Barzagar-Nazari  
Kassel Data Science Meetup  
18 February 2020

# Meet Clever Hans



Das Talent des klugen Hans blieb so lange verborgen, weil man beim Pferd suchte, was man beim Menschen hätte finden können.

The talent of Clever Hans remained hidden for so long, because one looked for in the horse what one could have found in man.

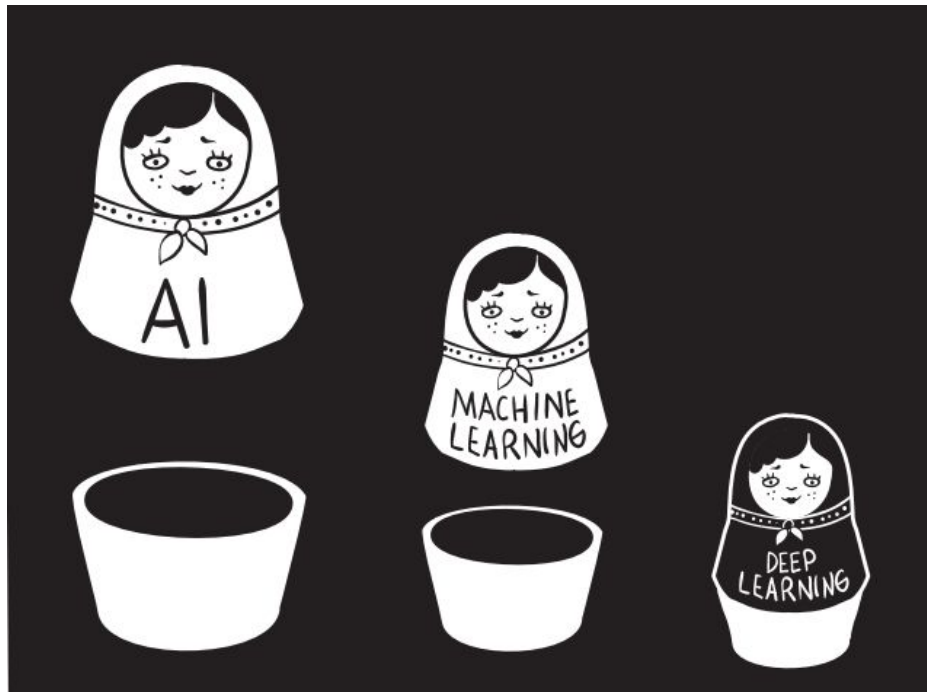
Oskar Pfungst, 1907

Das Talent der **Künstlichen Intelligenz** blieb  
so lange verborgen, weil man beim  
**Neuronalen Netz** suchte, was man beim  
Menschen hätte finden können.

The talent of **Artificial Intelligence**  
remained hidden for so long, because  
one looked for in the **Neural Network**  
what one could have found in man.

Some Data Scientist, 2020

# Artificial Intelligence



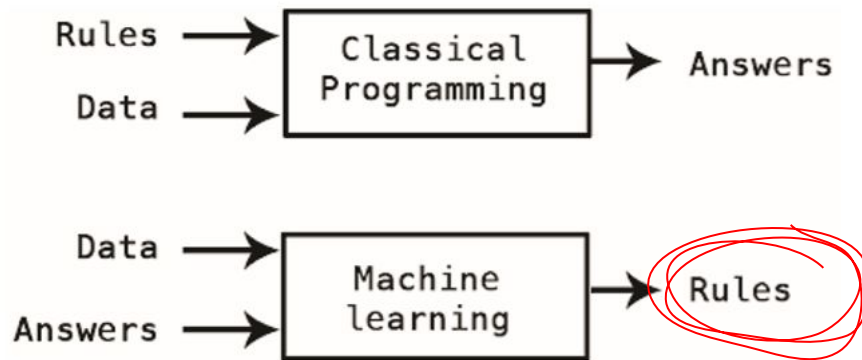
Screenshot from <https://weneedtotalk.ai/>

- Dominance of Deep Learning
- Overhyped? Probably not.
- Biological terminology and analogies...
- ...lead to some misconceptions

# Self-Learning Machines

Learn to map **data** to **answers** by looking  
for statistical patterns

Machine Learning outputs static **rules**,  
not intelligent agents



# Catastrophic Forgetting

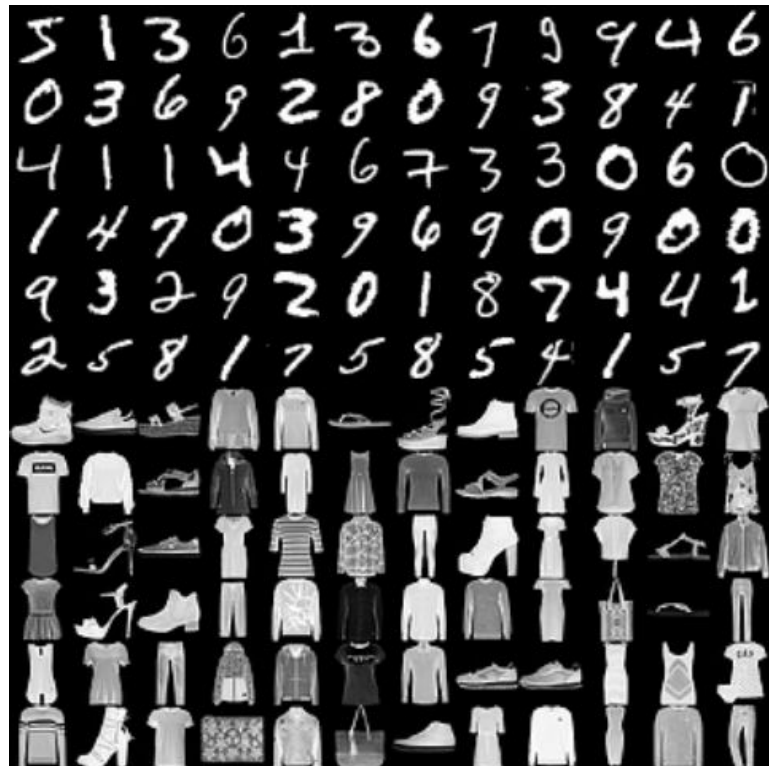


Catastrophic interference, also known as catastrophic forgetting, is the tendency of an artificial neural network to **completely and abruptly forget** previously learned information upon learning new information.

Wikipedia, 15 February 2020

# Catastrophic Forgetting Experiment

1. Define a Convolutional Neural Network
2. Train the network to classify digits using MNIST
3. Train the network to classify fashion categories using Fashion-MNIST



# Catastrophic Forgetting: Example

```
# define convnet
t800 <- keras_model_sequential(name = 'T-800') %>%
  layer_conv_2d(filters = 32, kernel_size = c(3,3), activation = 'relu', input_shape = input_shape) %>%
  layer_conv_2d(filters = 64, kernel_size = c(3,3), activation = 'relu') %>%
  layer_max_pooling_2d(pool_size = c(2, 2)) %>%
  layer_dropout(rate = 0.25) %>%
  layer_flatten() %>%
  layer_dense(units = 128, activation = 'relu') %>%
  layer_dropout(rate = 0.5) %>%
  layer_dense(units = num_classes, activation = 'softmax')

# compile model
t800 %>% compile(
  loss = loss_categorical_crossentropy,
  optimizer = optimizer_adadelata(),
  metrics = 'accuracy'
)
```

# Catastrophic Forgetting: Example

Model  
Model: "T-800"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 26, 26, 32)	320
conv2d_1 (Conv2D)	(None, 24, 24, 64)	18496
max_pooling2d (MaxPooling2D)	(None, 12, 12, 64)	0
dropout (Dropout)	(None, 12, 12, 64)	0
flatten (Flatten)	(None, 9216)	0
dense (Dense)	(None, 128)	1179776
dropout_1 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 10)	1290

Total params: 1,199,882  
Trainable params: 1,199,882  
Non-trainable params: 0

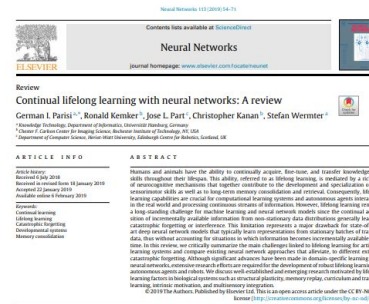
## Catastrophic Forgetting Experiment

Trained for 12 epochs on each dataset



# Catastrophic Forgetting

- Stability-Plasticity-Dilemma
- Active research area
- Current remedy: retrain and redeploy



Contents	
1. Introduction	55
2. Measuring Catastrophic Forgetting in Neural Networks	55
2.1. The Stability-Plasticity-Dilemma	55
2.2. The Catastrophic Forgetting Problem	55
2.3. The Catastrophic Forgetting Problem	55
2.4. Learning without Forgetting	55
3. Learning without Forgetting	55
3.1. Learning without Forgetting	55
3.2. Learning without Forgetting	55
3.3. Learning without Forgetting	55
3.4. Learning without Forgetting	55
3.5. Learning without Forgetting	55
3.6. Learning without Forgetting	55
3.7. Learning without Forgetting	55
3.8. Learning without Forgetting	55
3.9. Learning without Forgetting	55
3.10. Learning without Forgetting	55
3.11. Learning without Forgetting	55
3.12. Learning without Forgetting	55
3.13. Learning without Forgetting	55
3.14. Learning without Forgetting	55
3.15. Learning without Forgetting	55
3.16. Learning without Forgetting	55
3.17. Learning without Forgetting	55
3.18. Learning without Forgetting	55
3.19. Learning without Forgetting	55
3.20. Learning without Forgetting	55
3.21. Learning without Forgetting	55
3.22. Learning without Forgetting	55
3.23. Learning without Forgetting	55
3.24. Learning without Forgetting	55
3.25. Learning without Forgetting	55
3.26. Learning without Forgetting	55
3.27. Learning without Forgetting	55
3.28. Learning without Forgetting	55
3.29. Learning without Forgetting	55
3.30. Learning without Forgetting	55
3.31. Learning without Forgetting	55
3.32. Learning without Forgetting	55
3.33. Learning without Forgetting	55
3.34. Learning without Forgetting	55
3.35. Learning without Forgetting	55
3.36. Learning without Forgetting	55
3.37. Learning without Forgetting	55
3.38. Learning without Forgetting	55
3.39. Learning without Forgetting	55
3.40. Learning without Forgetting	55
3.41. Learning without Forgetting	55
3.42. Learning without Forgetting	55
3.43. Learning without Forgetting	55
3.44. Learning without Forgetting	55
3.45. Learning without Forgetting	55
3.46. Learning without Forgetting	55
3.47. Learning without Forgetting	55
3.48. Learning without Forgetting	55
3.49. Learning without Forgetting	55
3.50. Learning without Forgetting	55
3.51. Learning without Forgetting	55
3.52. Learning without Forgetting	55
3.53. Learning without Forgetting	55
3.54. Learning without Forgetting	55
3.55. Learning without Forgetting	55
3.56. Learning without Forgetting	55
3.57. Learning without Forgetting	55
3.58. Learning without Forgetting	55
3.59. Learning without Forgetting	55
3.60. Learning without Forgetting	55
3.61. Learning without Forgetting	55
3.62. Learning without Forgetting	55
3.63. Learning without Forgetting	55
3.64. Learning without Forgetting	55
3.65. Learning without Forgetting	55
3.66. Learning without Forgetting	55
3.67. Learning without Forgetting	55
3.68. Learning without Forgetting	55
3.69. Learning without Forgetting	55
3.70. Learning without Forgetting	55
3.71. Learning without Forgetting	55
3.72. Learning without Forgetting	55
3.73. Learning without Forgetting	55
3.74. Learning without Forgetting	55
3.75. Learning without Forgetting	55
3.76. Learning without Forgetting	55
3.77. Learning without Forgetting	55
3.78. Learning without Forgetting	55
3.79. Learning without Forgetting	55
3.80. Learning without Forgetting	55
3.81. Learning without Forgetting	55
3.82. Learning without Forgetting	55
3.83. Learning without Forgetting	55
3.84. Learning without Forgetting	55
3.85. Learning without Forgetting	55
3.86. Learning without Forgetting	55
3.87. Learning without Forgetting	55
3.88. Learning without Forgetting	55
3.89. Learning without Forgetting	55
3.90. Learning without Forgetting	55
3.91. Learning without Forgetting	55
3.92. Learning without Forgetting	55
3.93. Learning without Forgetting	55
3.94. Learning without Forgetting	55
3.95. Learning without Forgetting	55
3.96. Learning without Forgetting	55
3.97. Learning without Forgetting	55
3.98. Learning without Forgetting	55
3.99. Learning without Forgetting	55
3.100. Learning without Forgetting	55

Correspondence to: Knowledge Technology, Department of Informatics, University of Warwick, Coventry CV4 7AL, UK. Email: k.t.k@warwick.ac.uk  
© 2019 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).  
DOI: <https://doi.org/10.1016/j.neunet.2019.03.001>

## Measuring Catastrophic Forgetting in Neural Networks

Ronald Kemker, Marc McCleure, Angelina Ahlbin, Tyler Hayes, Christopher Kannan  
School of Computer Science  
Department of Informatics  
University of Warwick  
Coventry CV4 7AL, UK  
Email: r.kemker@warwick.ac.uk

Abstract  
Deep neural networks are used in many forms of the real world. As such, they are often required to learn new tasks while maintaining performance on previous tasks. This is a challenging problem because of the stability-plasticity dilemma. In this paper, we propose a new conceptualization of the catastrophic forgetting problem in terms of a symmetric trade-off between transfer and maintenance that can be captured by reference gradient alignment across examples. We then propose a new algorithm, Meta-Experience Replay (MER), that directly exploits this view by combining experience replay with optimization based meta-learning. This method learns parameters that make interference based on future gradients less likely and transfer based on future gradients more likely. We conduct experiments across continuous lifelong learning benchmarks and non-continuous reinforcement learning environments demonstrating that our approach consistently improves recently proposed baselines for continual learning. Our experiments show that the gap between the performance of MER and baseline algorithms grows both as the number of tasks increases and more non-stationary and as the fraction of the total experiences stored per example.

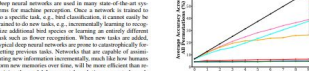


Figure 1: Catastrophic forgetting impairs incremental learning in neural networks. As a network is incrementally trained (solid line), ideally its performance would match that of a model trained offline with all of the data (dashed line). In this paper, we develop methods and benchmarks for measuring catastrophic forgetting. Our experiments show that even methods designed to prevent catastrophic forgetting perform significantly worse than an offline model. Incremental learning is key in many real-world applications because it allows the model to adapt after being deployed.

Incremental learning previously learned training data. From the problem of incremental learning, we can generalize to the problem of incremental learning. Incremental learning is a problem where a model is trained on a sequence of tasks, and the model is required to learn new tasks while maintaining performance on previous tasks. This is a challenging problem because of the stability-plasticity dilemma. In this paper, we propose a new conceptualization of the catastrophic forgetting problem in terms of a symmetric trade-off between transfer and maintenance that can be captured by reference gradient alignment across examples. We then propose a new algorithm, Meta-Experience Replay (MER), that directly exploits this view by combining experience replay with optimization based meta-learning. This method learns parameters that make interference based on future gradients less likely and transfer based on future gradients more likely. We conduct experiments across continuous lifelong learning benchmarks and non-continuous reinforcement learning environments demonstrating that our approach consistently improves recently proposed baselines for continual learning. Our experiments show that the gap between the performance of MER and baseline algorithms grows both as the number of tasks increases and more non-stationary and as the fraction of the total experiences stored per example.

Our experimental results show that the gap between the performance of MER and baseline algorithms grows both as the number of tasks increases and more non-stationary and as the fraction of the total experiences stored per example.

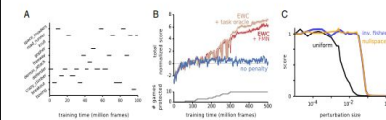


Fig. 4. Results of the incremental learning task. The x-axis shows the number of previous tasks. The y-axis shows the performance on a new task (red line) and on previous tasks (blue line). The results show that the performance of a model trained on a sequence of tasks is significantly better than a model trained on a single task. The results also show that the performance of a model trained on a sequence of tasks with a different learning rate is significantly better than a model trained on a sequence of tasks with a different learning rate. The results also show that the performance of a model trained on a sequence of tasks with a different learning rate and a different number of tasks is significantly better than a model trained on a sequence of tasks with a different learning rate and a different number of tasks.

previous's variance (as in a Laplace approximation) does contain a significant weight (Fig. 4C). Our initial experiments suggest that this small increase in the model's variance is due to the fact that the model is not able to learn the new task without forgetting the old task.

Published as a conference paper at ICLR 2019

## LEARNING TO LEARN WITHOUT FORGETTING BY MAXIMIZING TRANSFER AND MINIMIZING INTERFERENCE

Mathieu Bressan<sup>1</sup>, Agnès Cisse<sup>2</sup>, Robert Agazzi<sup>1</sup>, Xiao Luo<sup>1</sup>, Jina Roh<sup>1</sup>, Yuhui Tang<sup>1</sup>, and Gaurav Tripathi<sup>1</sup>

<sup>1</sup>IBM Research, Yorktown Heights, NY  
<sup>2</sup>Statistics and Computer Science Departments, Stanford NLP Group, Stanford University  
<sup>3</sup>MIT IBM Watson AI Lab  
<sup>4</sup>Department of Brain and Cognitive Sciences, MIT

Abstract  
Lack of performance when it comes to continual learning over non-stationary distributions of data remains a major challenge in scaling neural network learning to more human realistic settings. In this work we propose a new conceptualization of the catastrophic forgetting problem in terms of a symmetric trade-off between transfer and maintenance that can be captured by reference gradient alignment across examples. We then propose a new algorithm, Meta-Experience Replay (MER), that directly exploits this view by combining experience replay with optimization based meta-learning. This method learns parameters that make interference based on future gradients less likely and transfer based on future gradients more likely. We conduct experiments across continuous lifelong learning benchmarks and non-continuous reinforcement learning environments demonstrating that our approach consistently improves recently proposed baselines for continual learning. Our experiments show that the gap between the performance of MER and baseline algorithms grows both as the number of tasks increases and more non-stationary and as the fraction of the total experiences stored per example.

## 1 SOLVING THE CONTINUAL LEARNING PROBLEM

A long held goal of AI is to build agents capable of operating autonomously for long periods. Such agents must incrementally learn and adapt to a changing environment while maintaining performance of what they have learned before, a setting known as lifelong learning (Thrun, 1994; 1996). In this paper we explore a natural extension of this problem to the setting of continual learning (Colt, 1994). In continual learning, we assume that the data is drawn from a sequence of tasks, each with a different distribution. The goal is to learn a model that can perform well on all tasks, even those that have not been seen before. This is a challenging problem because of the stability-plasticity dilemma. In this paper, we propose a new conceptualization of the catastrophic forgetting problem in terms of a symmetric trade-off between transfer and maintenance that can be captured by reference gradient alignment across examples. We then propose a new algorithm, Meta-Experience Replay (MER), that directly exploits this view by combining experience replay with optimization based meta-learning. This method learns parameters that make interference based on future gradients less likely and transfer based on future gradients more likely. We conduct experiments across continuous lifelong learning benchmarks and non-continuous reinforcement learning environments demonstrating that our approach consistently improves recently proposed baselines for continual learning. Our experiments show that the gap between the performance of MER and baseline algorithms grows both as the number of tasks increases and more non-stationary and as the fraction of the total experiences stored per example.

Our experimental results show that the gap between the performance of MER and baseline algorithms grows both as the number of tasks increases and more non-stationary and as the fraction of the total experiences stored per example.

Our experimental results show that the gap between the performance of MER and baseline algorithms grows both as the number of tasks increases and more non-stationary and as the fraction of the total experiences stored per example.

Published as a conference paper at ICLR 2019



Figure 1: The stability-plasticity dilemma (catastrophic forgetting) with respect to the forward and backward passes. The forward pass is shown as a solid line, and the backward pass is shown as a dashed line. The diagram illustrates how the forward pass is affected by the backward pass, and vice versa, leading to catastrophic forgetting. The diagram also shows that the forward pass is affected by the backward pass, and vice versa, leading to catastrophic forgetting. The diagram also shows that the forward pass is affected by the backward pass, and vice versa, leading to catastrophic forgetting.

# Taking Shortcuts

# Remember Clever Hans?

- AI systems do not truly understand what they are doing
- Deep Neural Networks look for a function, that maps Inputs to Outputs
- Sometimes, they take shortcuts
- “Clever Hans Features”

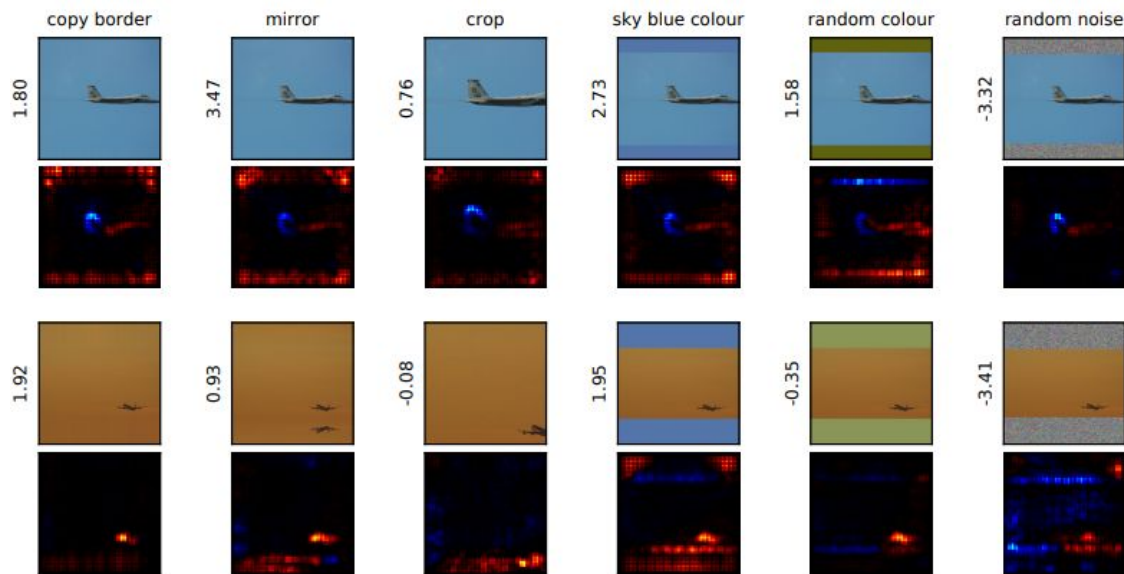


# Unusual Suspects

Lapuschkin et al. (2019) discover  
that even typical image  
preprocessing steps might introduce  
Clever Hans Features

Clever Hans Feature: image padding

<https://doi.org/10.1038/s41467-019-08987-4>



**Supplementary Figure 27:** Samples from class “aeroplane” and predicted scores for class “aeroplane”, with corresponding relevance maps, as affected by different preprocessing strategies to obtain square images. Padding with (high frequency) random noise effectively decreases the predictor output and removes the “border artifact”. Using low frequency areas (of the right color) for padding increases the predictor output for class “aeroplane” and may even introduce the “border artifact” in the first place.

# Criminals don't smile

Two researchers claim to be able to predict a person's "criminality" from its portrait using convnets.

Fortunately, other researchers quickly pointed out this study's flaws.

Clever Hans Feature: most likely the network learned to distinguish relaxed from tense facial expressions.

<https://arxiv.org/abs/1611.04135>



(a) Three samples in criminal ID photo set  $S_c$ .



(b) Three samples in non-criminal ID photo set  $S_n$

# Adversarial Attacks

# Attacking AI Systems



$x$

“panda”

57.7% confidence

$+ .007 \times$



$\text{sign}(\nabla_x J(\theta, x, y))$

“nematode”

8.2% confidence

$=$



$x +$

$\epsilon \text{sign}(\nabla_x J(\theta, x, y))$

“gibbon”

99.3 % confidence

Goodfellow, Shlens & Szegedy (2015)  
<https://arxiv.org/pdf/1412.6572.pdf>

# Targeted Attack Example

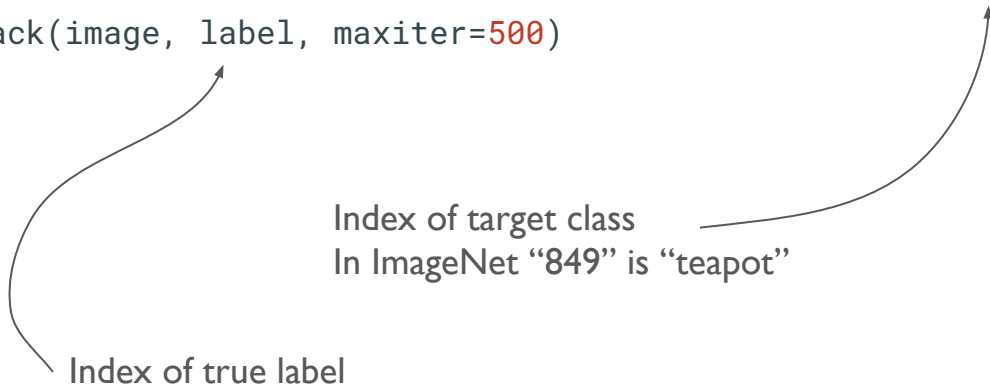
Class	Label	Score
n02123045	tabby	0.400
n02124075	Egyptian_cat	0.400
n02123159	tiger_cat	0.175
n02127052	lynx	0.007
n04553703	washbasin	0.002

Classification based on ResNet50 with  
ImageNet weights; Original resized to  
224px \* 224 px



# Targeted Attack Example

```
import foolbox
[...]  
attack = LBFGSAttack(model=fmodel, criterion=TargetClassProbability(849, p=.50))  
adversarial = attack(image, label, maxiter=500)
```



# Targeted Attack Example

Original (resized)  
Tabby (40%)



Adversarial Image  
Teapot (97,4%)






























# Attacking from the Physical World

- Printed fake signs
- Adversarial Patches
- Evaluated on different Architectures
- Evaluated in different scenarios (e.g. drive-by)

Table 1: Sample of physical adversarial examples against LISA-CNN and GTSRB-CNN.

Distance/Angle	Subtle Poster	Subtle Poster Right Turn	Camouflage Graffiti	Camouflage Art (LISA-CNN)	Camouflage Art (GTSRB-CNN)
5' 0°					
5' 15°					
10' 0°					
10' 30°					
40' 0°					
Targeted-Attack Success	100%	73.33%	66.67%	100%	80%



# Adversarial Attacks

- White Box Attacks
- Black Box Attacks
- Training with Adversarial Examples improves robustness
- Implications for cybersecurity

# Intelligence

# Let's ask Open AI's GPT-2

## **What is intelligent about Artificial Intelligence?**

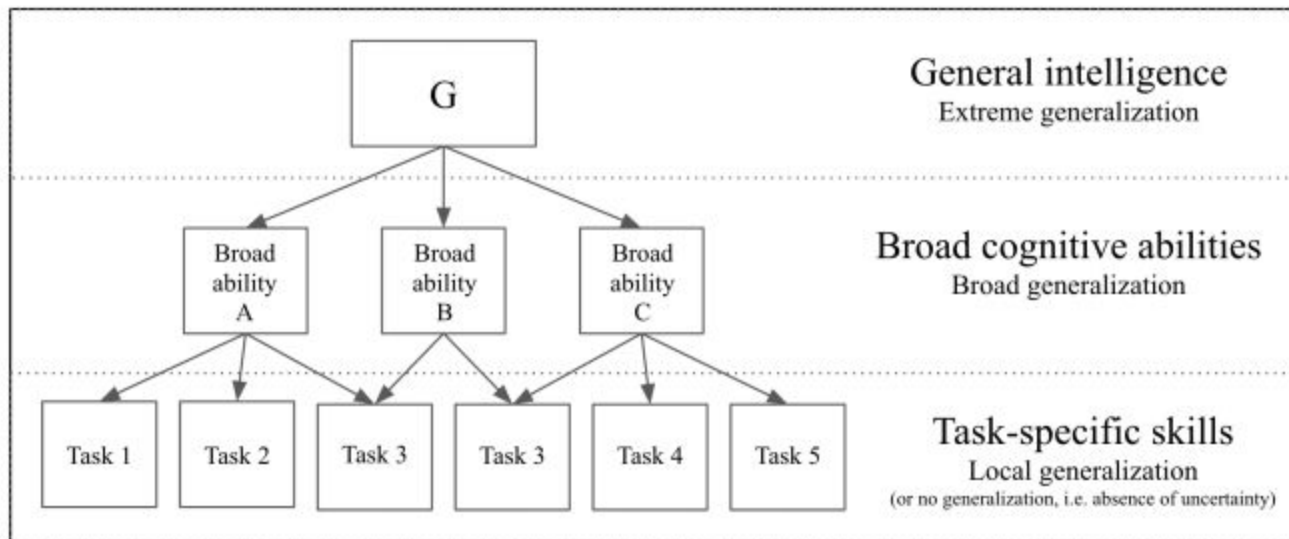
Why can't I describe it without crying?

Well you really can't and I think that's the key. There is a lot of unintelligent and odd behavior that I feel it's very easy to generalise about AI but is really specific to a group of personal behavior.

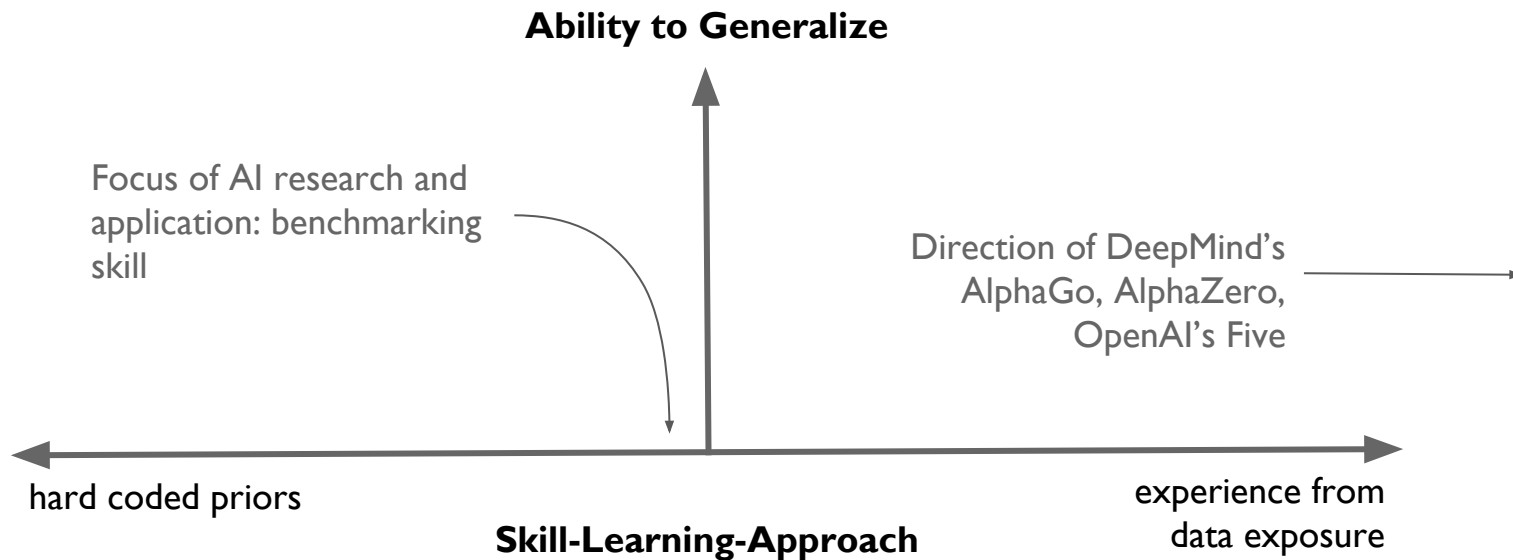
I think the key to this problem is behavioural universality. Inherently different people react to the same situations in different ways and this leads to confusion. I think people cannot help but general...

<http://talktotransformer.com>,  
accessed on December 2nd, 2019

# Hierarchical Model of Intelligence



# Skills and Abilities



“The intelligence of a system is a measure of its skill-acquisition efficiency over a scope of tasks, with respect to priors, experience, and generalization difficulty.”

Chollet (2019)

# Abstraction and Reasoning Corpus (ARC)

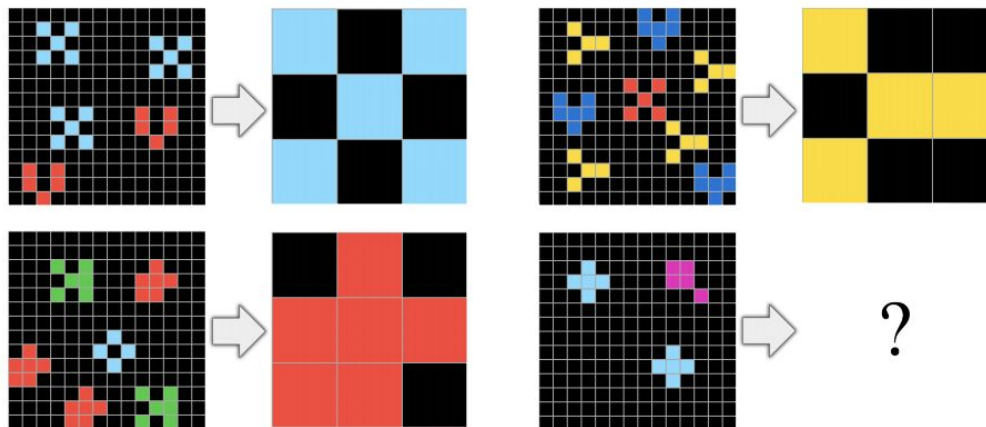


Figure 10: A task where the implicit goal is to count unique objects and select the object that appears the most times (the actual task has more demonstration pairs than these three).

# Beyond the Test Error



# Beyond the Test Error

- AI performs human tasks, but currently quite differently than humans
- Understand the data generating process
- Is the data representative?
- Add Explainable Machine Learning to your toolbox
- Choose appropriate statistical tools
- Careful formulation of AI use cases



Thank you!

1.1. or 1.2. or 1.3. or 1.4. in 1.5. in 1.6. & 1.7. f  
2.1. b 2.2. d 2.3. n 2.4. in 2.5. in 2.6. f 2.7. g  
3.1. f 3.2. n 3.3. j 3.4. n  
4.1. w 4.2. o 4.3. o 4.4. p 4.5. q 4.6. r 4.7. f  
5.4. p 5.5. p 5.6. p 5.7. f  
6.7. n 6.8. m 6.9. q 6.7. 8

$\frac{2}{3} + \frac{3}{9} =$   
 $2674318 =$

Basket of apples