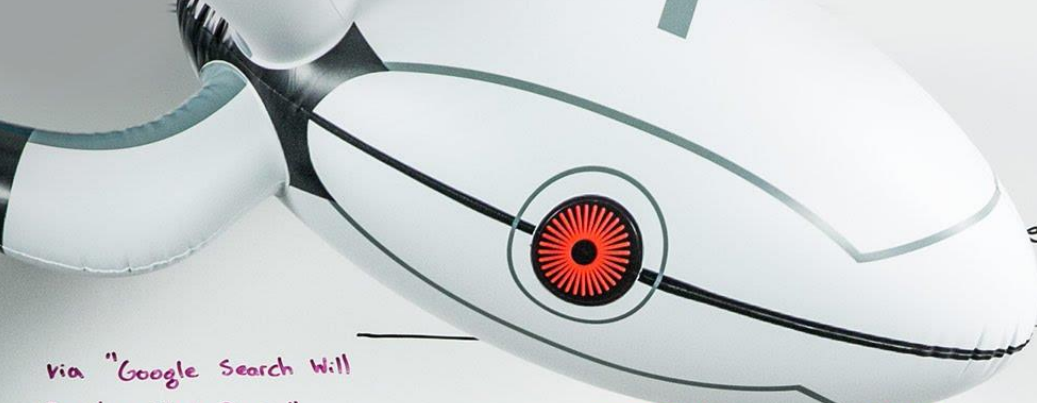


# What Do Deep Learning and Machine Learning Mean For the Future of SEO?





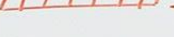



learning


Via "Google Search Will Be Your Next Brain" by Steven Levy on Medium

  
 Geoff Hinton, distinguished engineer, expert in neural nets

"versed in the blackart of organizing several layers of artificial neurons so that the entire system could be trained, or even train itself, to divine coherence from random inputs..."

-  - Learning features
-  - Classify types
-  - Extract or create features/metrics that predict a desired result
-  - Build an algorithm that consistently produces the desired outcome
-  - Grade the results and improve

  
 Originally, you needed people to feed in the inputs and features

  
 With unsupervised learning, the system can extract its own features and improve w/o people



Google Algo Circa 2005



Google Algo

→ Results best correlated with successful searches

What does this mean for SEO?

- Less distinct, more factors
- More complex, less domain
- ...



Recent Trends

Dec. 29<sup>th</sup>, 2016

# Applied Deep Learning

YUN-NUNG (VIVIAN) CHEN

[WWW.CSIE.NTU.EDU.TW/~YVCHEN/F105-ADL](http://WWW.CSIE.NTU.EDU.TW/~YVCHEN/F105-ADL)

# Outline

---

Deep Learning Issues

From “Going Deeper” to “Compressing more lightly”

CNN v.s. RNN

Deep Learning in Robotics

Generative Models

# Outline

---

## **Deep Learning Issues**

From “Going Deeper” to “Compressing more lightly”

CNN v.s. RNN

Deep Learning in Robotics

Generative Models

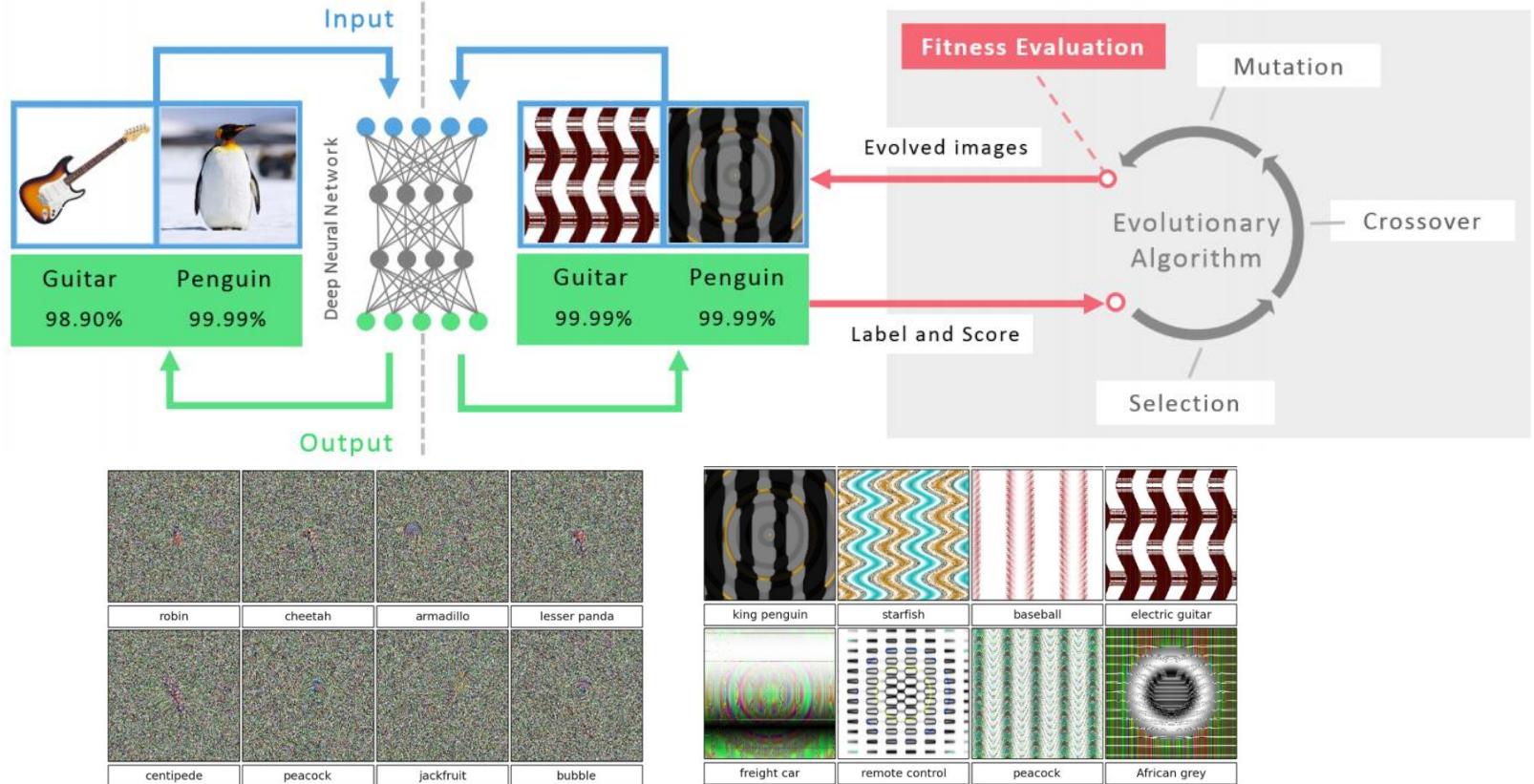
# DNN are easily fooled

1

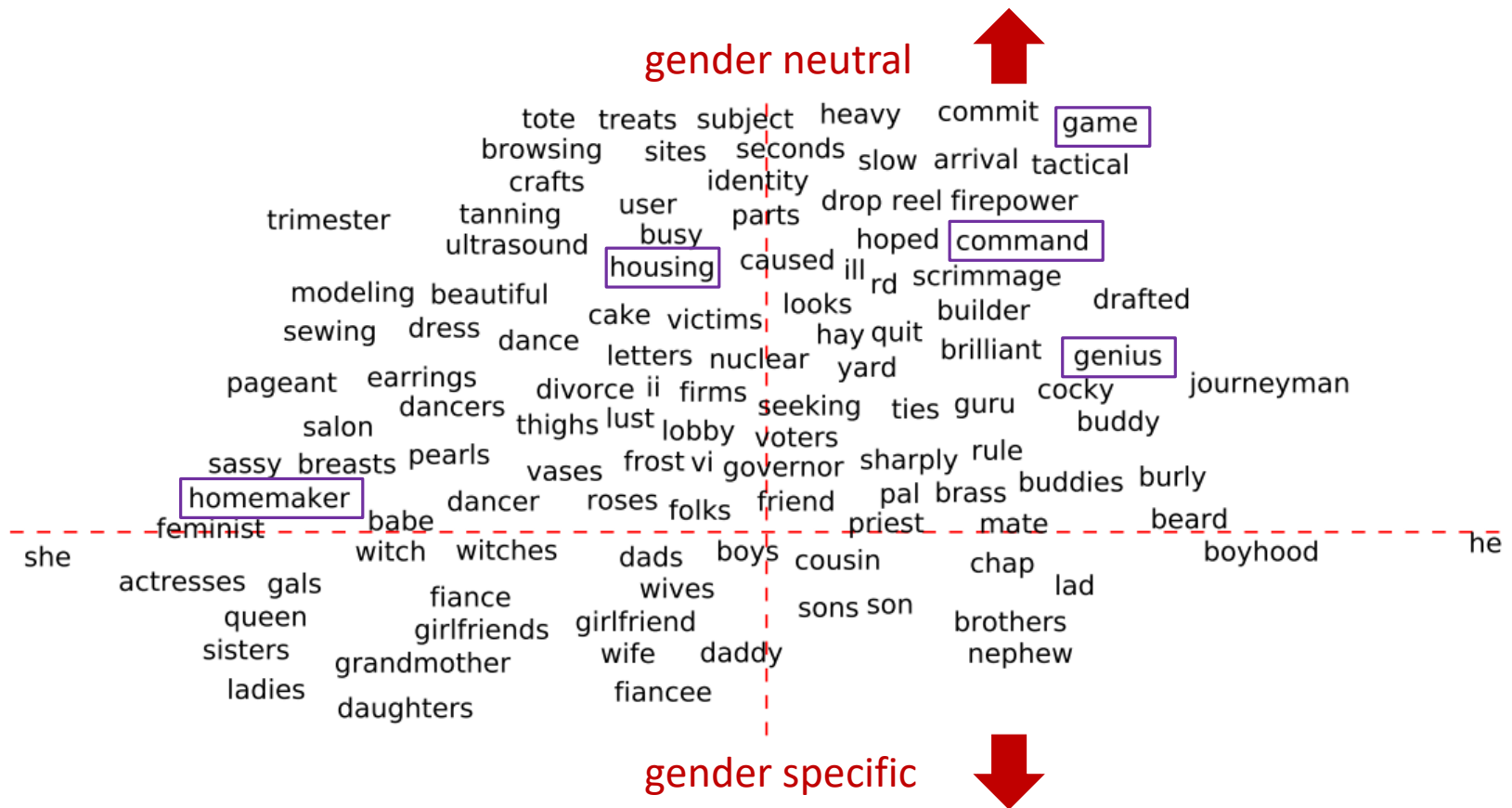
State-of-the-art DNNs can recognize real images with high confidence

2

But DNNs are also easily fooled: images can be produced that are unrecognizable to humans, but DNNs believe with 99.99% certainty are natural objects



# Bias Issue



# Outline

---

Deep Learning Issues

**From “Going Deeper” to “Compressing more lightly”**

CNN v.s. RNN

Deep Learning in Robotics

Generative Models

# Model Compression

---

Deeper models achieve better accuracy: GoogLeNet, ResNet  
SqueezeNet aims at AlexNet-level accuracy with smaller model, motivated by

- i. Smaller CNNs require less communication across servers during distributed training
- ii. less bandwidth to export a new model from the cloud to an autonomous car
- iii. more feasible to deploy on FPGAs and other hardware with limited memory

# Outline

---

Deep Learning Issues

From “Going Deeper” to “Compressing more lightly”

**CNN v.s. RNN**

Deep Learning in Robotics

Generative Models



# CNN v.s. RNN

---

CNN is also able to capture knowledge from windowed data for sequential input

RNN mainly benefits from learning the sequential information → order matters

Efficiency? Effectiveness? Combination?

# Outline

---

Deep Learning Issues

From “Going Deeper” to “Compressing more lightly”

CNN v.s. RNN

**Deep Learning in Robotics**

Generative Models

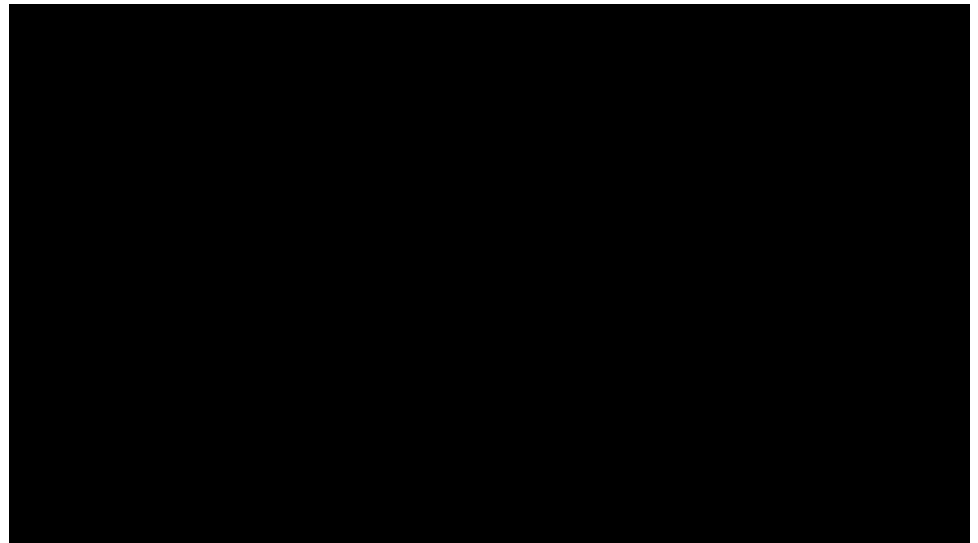
# Deep Learning in Robotics

---

Traditional: humanoids or manipulators

Trend: policy search framework / reinforcement learning

- Failure cost is expensive
  - Cloud robotics
  - Learning from demonstration
- Decisions should be made in real time
  - Model size reduction
  - Transferring learned knowledge for a new task



# Outline

---

Deep Learning Issues

From “Going Deeper” to “Compressing more lightly”

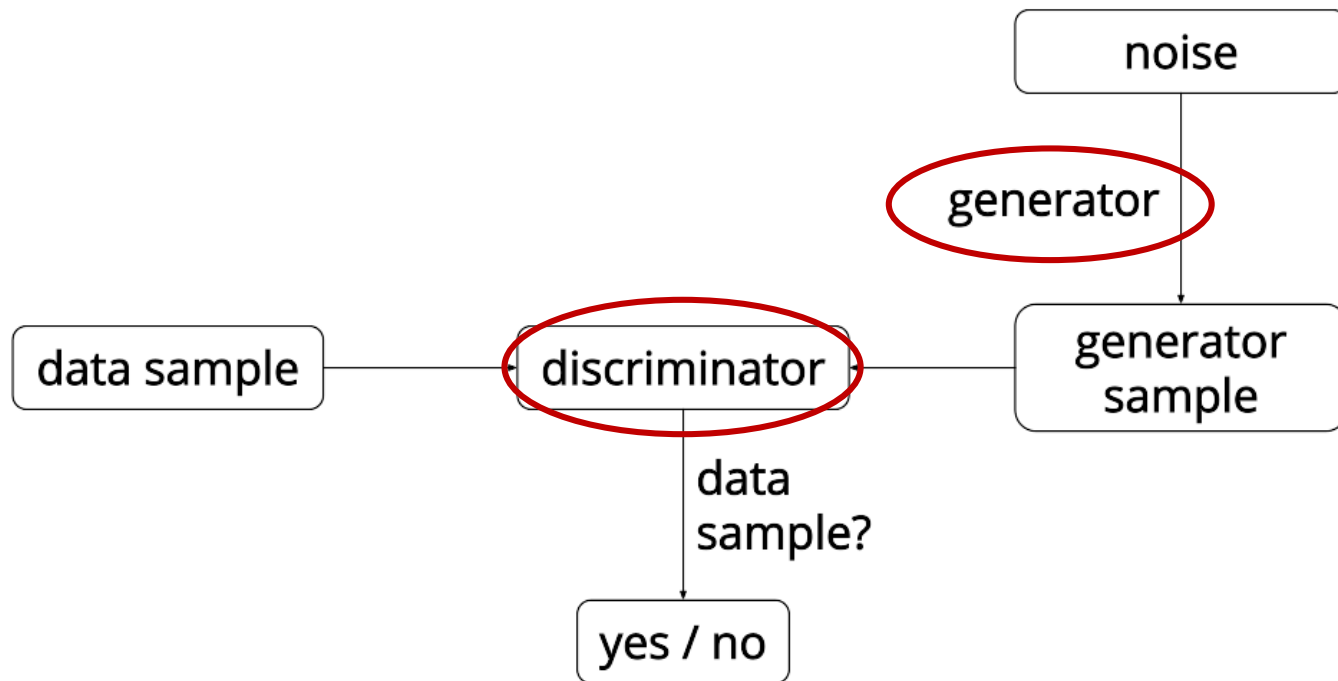
CNN v.s. RNN

Deep Learning in Robotics

**Generative Models**

# Generative Adversarial Networks (GAN)

---



Two players reach a Nash equilibrium to produce an optimal generator

# Concluding Remarks

---

## Deep Learning Issues

- DNNs are easily fooled
- Bias issue

## From “Going Deeper” to “Compressing more lightly”

- less communication
- less bandwidth to export
- feasible to deploy

## CNN v.s. RNN

## Deep Learning in Robotics: policy search/reinforcement learning

- Expensive cost: cloud robot, learning from demonstration
- Real-time decision: model compression, knowledge transfer

## Generative Models: GAN

- Adversarial framework: generator & discriminator