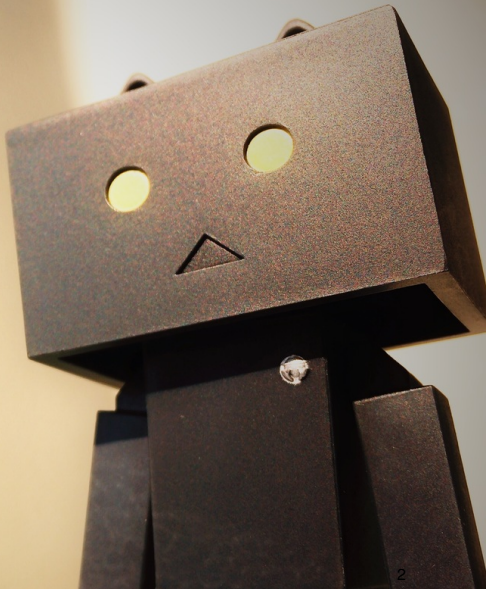
WTF am I Doing?

Jeff Abrahamson

7 May 2017

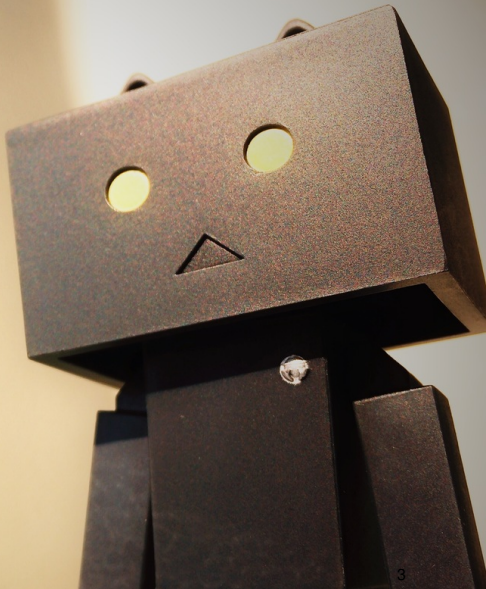
The problem

Understand my behaviour



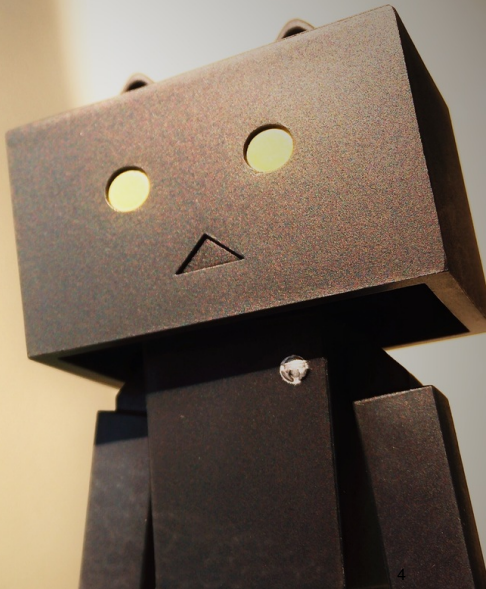
The problem

Predict



The problem

But it's a hobby. . .



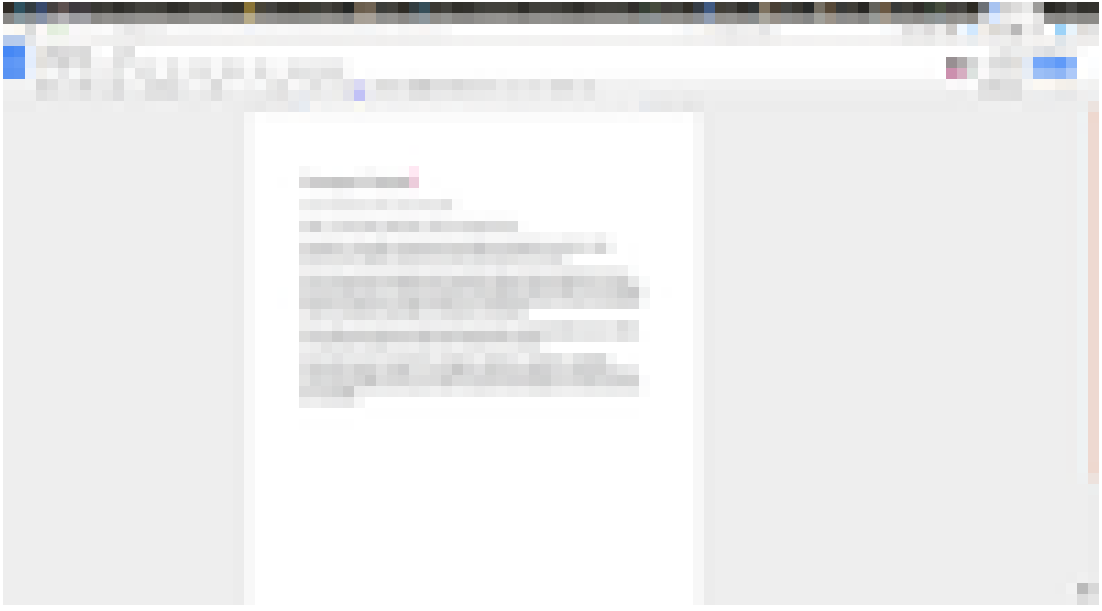

Where’s the data?

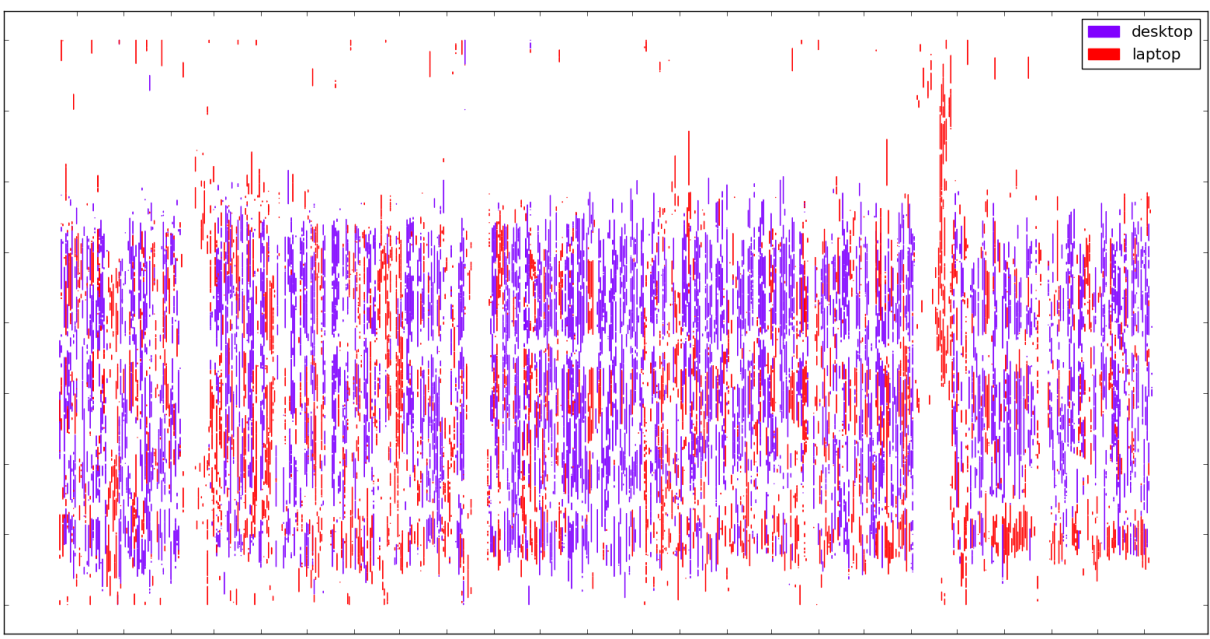

Where’s the data?
My computer

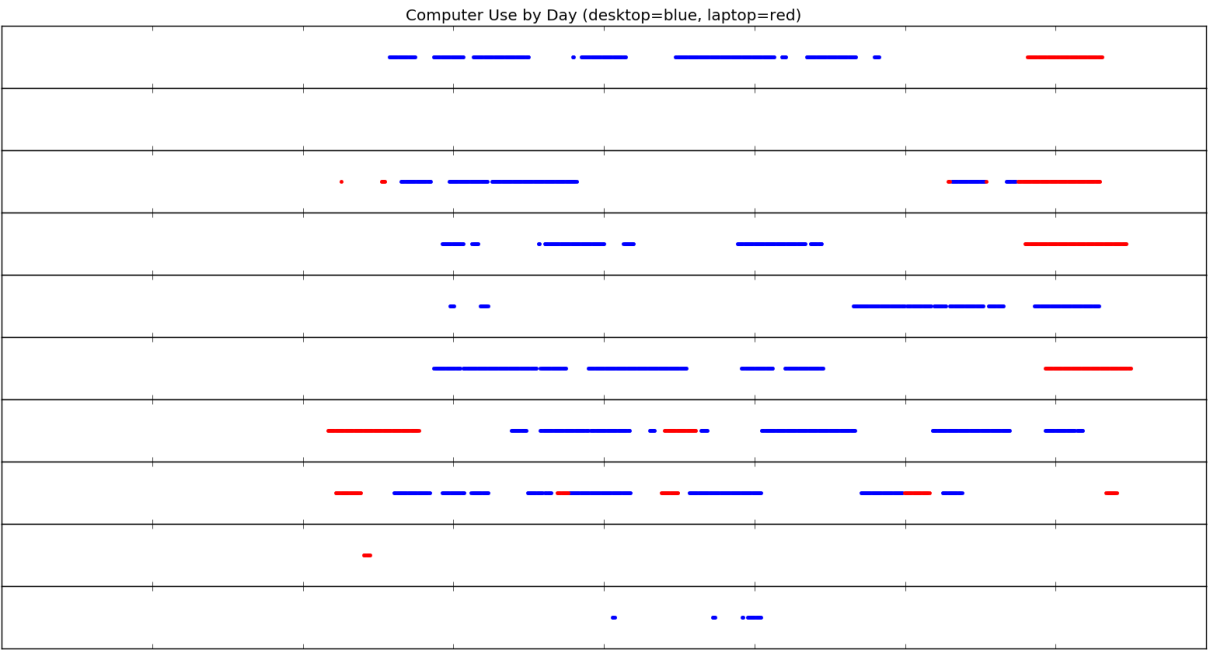

Where’s the data?
Window titles

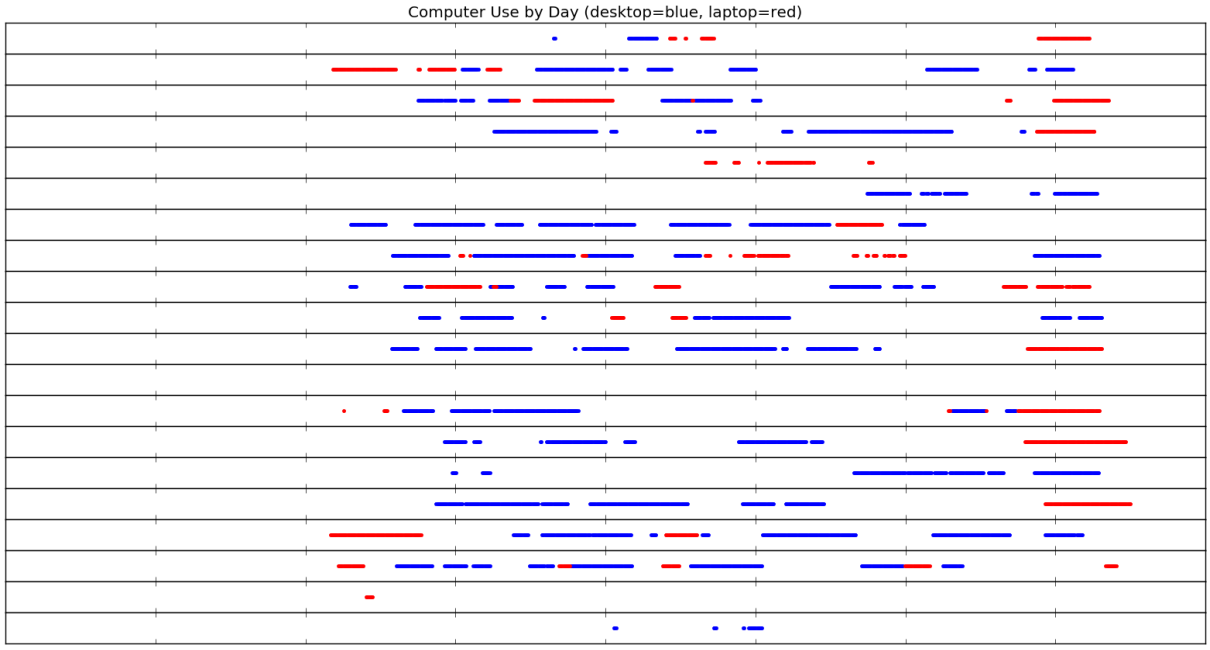

Where’s the data?
Window thumbnails

1493904959 emacs@birdsong - talk.tex : /home/jeff/src/jma/talks/2017-05__PyDataLondon/talk.tex









supervised

unsupervised

model

General strategy: features

General strategy: labeling

General strategy: auto-labeling

General strategy: sequences

General strategy: prediction

stategy 1 : bag of words

stategy 1 : tf-idf

stategy 2 : word2vec

two strategies: find a sequence

Bag of Words

```
The cat is orange.
The dog runs fast.
```

Bag of Words

```
6   1    7    2
The cat is orange.
The dog runs fast.
6    3     4    5
```

Bag of Words

6 1 7 2

The cat is orange.

The dog runs fast.

6 3 4 5

Bag of Words

```
[[6, 1, 7, 2],

 [6, 3, 4, 5]]
```

Bag of Words

```
[[6, 1, 7, 2]
 [1, 1, 0, 0, 0, 1, 1]
 [0, 0, 1, 1, 1, 1, 0]
 [6, 3, 4, 5]]
```

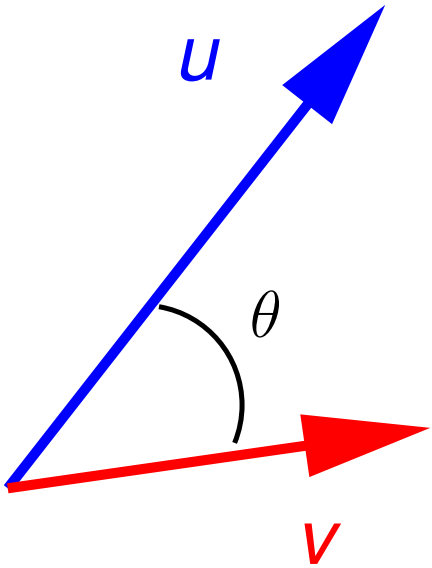
Bag of Words

```
[1, 1, 0, 0, 0, 1, 1]
[0, 0, 1, 1, 1, 1, 0]
```

Bag of Words

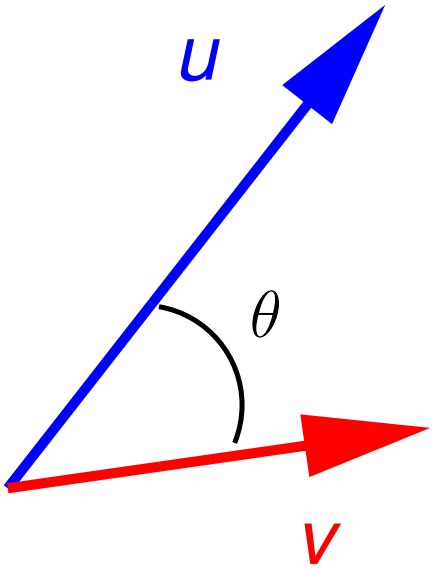
```
The cat is orange.
 [1, 1, 0, 0, 0, 1, 1]
 [0, 0, 1, 1, 1, 1, 0]
The dog runs fast.
```

Cosine Similarity



$$\cos\theta = \frac{u \cdot v}{\| u \| \| v \|}$$

Cosine Similarity



$$\cos\theta = u \cdot v$$

(if u and v have norm 1)

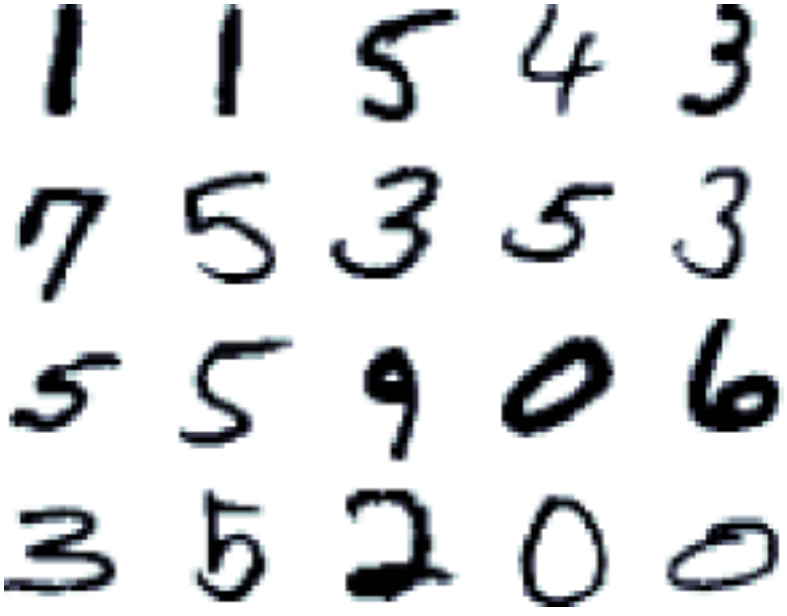
Cosine Similarity

```
The cat is orange.
 [1, 1, 0, 0, 0, 1, 1]
 [0, 0, 1, 1, 1, 1, 0]
The dog runs fast.
```

Cosine Similarity

$$u = [1,\ 1,\ 0,\ 0,\ 0,\ 1,\ 1]$$
$$v = [0,\ 0,\ 1,\ 1,\ 1,\ 1,\ 0]$$
$$u \cdot v = 0 + 0 + 0 + 0 + 0 + 1 + 0 = 1$$

$$\cos\theta = \frac{u \cdot v}{\| u \| \| v \|} = \frac{1}{\sqrt{4} \cdot \sqrt{4}} = \frac{1}{4}$$



A far too short example

Linear neuron

$$y = b + \sum_i x_i w_i$$

Linear neuron

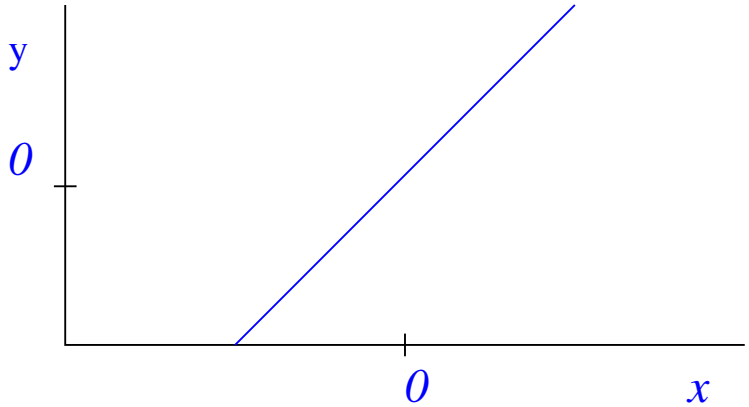
$$y = b + \sum_i x_i w_i$$

where

$$y = \text{output}$$
$$b = \text{bias}$$
$$x_i = i^{\text{th}} \text{ input}$$
$$w_i = \text{weight on } i^{\text{th}} \text{ input}$$

Linear neuron

$$y = b + \sum_i x_i w_i$$



Example: handwriting recognition of digits

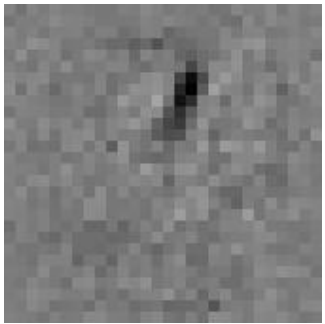
- Input neurons: pixels
- Output neurons: classes (digits)
- Connect them all! *(bipartite)*

Example: handwriting recognition of digits

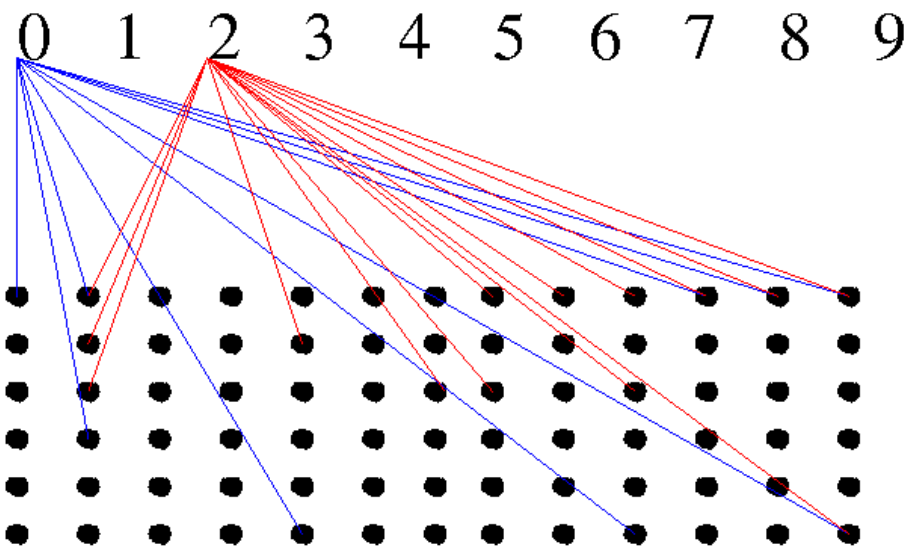
- Input neurons: pixels
- Output neurons: classes (digits)
- Connect them all! *(bipartite)*
- Initialize input weights to random

Example: handwriting recognition of digits

- Input neurons: pixels
- Output neurons: classes (digits)
- Connect them all! *(bipartite)*
- Initialize input weights to random



Example: handwriting recognition of digits



To train this ANN:

- Increment weights from active pixels going to correct class
- Decrement weights from active pixels going to predicted class

To train this ANN:

- Increment weights from active pixels going to correct class
- Decrement weights from active pixels going to predicted class

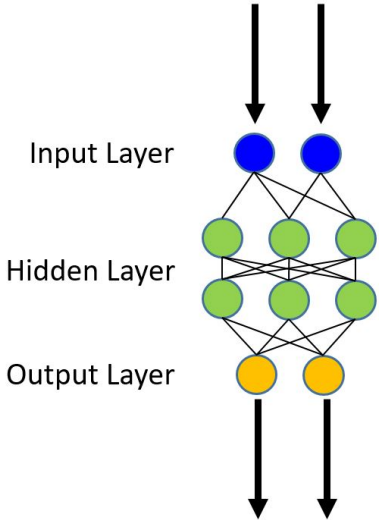
When it's right, nothing happens. This is good.

stategy 1 : bag of words

stategy 1 : tf-idf

stategy 2 : word2vec

two strategies: find a sequence

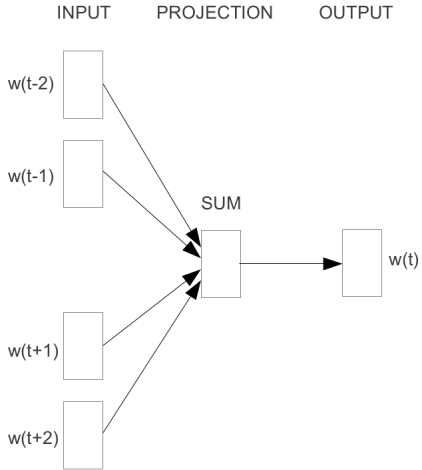


stategy 1 : bag of words

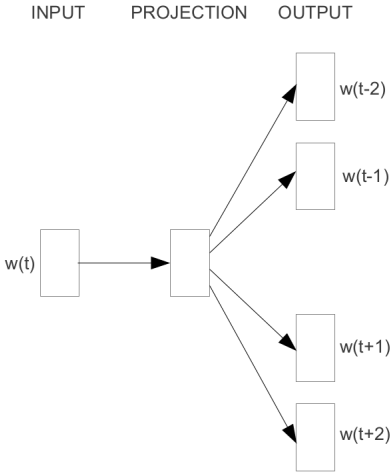
stategy 1 : tf-idf

stategy 2 : word2vec

two strategies: find a sequence

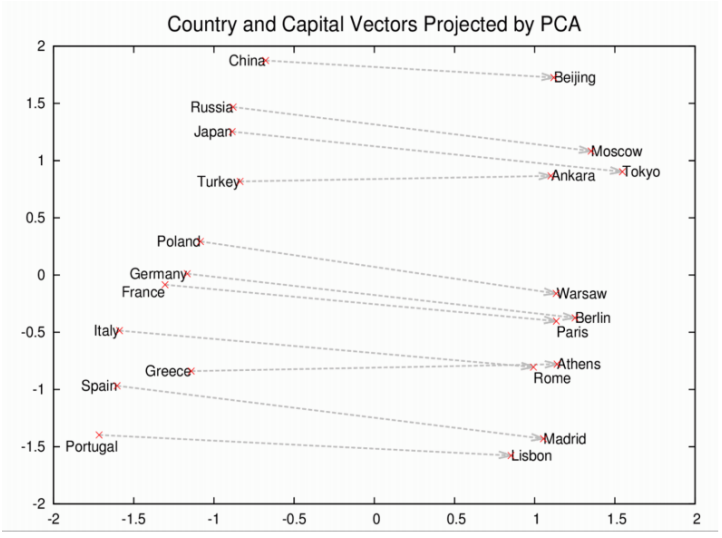


CBOW



Skip-gram

Google



Google

	Term	Similarity
	"shift"	0.933104
	"gown"	0.887743
	"skirt"	0.881672
	"bandage"	0.880162
	"midi"	0.869786

Similar to ‘dress’

Lyst, courtesy Eddie Bell

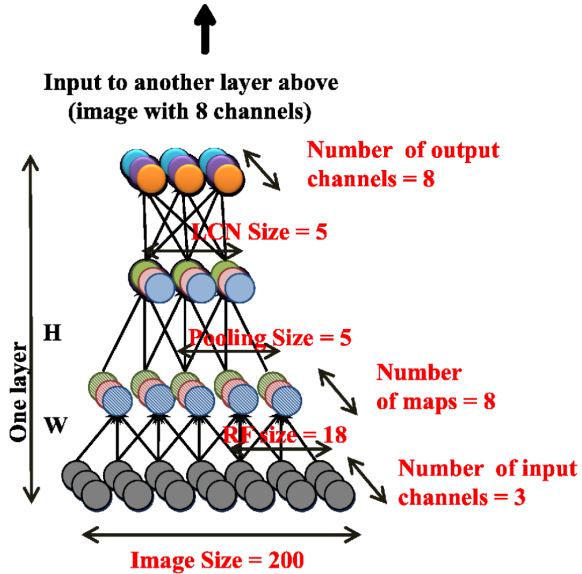
General strategy: features

General strategy: labeling

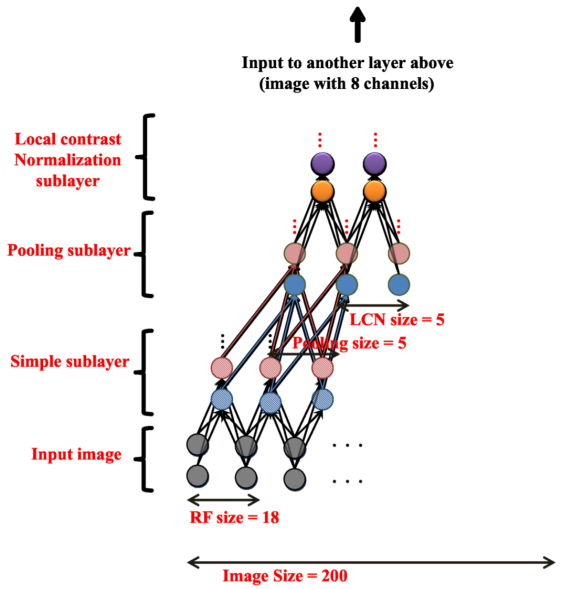
General strategy: auto-labeling

General strategy: sequences

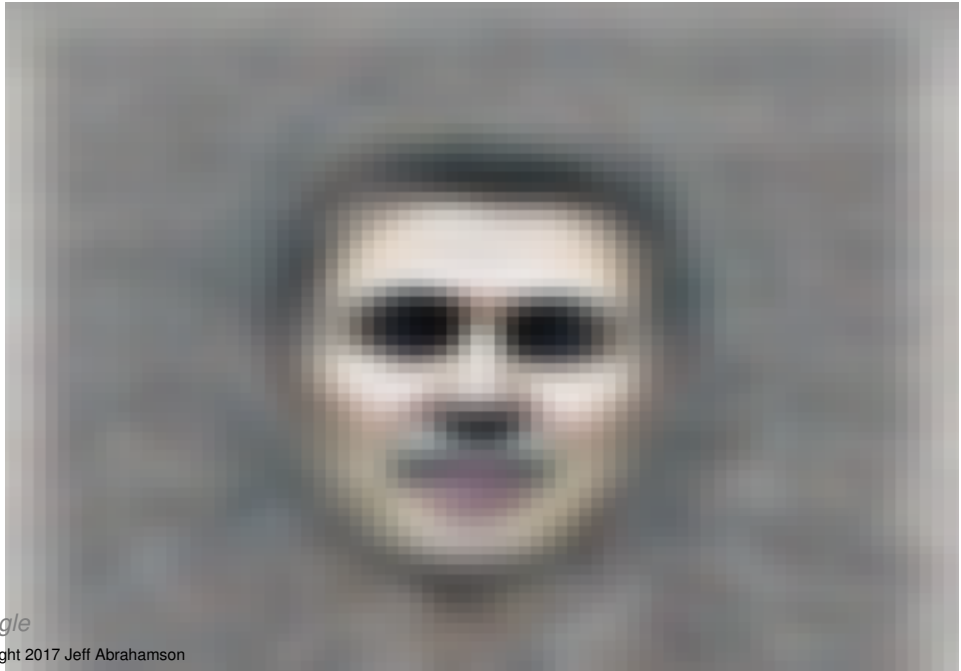
General strategy: prediction



Google



Google



Google

Google

$$a_1, a_2, \ldots, a_k \Rightarrow a_{k+1}$$