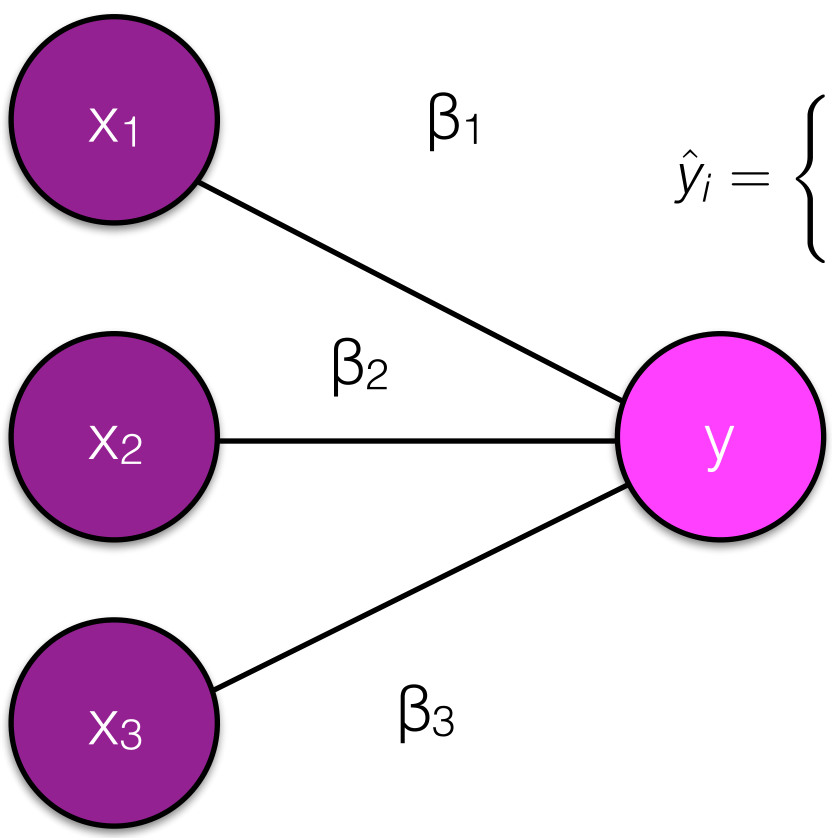
Deconstructing Data Science

David Bamman, UC Berkeley

Info 290
Lecture 17: Neural networks (2)

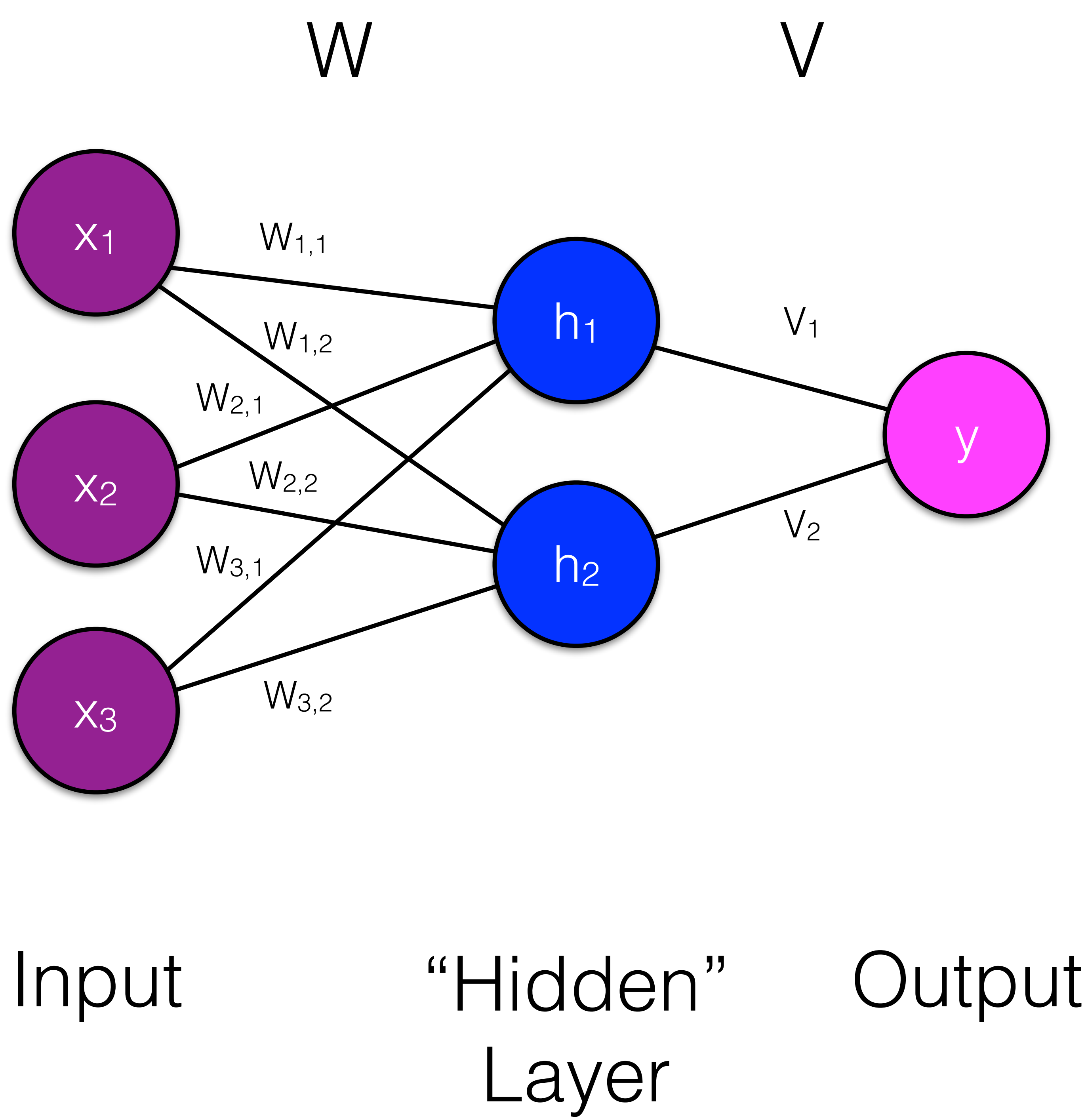
Mar 21, 2017

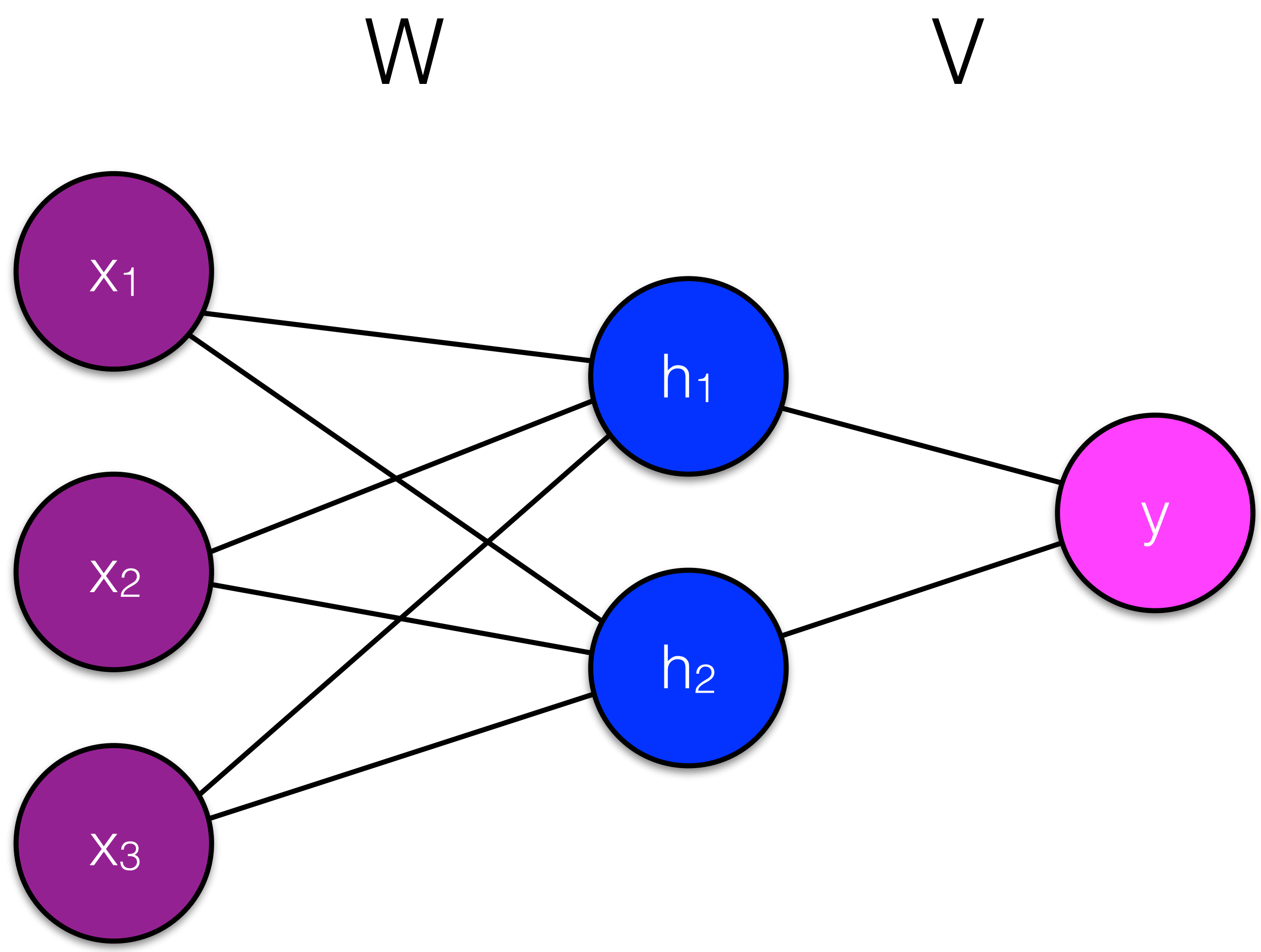
The perceptron, again



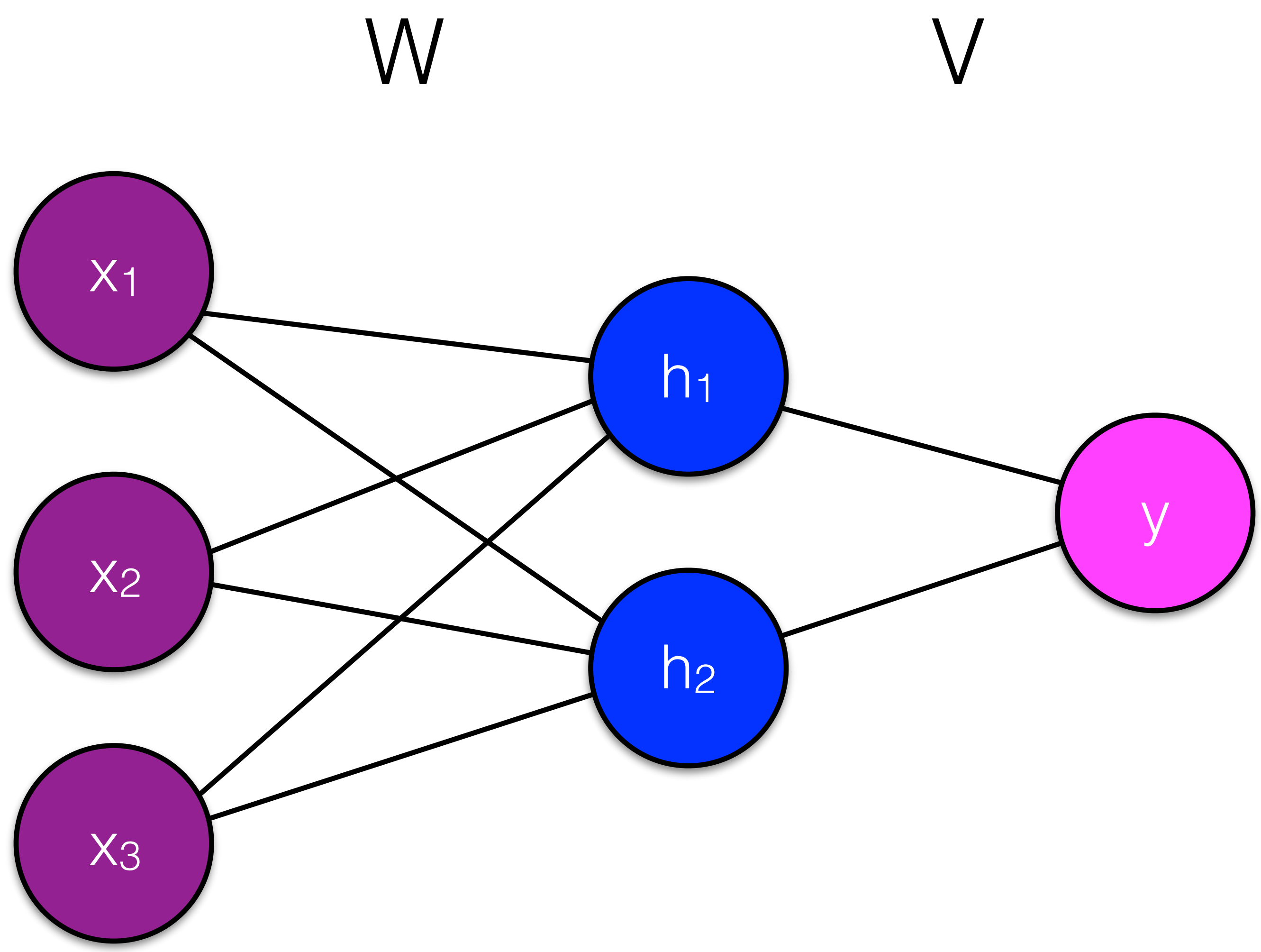
$$\hat{y}_i = \begin{cases} 1 & \text{if } \sum_i^F x_i\beta_i \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

	x	β
not	1	-0.5
bad	1	-1.7
movie	0	0.3


W
V
x1
x2
x3
W1,1
W1,2
W2,1
W2,2
W3,1
W3,2
h1
h2
V1
V2
y
Input
"Hidden" Layer
Output

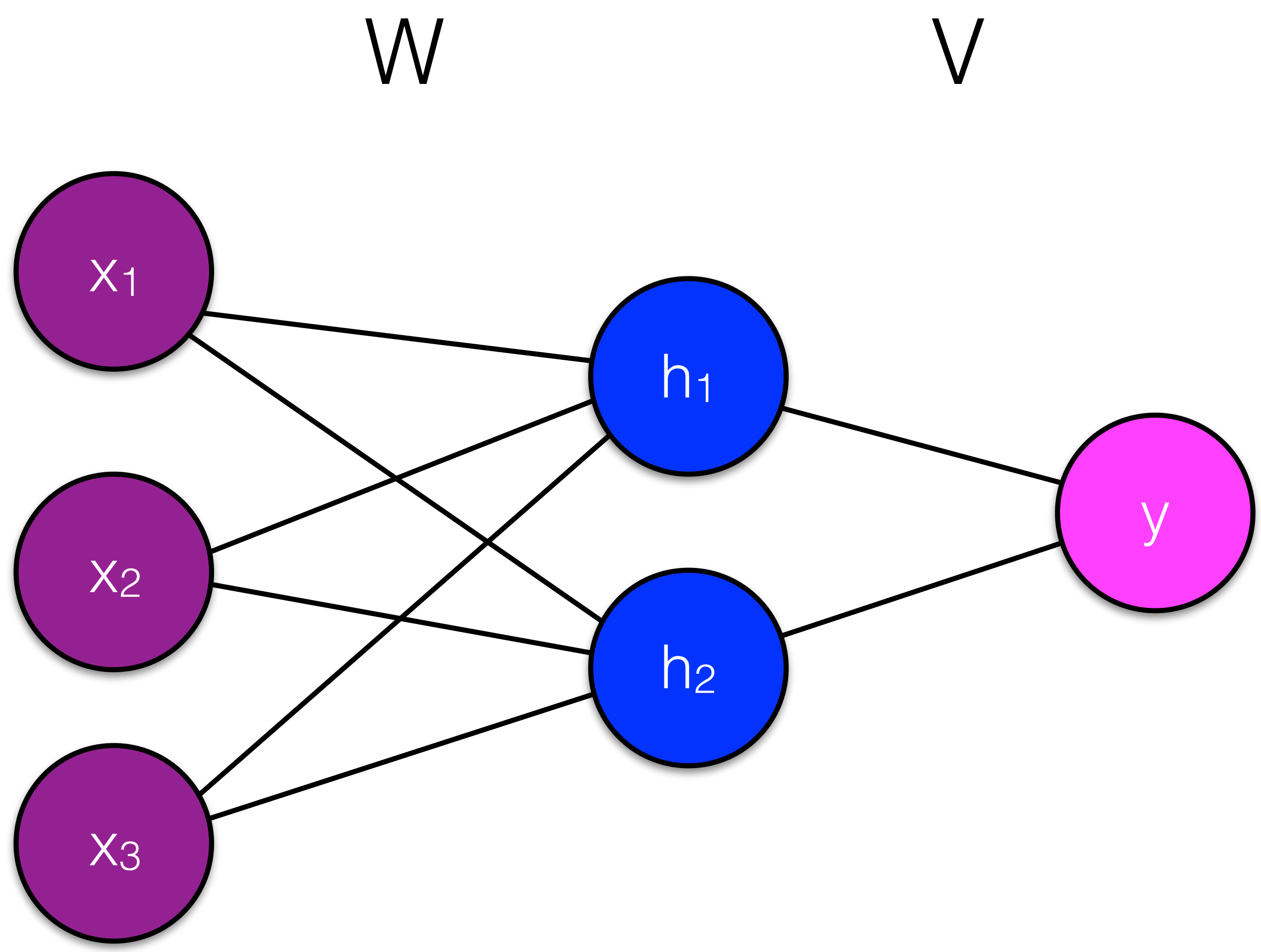


	x	W		V	y
not	1	-0.5	1.3	4.1	-1
bad	1	0.4	0.08	-0.9	
movie	0	1.7	3.1		



$$h_j = f\left(\sum_{i=1}^{F} x_i W_{i,j}\right)$$

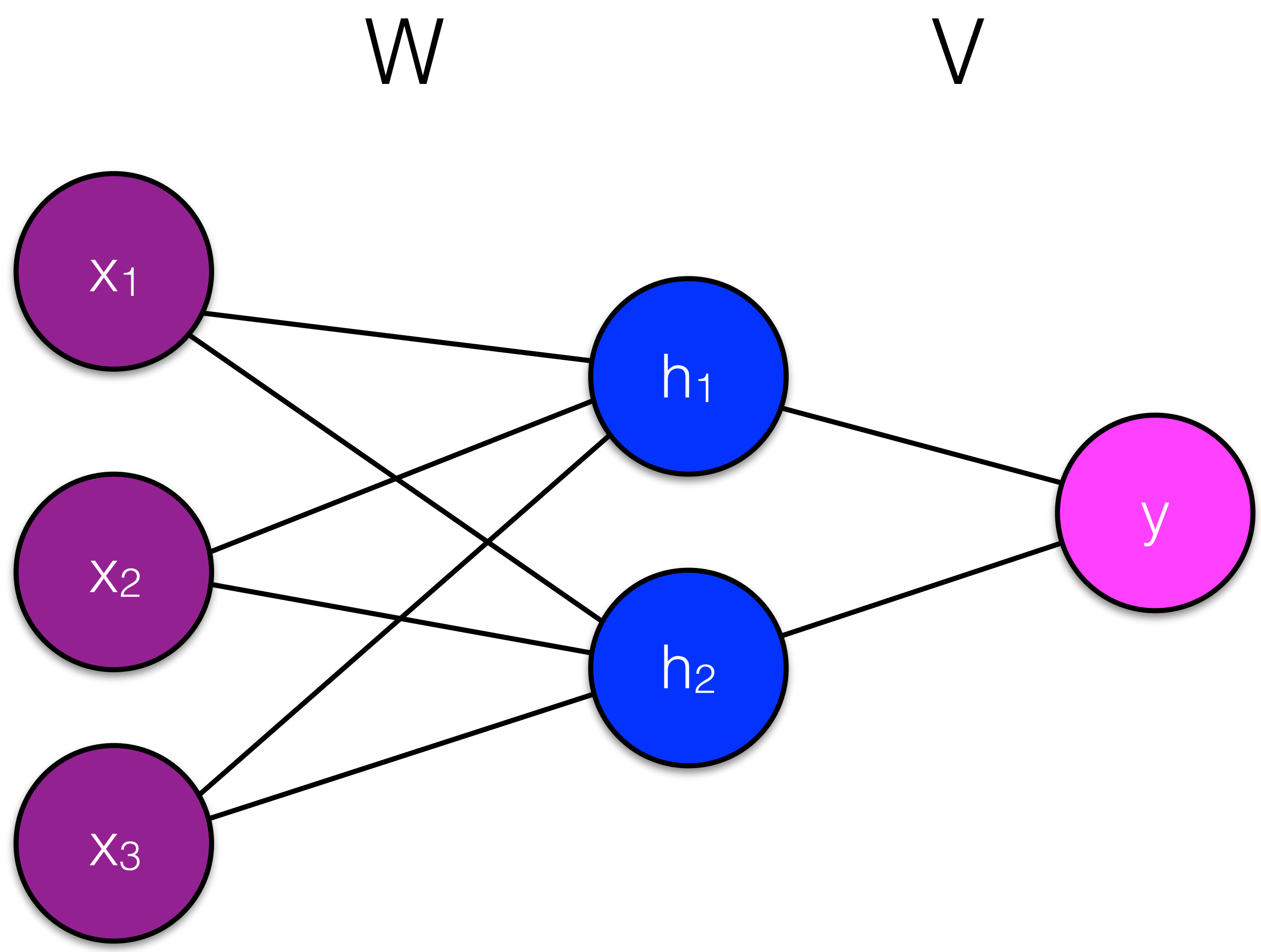
the hidden nodes are completely determined by the input and weights



$$h_1 = \sigma\left(\sum_{i=1}^{F} x_i W_{i,1}\right)$$

$$\hat{y} = V_1 h_1 + V_2 h_2$$

$$h_2 = \sigma\left(\sum_{i=1}^{F} x_i W_{i,2}\right)$$



$$\hat{y} = V_1 \underbrace{\left(\sigma\left(\sum_{i=1}^{F} x_i W_{i,1}\right)\right)}_{h_1} + V_2 \underbrace{\left(\sigma\left(\sum_{i=1}^{F} x_i W_{i,2}\right)\right)}_{h_2}$$

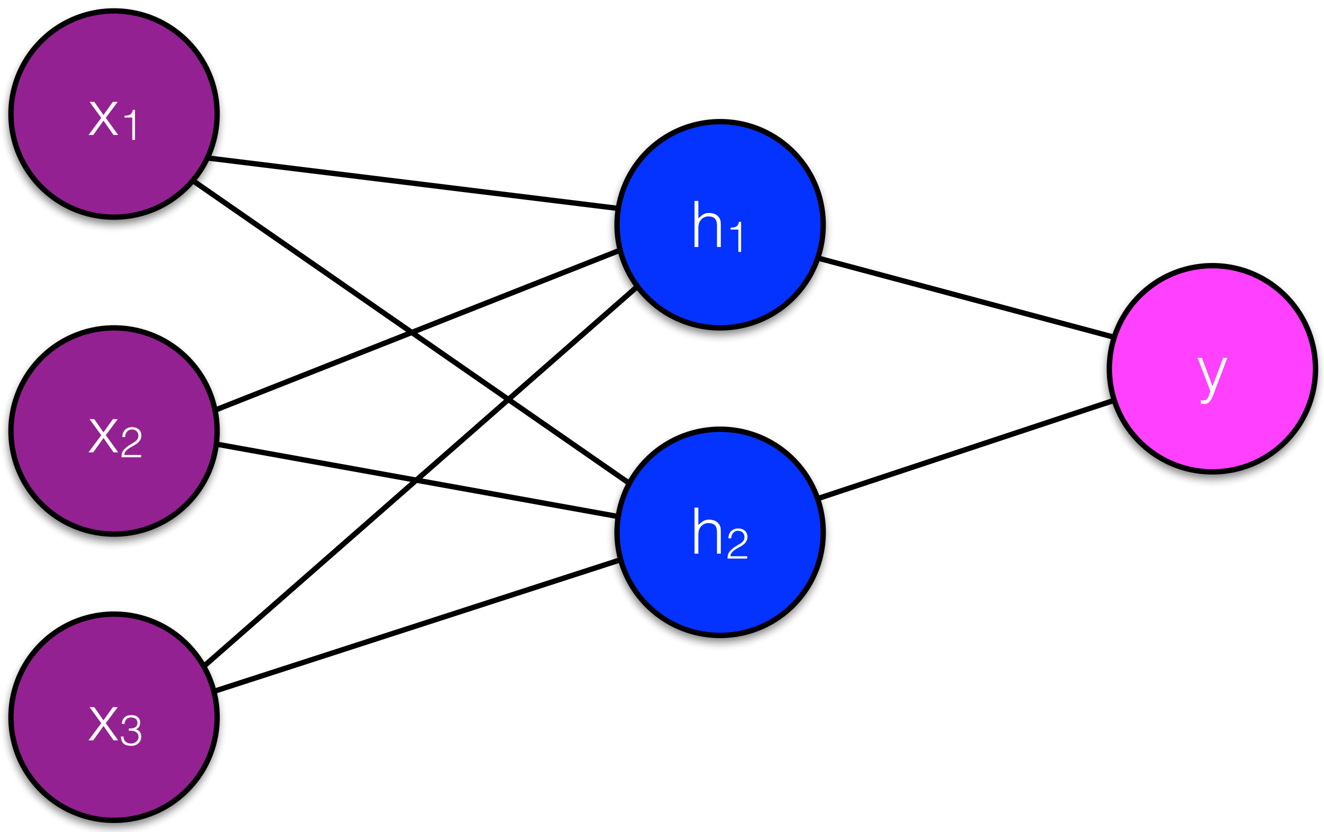
we can express y as a function only of the input x and the weights W and V

$$\hat{y} = V_1 \underbrace{\left(\sigma\left(\sum_{i=1}^{F} x_i W_{i,1}\right)\right)}_{h_1} + V_2 \underbrace{\left(\sigma\left(\sum_{i=1}^{F} x_i W_{i,2}\right)\right)}_{h_2}$$

This is hairy, but differentiable

Backpropagation: Given training samples of <x,y> pairs, we can use stochastic gradient descent to find the values of W and V that minimize the loss.

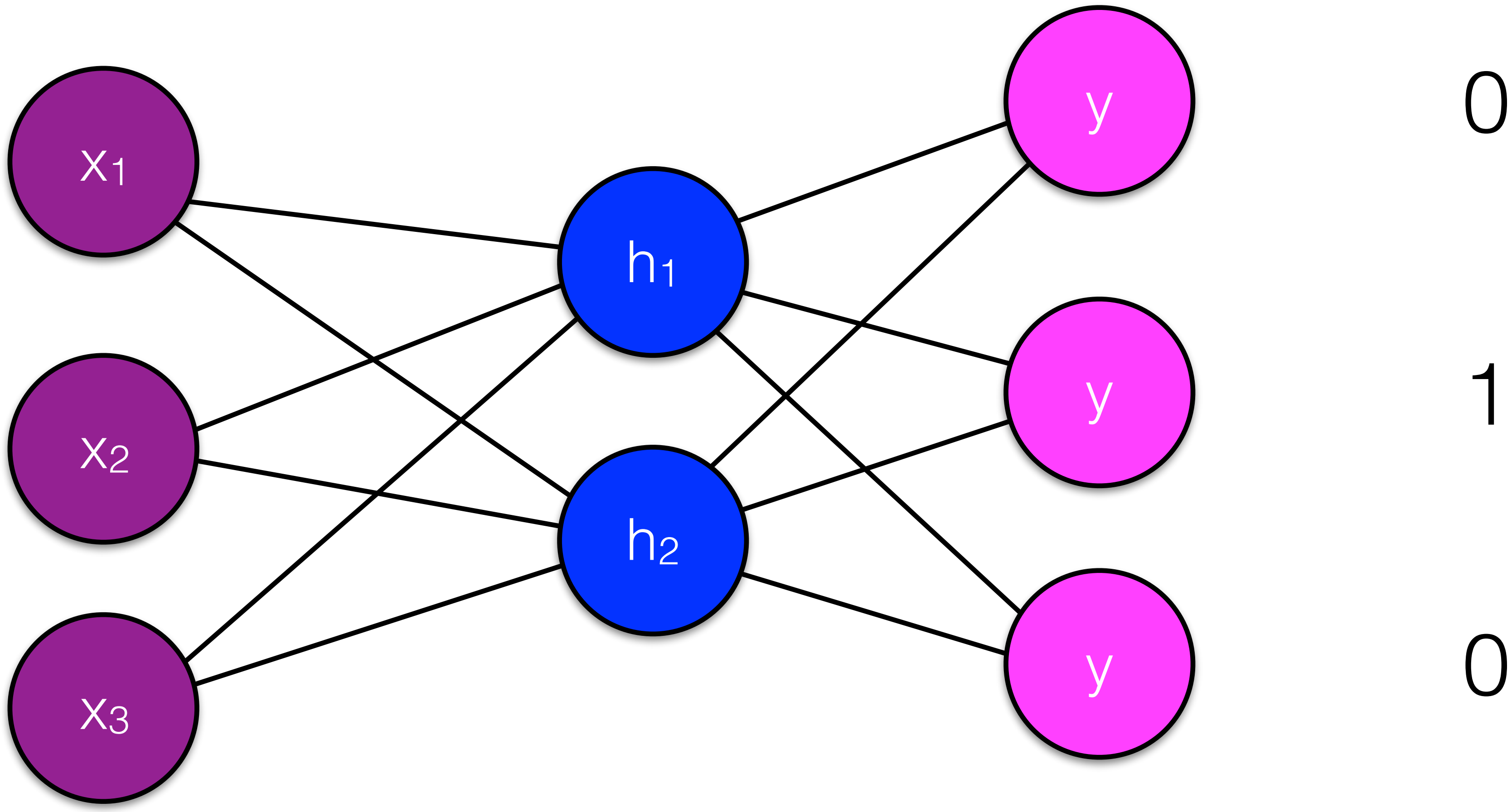
Neural network structures



1

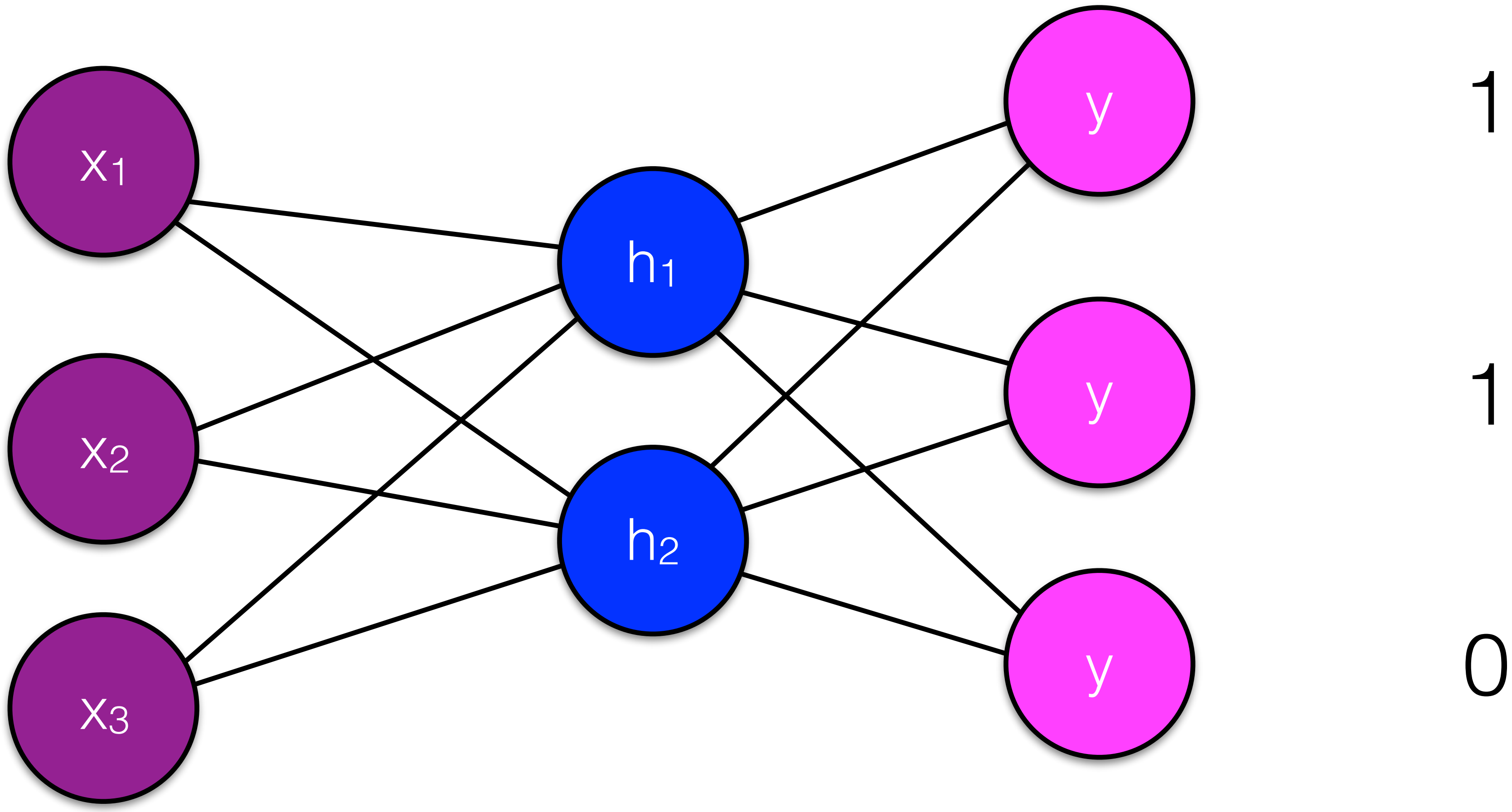
Output one real value

Neural network structures



Multiclass: output 3 values, only one = 1 in training data

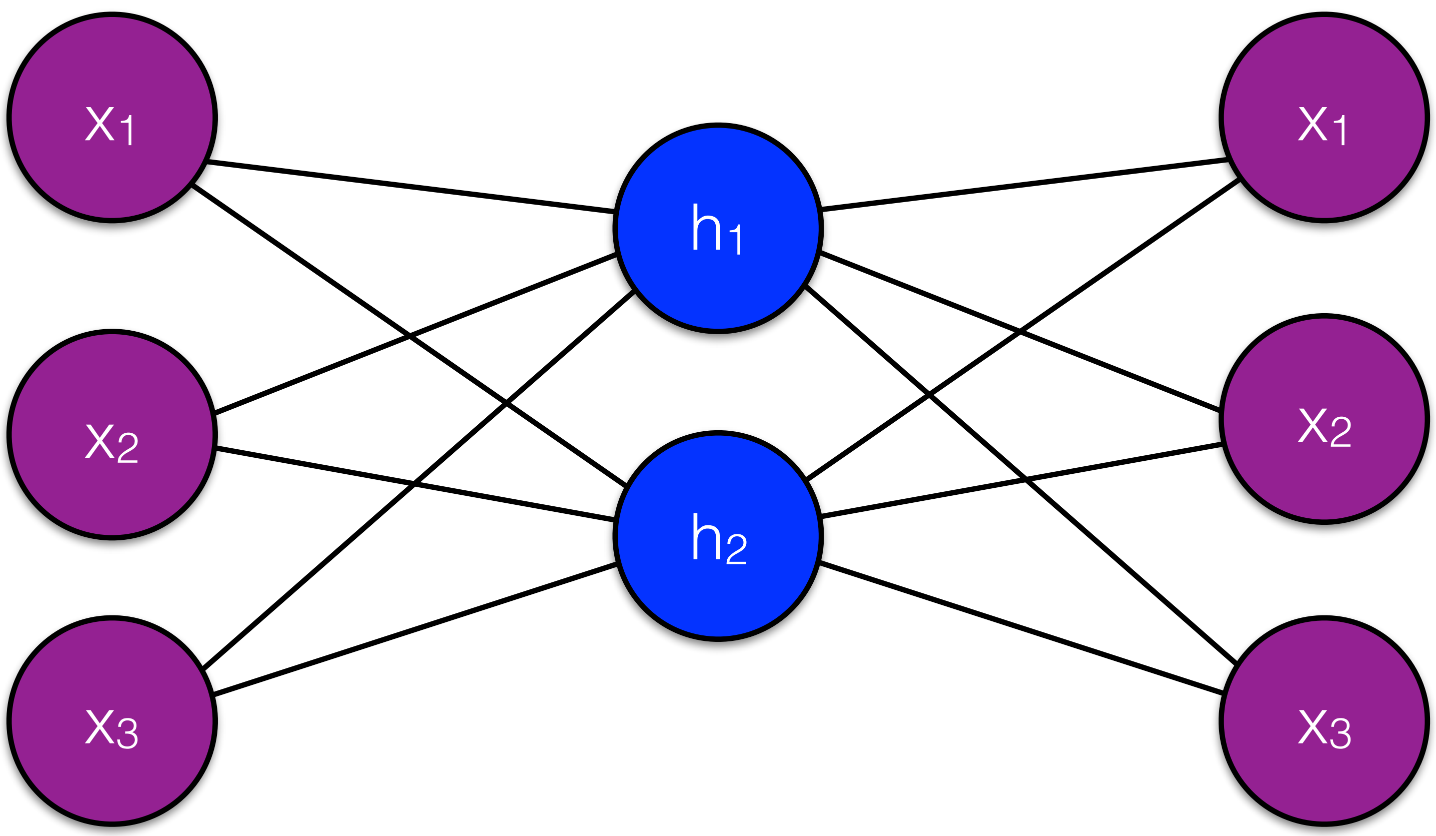
Neural network structures



output 3 values, several = 1 in training data

Autoencoder

- Unsupervised neural network, where y = x
- Learns a low-dimensional representation of x



Word embeddings

- Learning low-dimensional representations of words by framing a predicting task: using context to predict words in a surrounding window

the black cat jumped on the table

Dimensionality reduction

…	…
the	1
a	0
an	0
for	0
in	0
on	0
dog	0
cat	0
…	…

the is a point in V-dimensional space

the

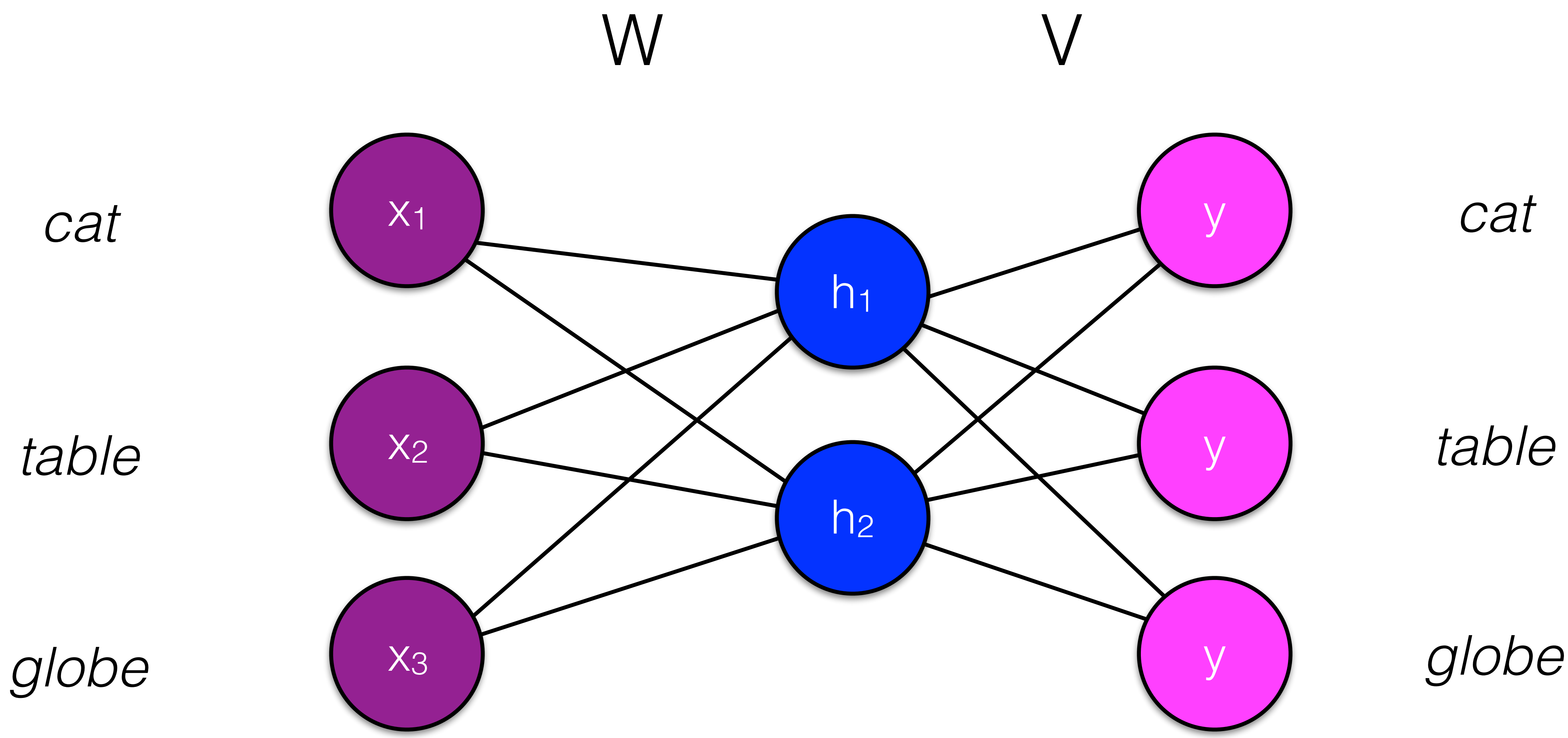
4.1
-0.9

the is a point in 2-dimensional space

Word embeddings

- Transform this into a supervised prediction problem

x	y
the	cat
black	cat
jumped	cat
on	cat
the	cat
table	cat

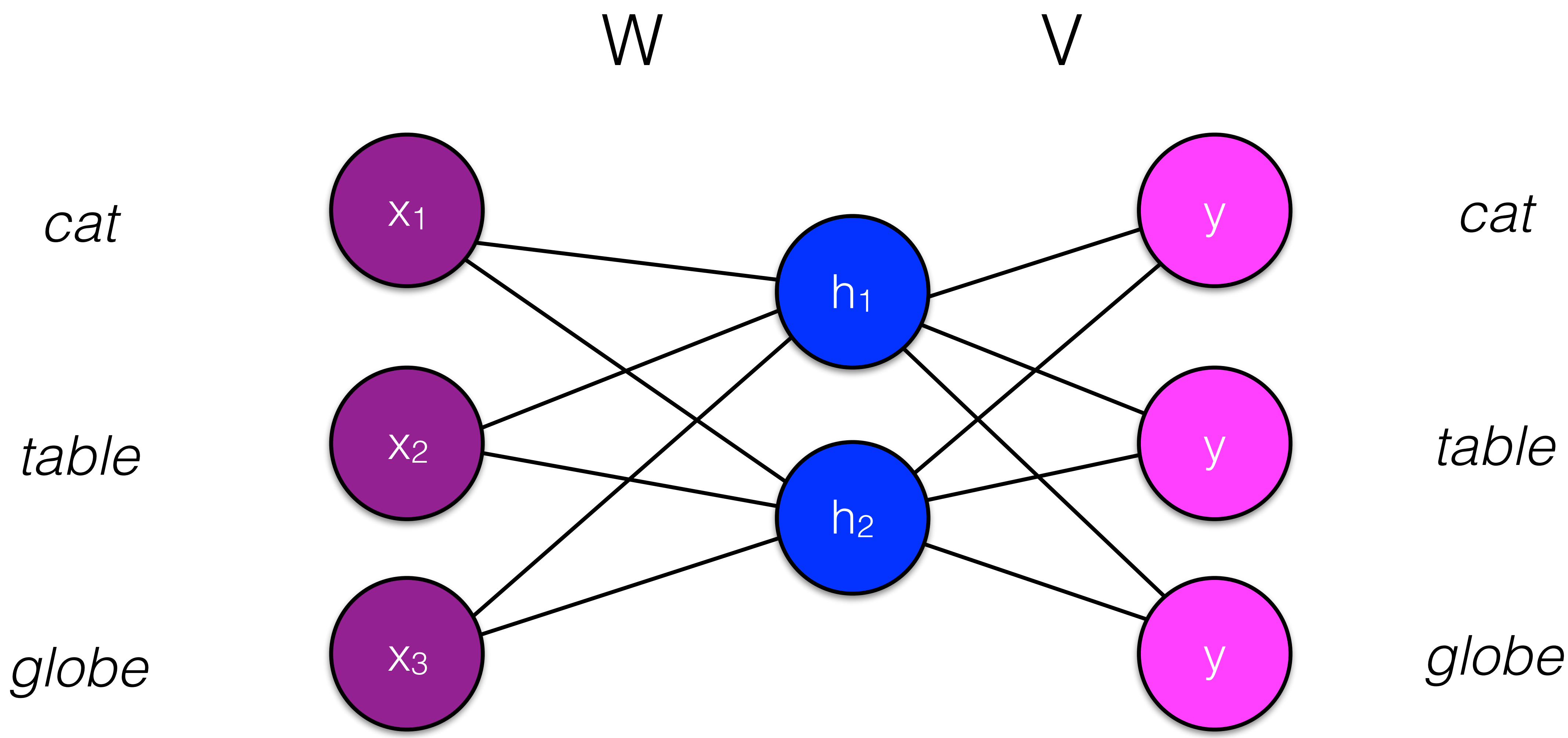


	x
cat	0
table	1
globe	0

W	
-0.5	1.3
0.4	0.08
1.7	3.1

V		
4.1	0.7	0.1
-0.9	1.3	0.3

y
1
0
0



Only one of the inputs is nonzero.

= the inputs are really W_{table}

W	
-0.5	1.3
0.4	0.08
1.7	3.1

V		
4.1	0.7	0.1
-0.9	1.3	0.3

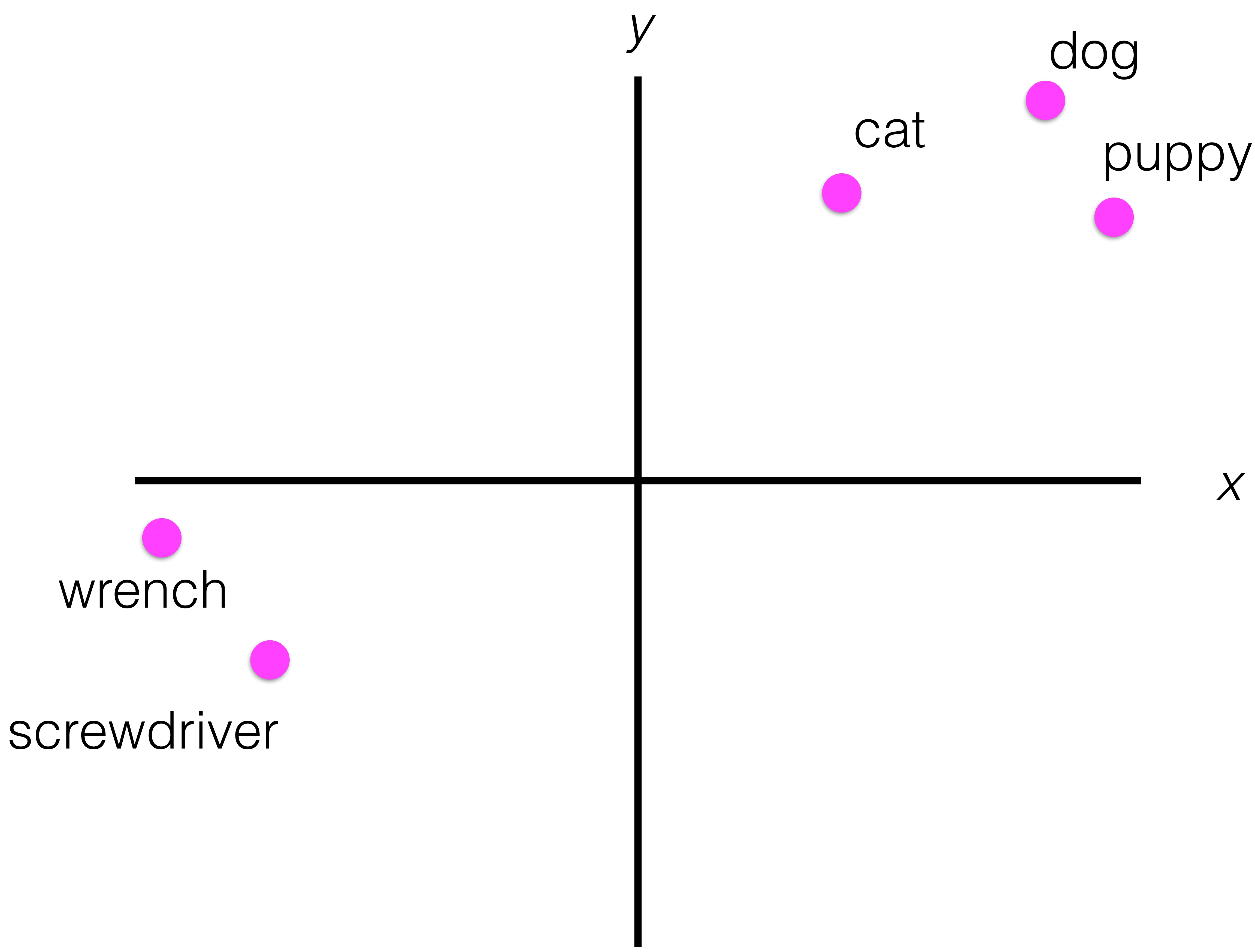
Word embeddings

- Can you predict the output word from a vector representation of the input word?

Word embeddings

- Output: low-dimensional representation of words directly read off from the weight matrices.

V		
cat	table	globe
4.1	0.7	0.1
-0.9	1.3	0.3


y
dog
cat
puppy
x
wrench
screwdriver

- Why this behavior? *dog*, *cat* show up in similar positions

the	black	cat	jumped	on	the	table
the	black	dog	jumped	on	the	table
the	black	puppy	jumped	on	the	table
the	black	skunk	jumped	on	the	table
the	black	shoe	jumped	on	the	table

- Why this behavior? *dog*, *cat* show up in similar positions

the	black	[0.4, 0.08]	jumped	on	the	table
the	black	[0.4, 0.07]	jumped	on	the	table
the	black	puppy	jumped	on	the	table
the	black	skunk	jumped	on	the	table
the	black	shoe	jumped	on	the	table

To make the same predictions, these numbers need to be close to each other.

Dimensionality reduction

…	…
the	1
a	0
an	0
for	0
in	0
on	0
dog	0
cat	0
…	…

the is a point in V-dimensional space;
representations for all words are completely independent

the

4.1
-0.9

the is a point in 2-dimensional space
representations are now structured

Euclidean distance

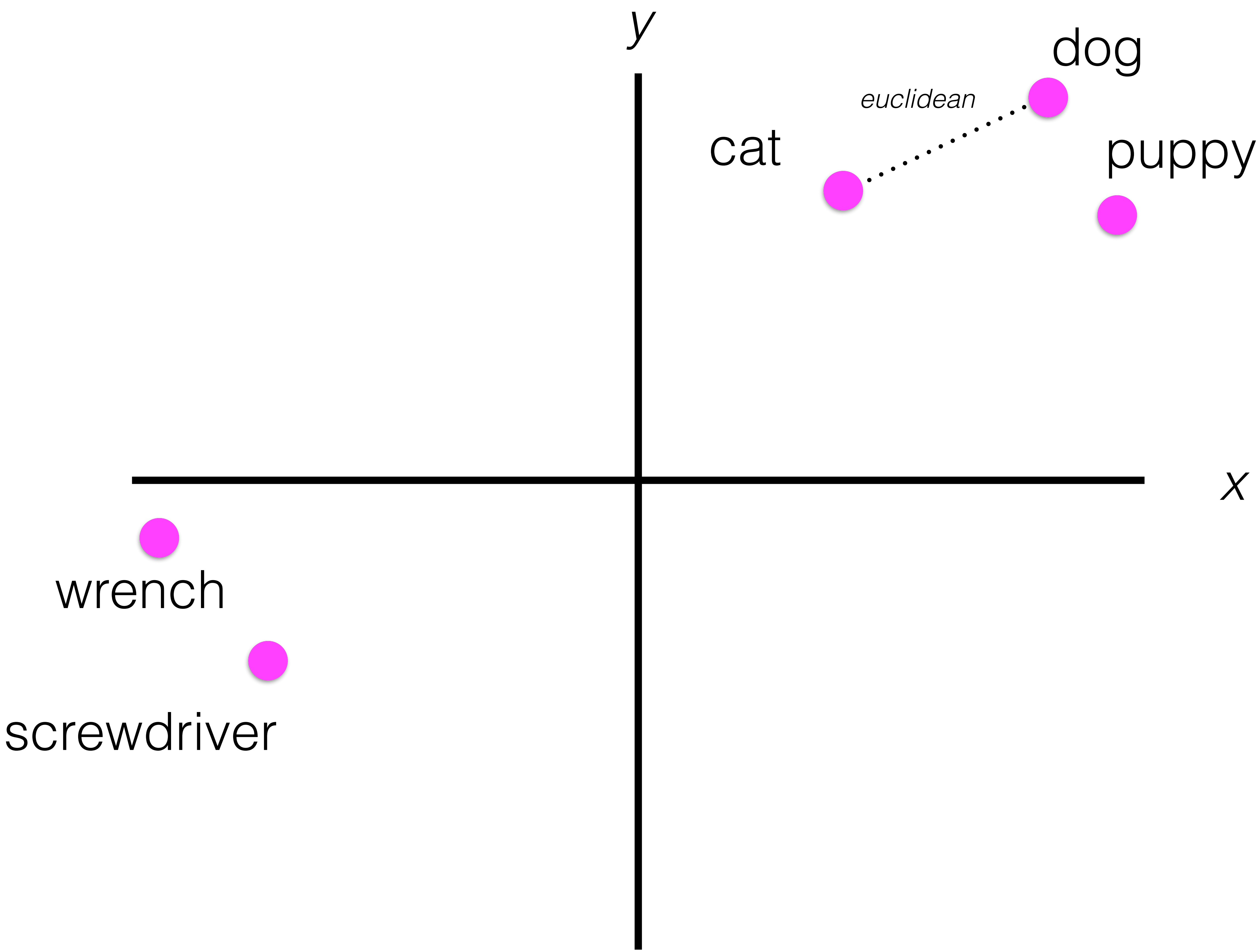
$$\sqrt{\sum_{i=1}^{F} (x_i - y_i)^2}$$

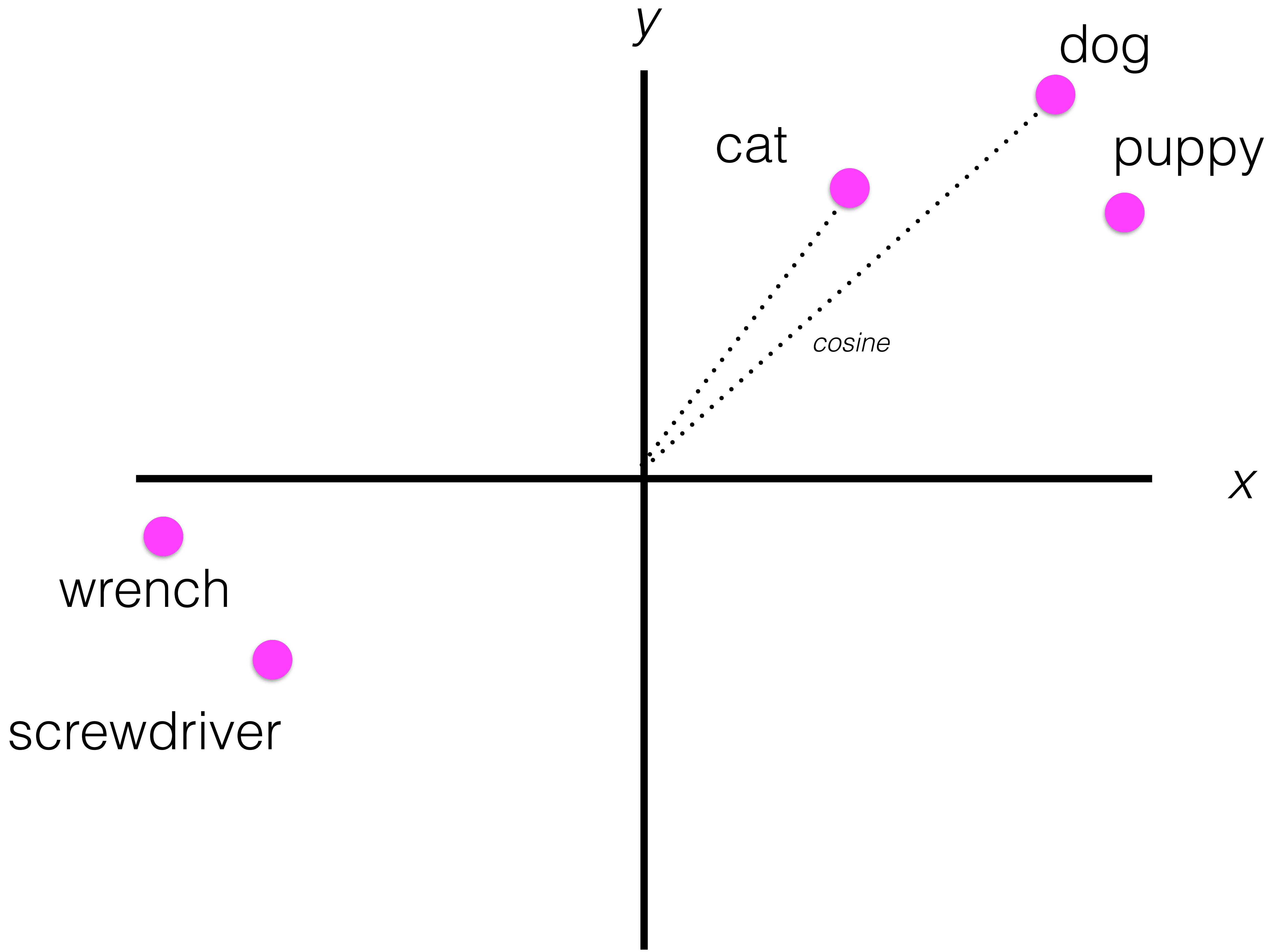
Cosine Similarity

x1	x2	x3
1	1	1
1	1	1
1	1	0
1	0	0
1	0	1
0	1	1
0	1	1
1	1	1

$$cos(x, y) = \frac{\sum_{i=1}^{F} x_i y_i}{\sqrt{\sum_{i=1}^{F} x_i^2}\sqrt{\sum_{i=1}^{F} y_i^2}}$$

- Euclidean distance measures the magnitude of distance between two points
- Cosine similarity measures their orientation


y
dog
euclidean
cat
puppy
x
wrench
screwdriver


y
dog
cat
puppy
cosine
x
wrench
screwdriver

Analogical inference

- Mikolov et al. 2013 show that vector representations have some potential for analogical reasoning through vector arithmetic.

apple - apples ≈ car - cars

king - man + woman ≈ queen

Mikolov et al., (2013), “Linguistic Regularities in Continuous Space Word Representations” (NAACL)

Bolukbasi et al. (2016)

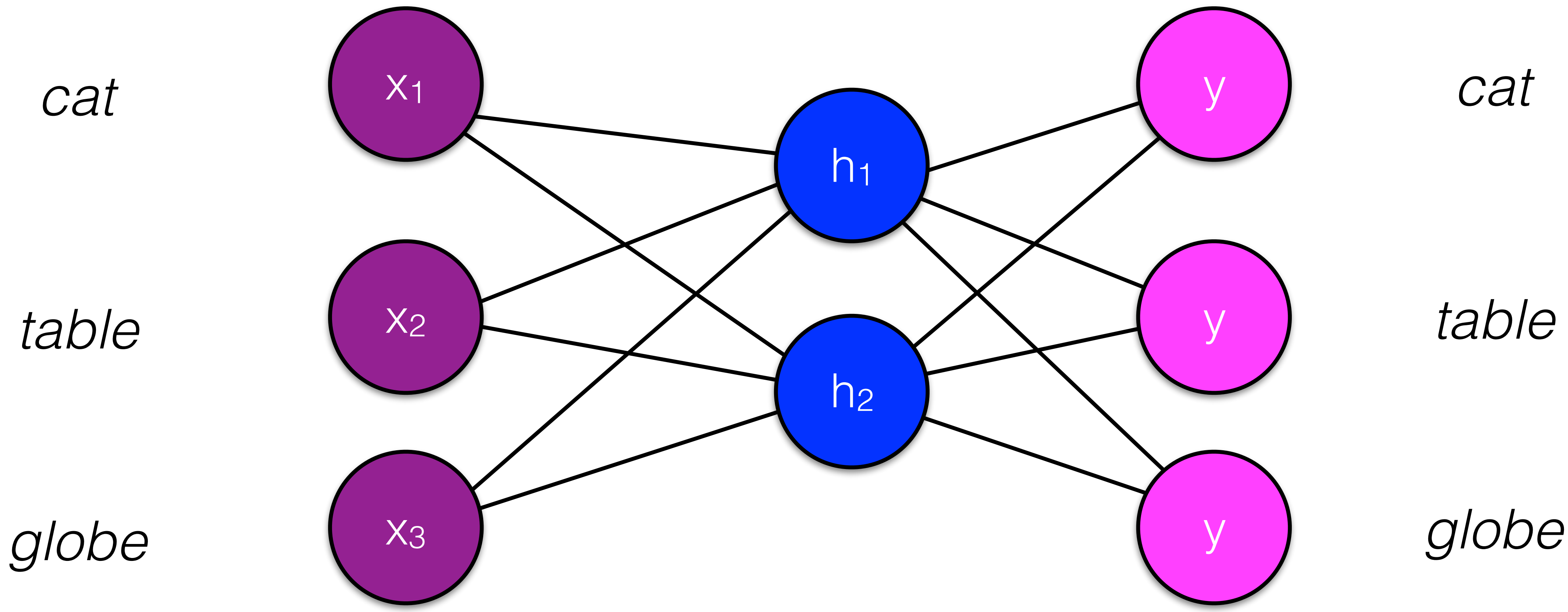
- Word vectors are trained on real data (web page, news, etc.) and reflect the inherent biases in how language is used

Code

http://mybinder.org/repo/dbamman/dds

code/vector_similarity

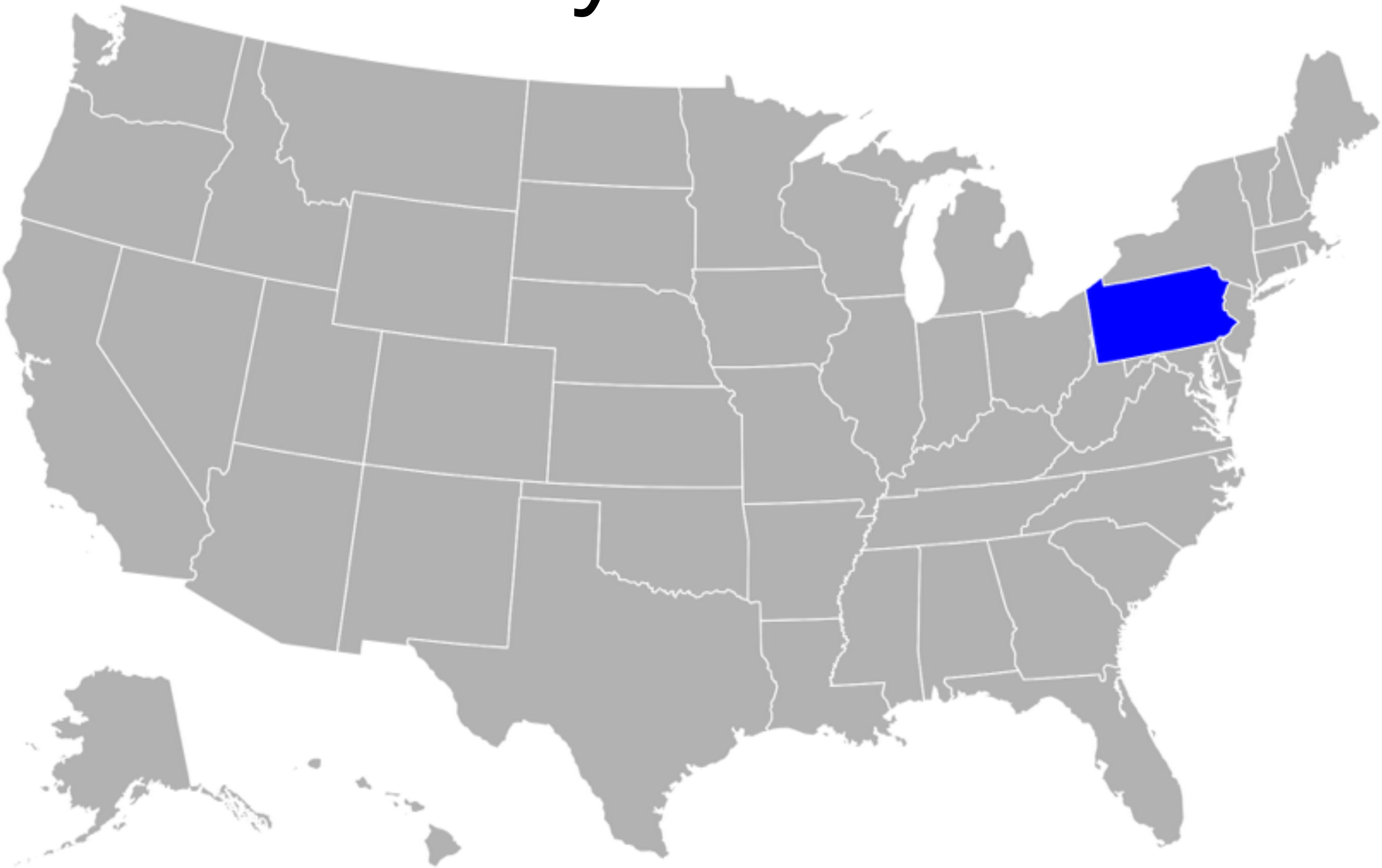
Assumptions



Lexical Variation

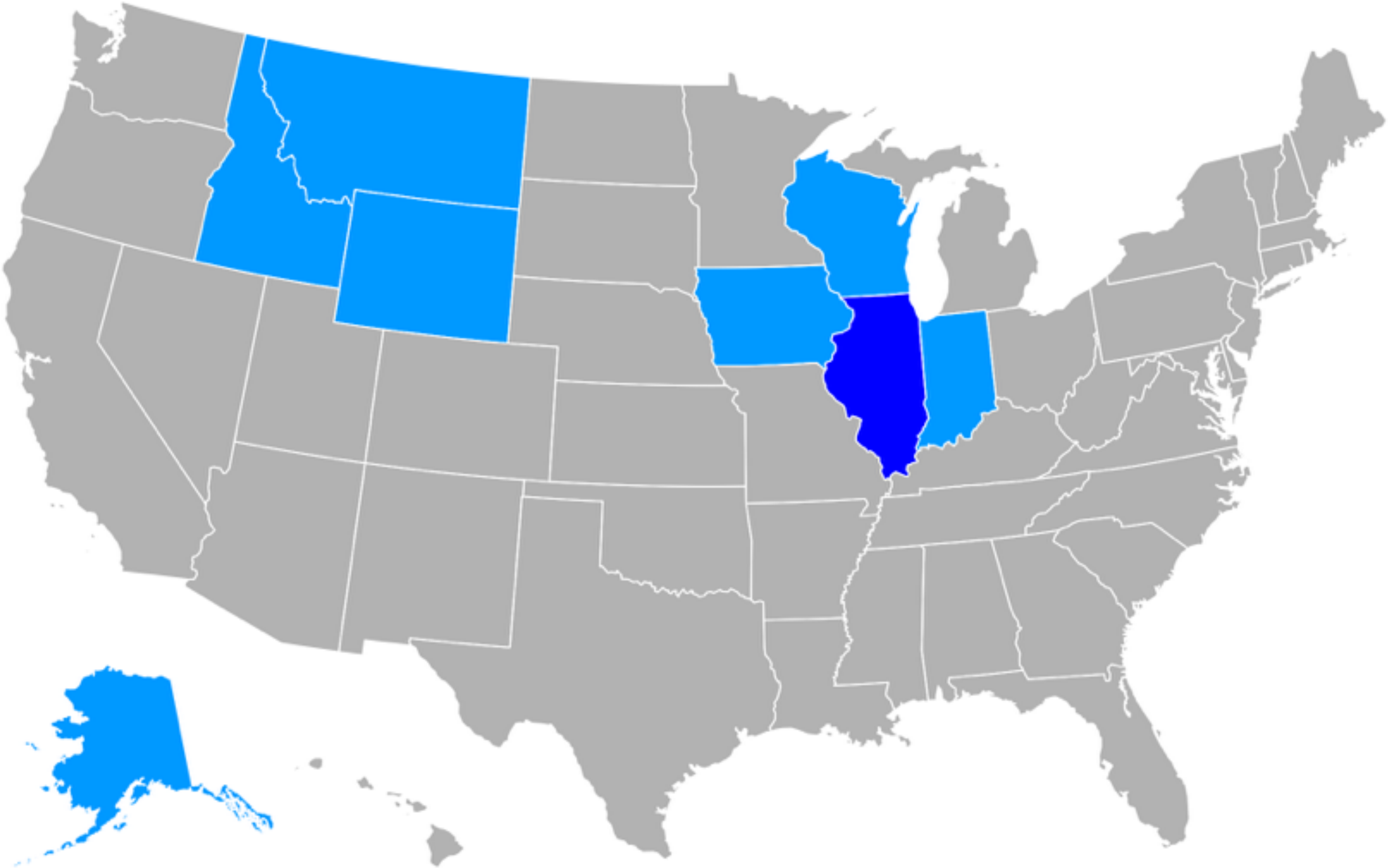
- People use different words in different regions.
- Lexical variation in social media
  Eisenstein et al. 2010; O'Connor et al. 2010; Eisenstein et al. 2011; Hong et al. 2012; Doyle 2014
- Text-based geolocation
  Wing and Baldridge 2011; Roller et al. 2012; Ikawa et al. 2012

“yinz”



Normalized document frequencies from 93M geotagged tweets (1.1B words).

“bears”



Normalized document frequencies from 93M geotagged tweets (1.1B words).

“bears” (IL)

- who's all watching the **bears** game ? :)
- watching **bears** game . no need to watch my lions . saw the score at the bottom of the screen . stafford and johnson are taking care of things .
- @USERNAME packers fans would be screaming that at **bears** fans if it had happened to chicago , all while laughing . schadenfreude .

“bears” (AK)

- troopers tracking brown **bears** on k beach 6/22/13 troopers ask that local_residents do not call law_enforcement ... @URL
- sci-tech : webcams make alaska **bears** accessible .
- angel rocks trail open ; dead calf moose might have attracted **bears** : fairbanks — state parks rangers on thursday ... @URL

Problem

How can we learn lexical representations that are sensitive to geographical variation not simply in word *frequency*, but in **meaning**?

- Vector-space models of lexical semantics
  Lin 1998; Turney and Pantel 2010, Reisinger and Mooney 2010, Socher et al. 2013, Mikolov et al. 2013, inter alia

- Low-dimensional "embeddings" ($w \in \mathbb{R}^K$)
  Bengio et al. 2006, Collobert and Weston 2008, Mnih and Hinton 2008, Turian et al. 2010, Socher et al. 2011ff., Collobert et al. 2011, Mikolov et al. 2013; Levy and Goldberg 2014.

“bears”

- who's all watching the **bears** game ? :)
- watching **bears** game . no need to watch my lions . saw the score at the bottom of the screen . stafford and johnson are taking care of things .
- troopers tracking brown **bears** on k beach troopers ask that local_residents do not call law_enforcement ... @URL
- sci-tech : webcams make alaska **bears** accessible .

“bears”

- who's all watching the **bears** game ? :) IL
- watching **bears** game . no need to watch my lions . saw the score at the bottom of the screen . stafford and johnson are taking care of things . IL
- troopers tracking brown **bears** on k beach 6/22/13 troopers ask that local_residents do not call law_enforcement ... @URL AK
- sci-tech : webcams make alaska **bears** accessible . AK

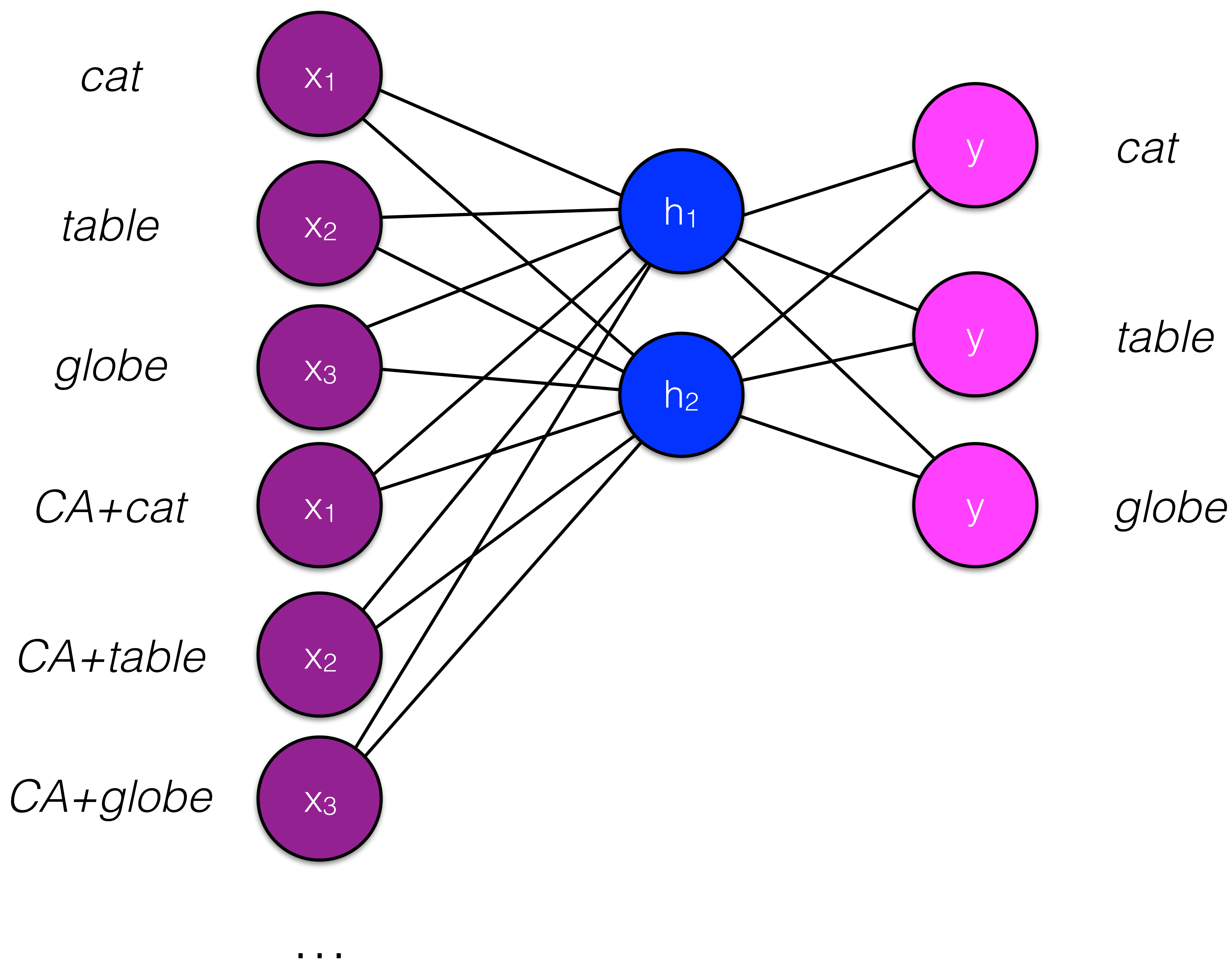
Context + Metadata

my boy's wicked smart

MA

y = smart

x = {wicked, wicked+MA}


cat
table
globe
CA+cat
CA+table
CA+globe
x1
x2
x3
x1
x2
x3
h1
h2
y
y
y
cat
table
globe
...

Model

K

Word $|V|$

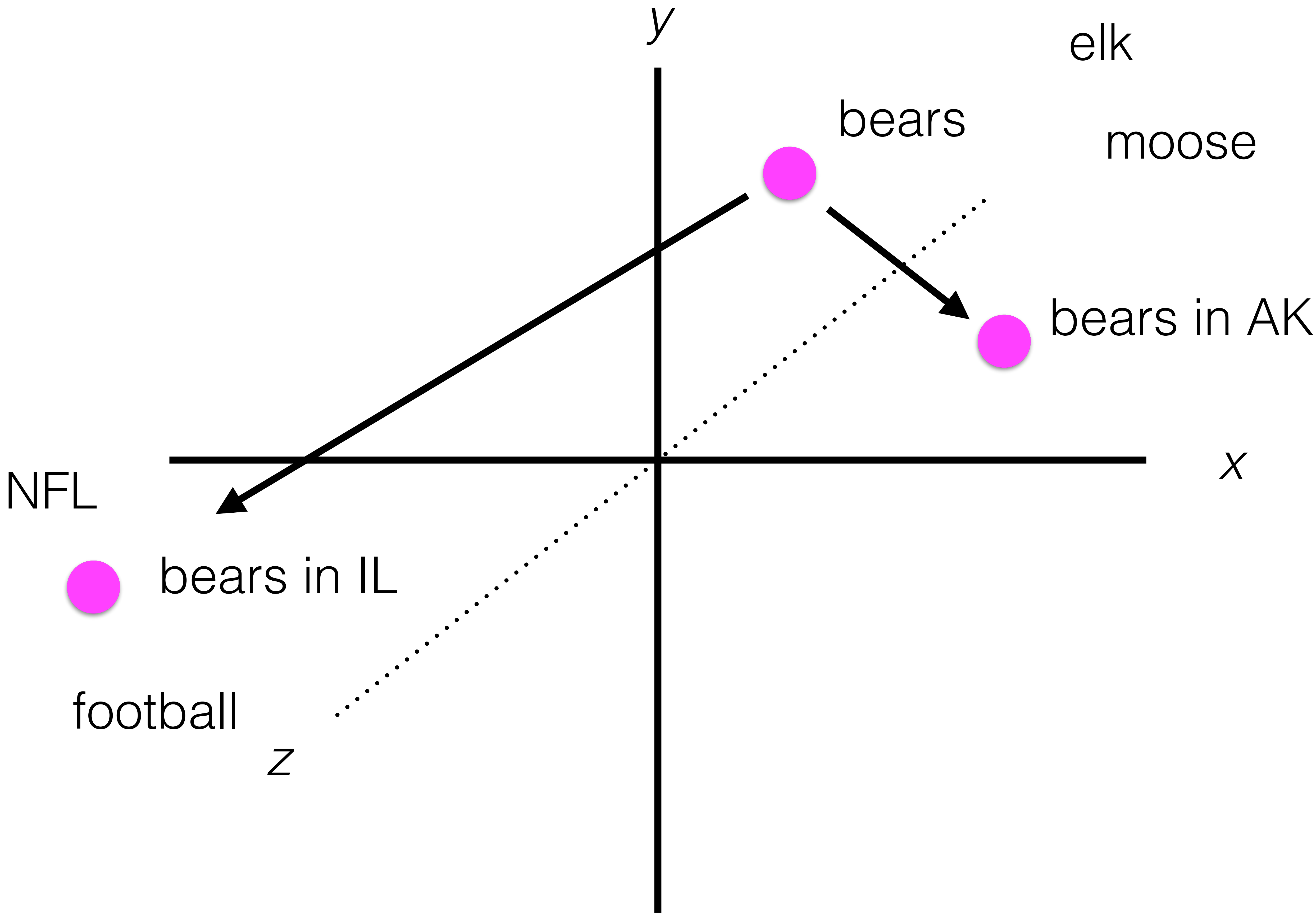
bears	3.76	10.4	-1.3
red	0.3	4.10	13.3
the	0.1	3.3	-1.2
zoo	-10.3	-13.1	1.4

Word+Alabama $|V|$

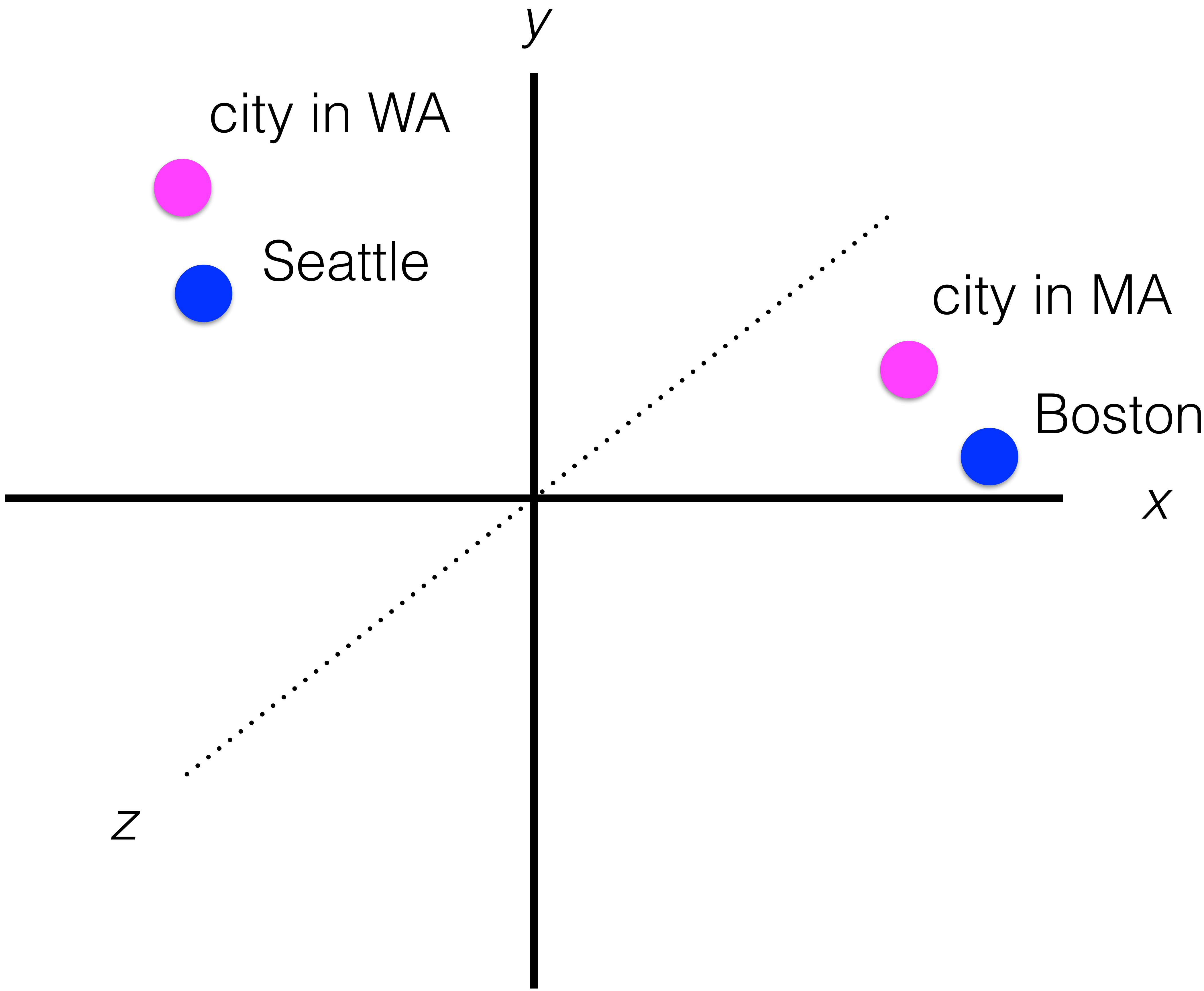
bears	-0.30	-3.1	1.04
red	4.5	0.3	-1.3
the	1.3	-1.2	0.1
zoo	5.2	7.2	1.5

Word+Alaska $|V|$

bears	5.6	8.3	-0.8
red	3.1	0.14	6.8
the	-0.1	-0.7	1.4
zoo	6.7	2.1	-3.7


y
elk
bears
moose
bears in AK
x
NFL
bears in IL
football
z

“let’s go into the city”


y
city in WA
Seattle
city in MA
Boston
x
z

city

- valley
- bay
- downtown
- chinatown
- south bay
- area
- east bay
- neighborhood
- peninsula

Terms with the highest cosine similarity to *city* in California

city

- suburbs
- town
- hamptons
- big city
- borough
- neighborhood
- downtown
- upstate
- big apple

Terms with the highest cosine similarity to *city* in New York.

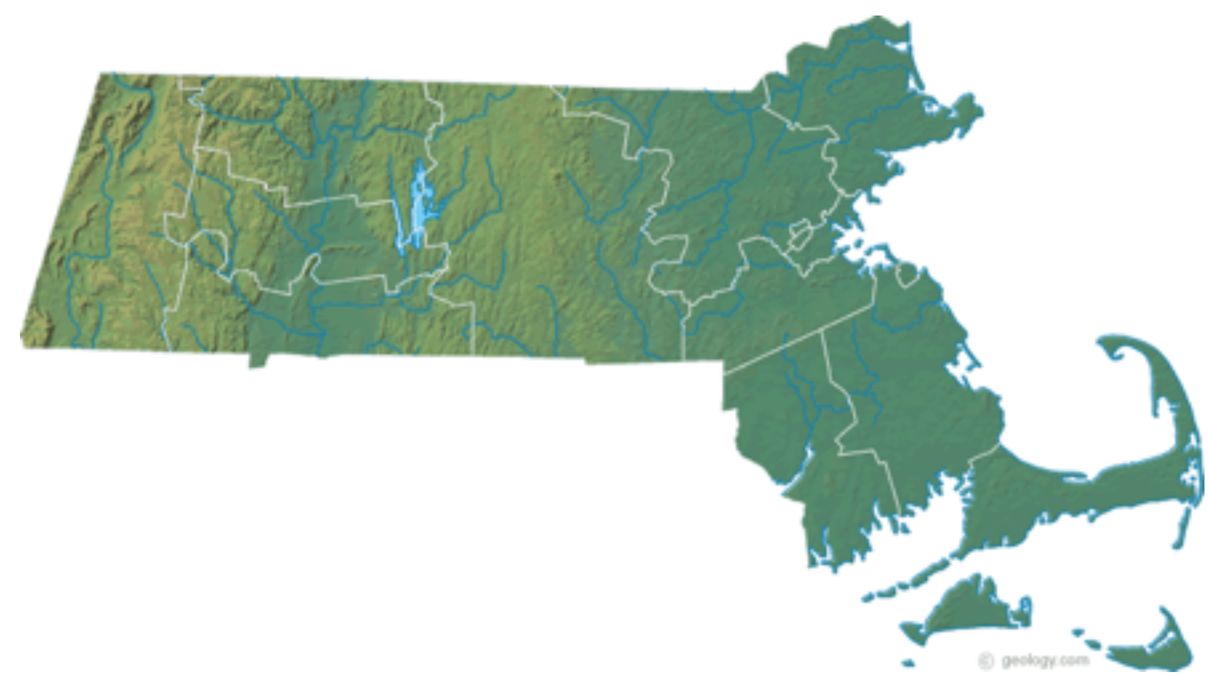
wicked

- evil
- pure
- gods
- mystery
- spirit
- king
- above
- righteous
- magic

Terms with the highest cosine similarity to *wicked* in Kansas

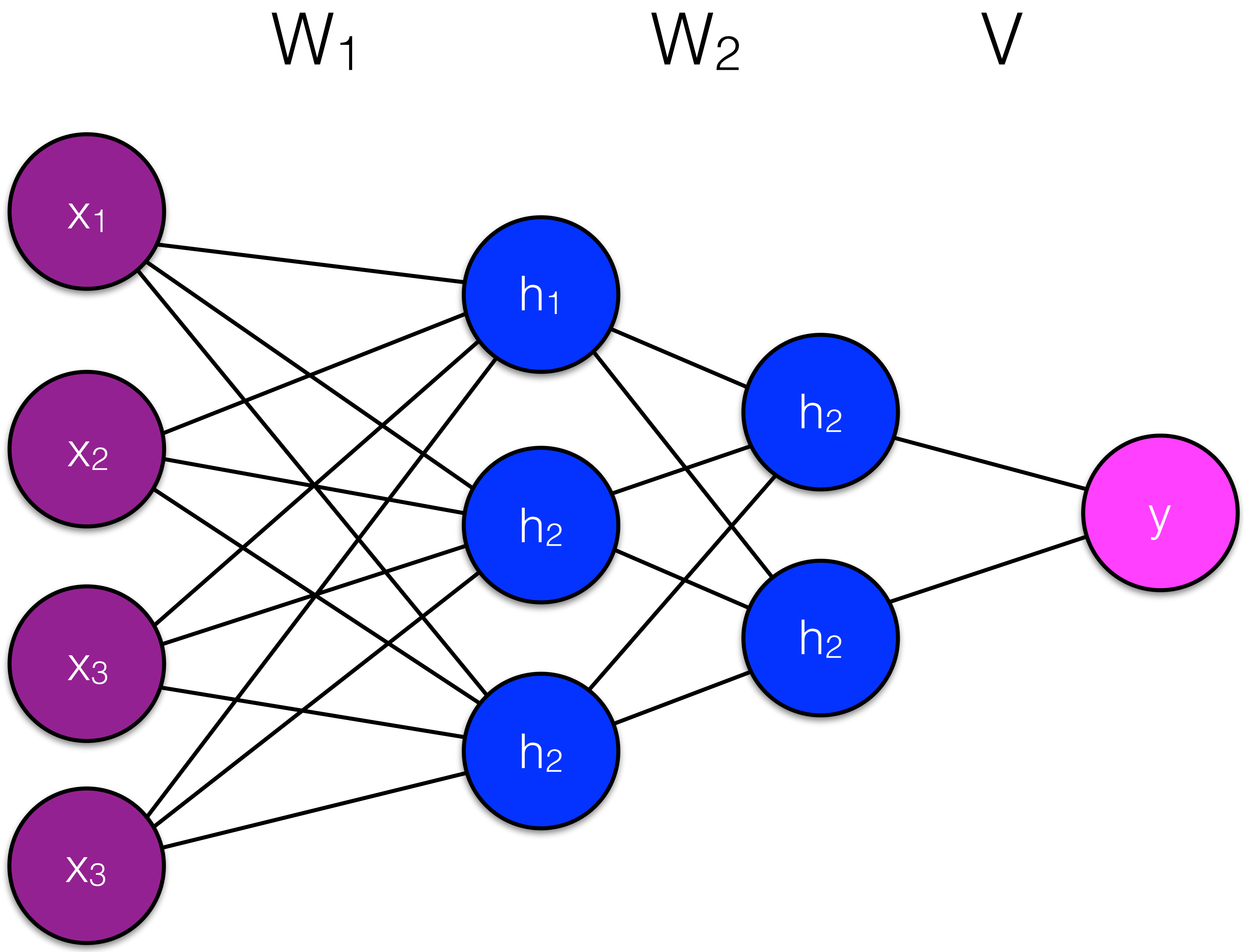
wicked

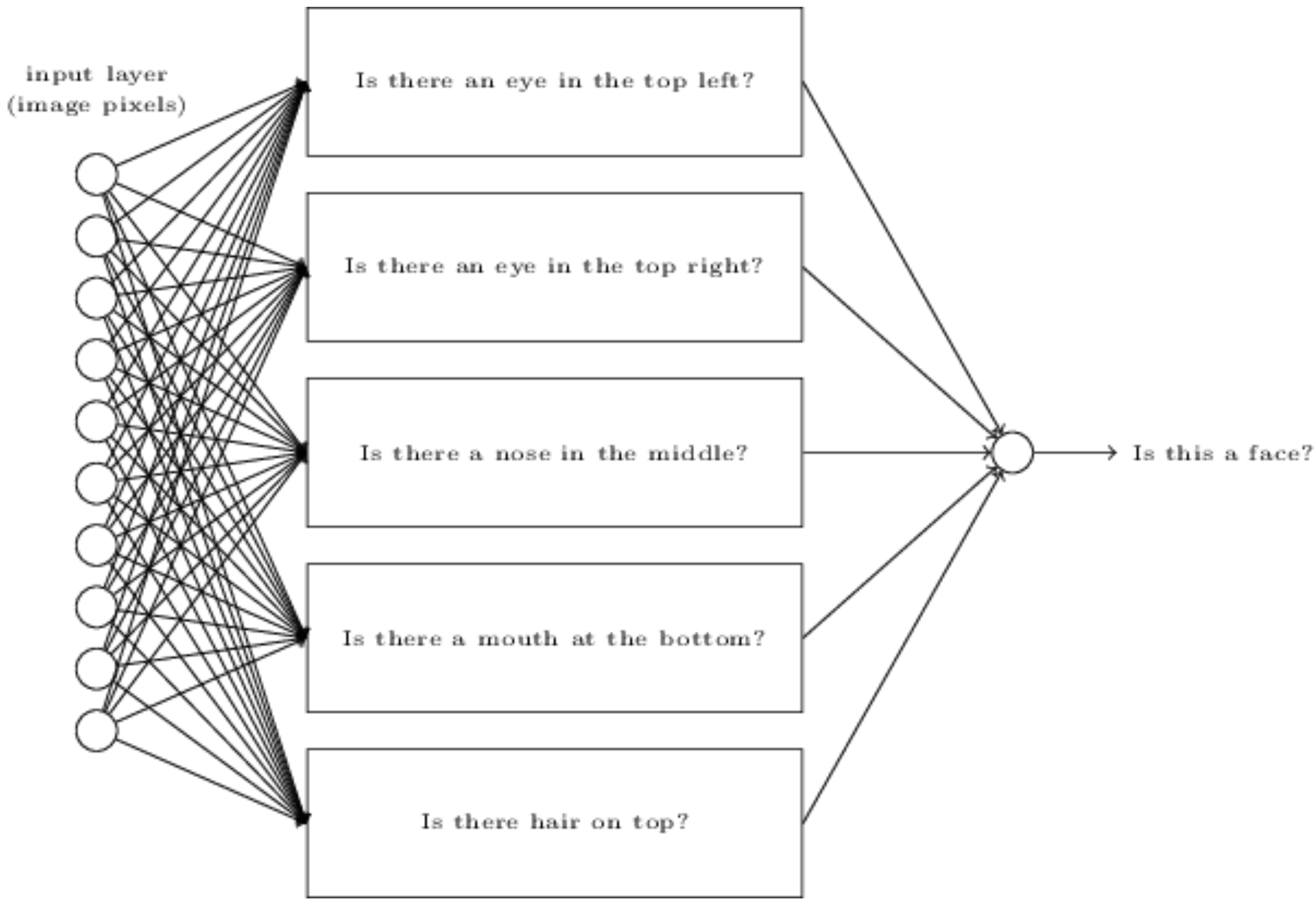
- super
- ridiculously
- insanely
- extremely
- goddamn
- surprisingly
- kinda
- #sarcasm
- sooooooo



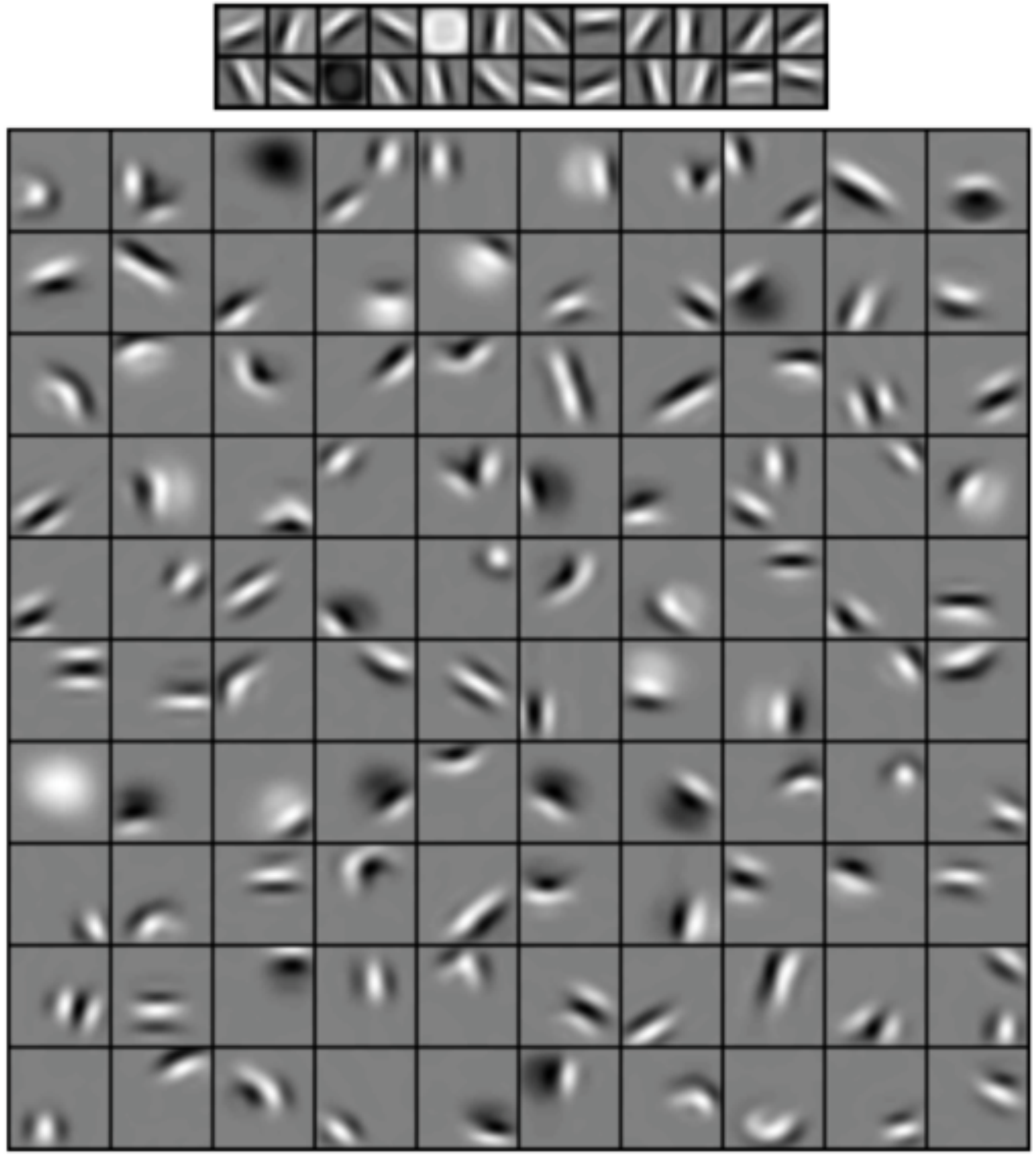
Terms with the highest cosine similarity to *wicked* in Massachusetts

Deeper networks





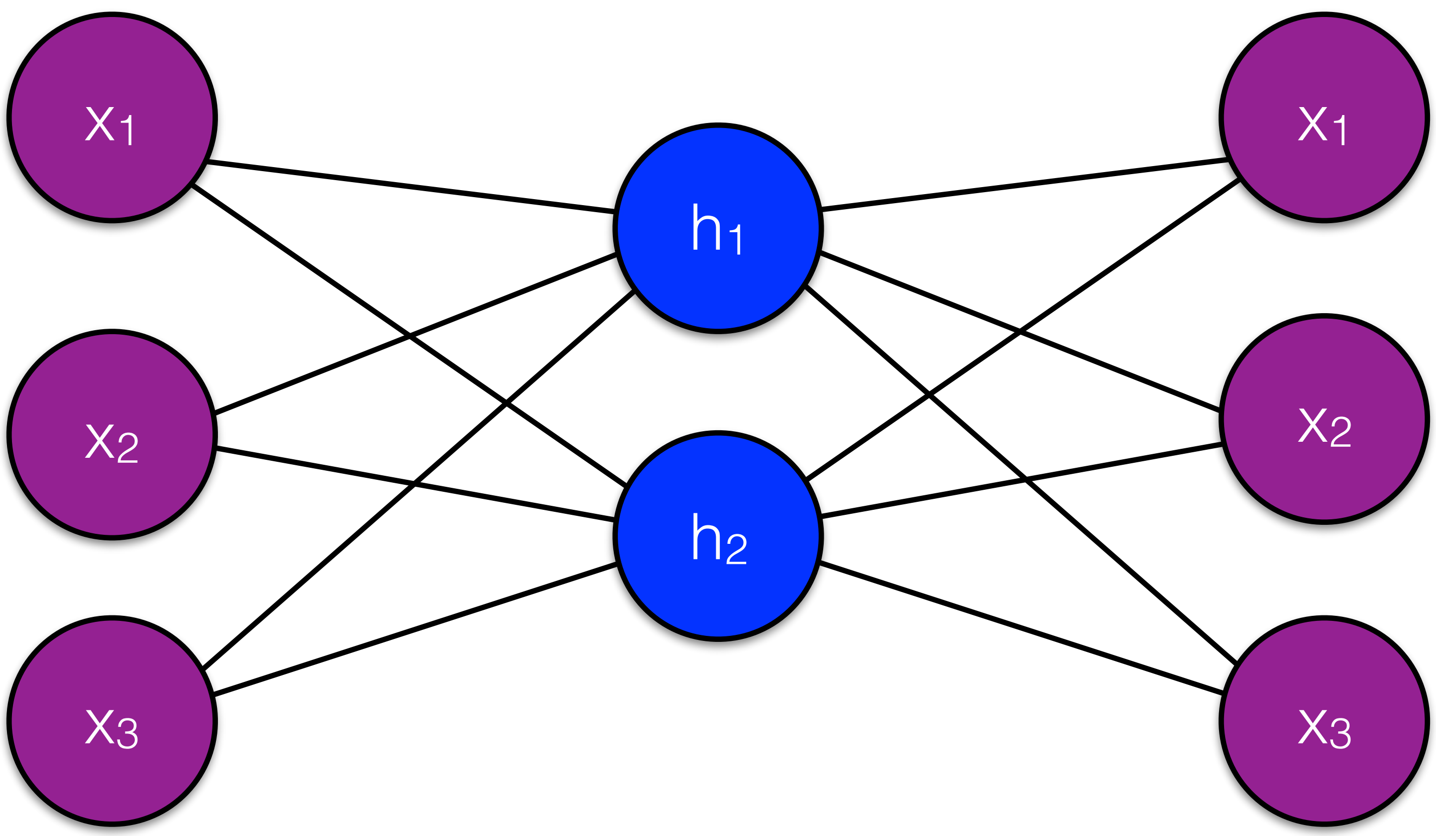
http://neuralnetworksanddeeplearning.com/chap1.html



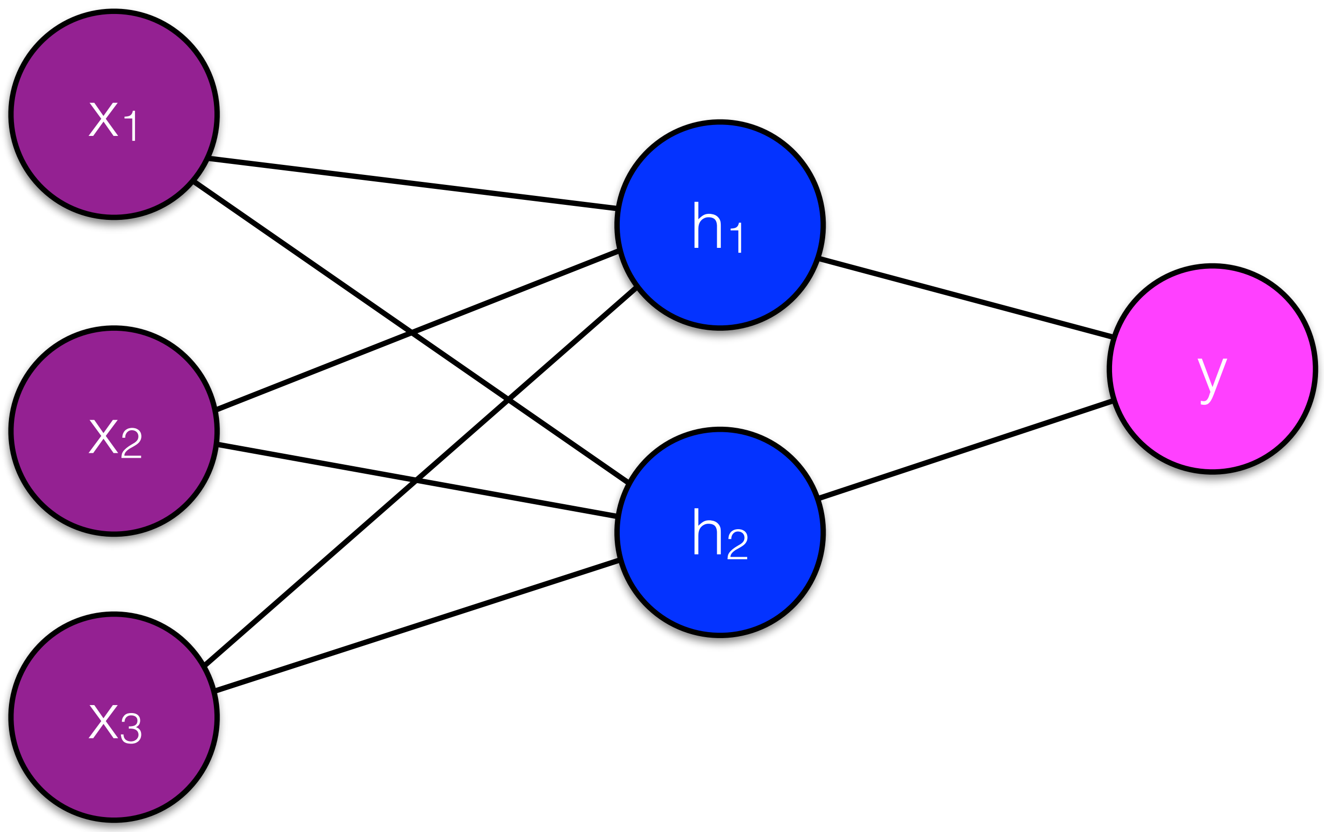
Higher order features learned for image recognition
Lee et al. 2009 (ICML)

Autoencoder

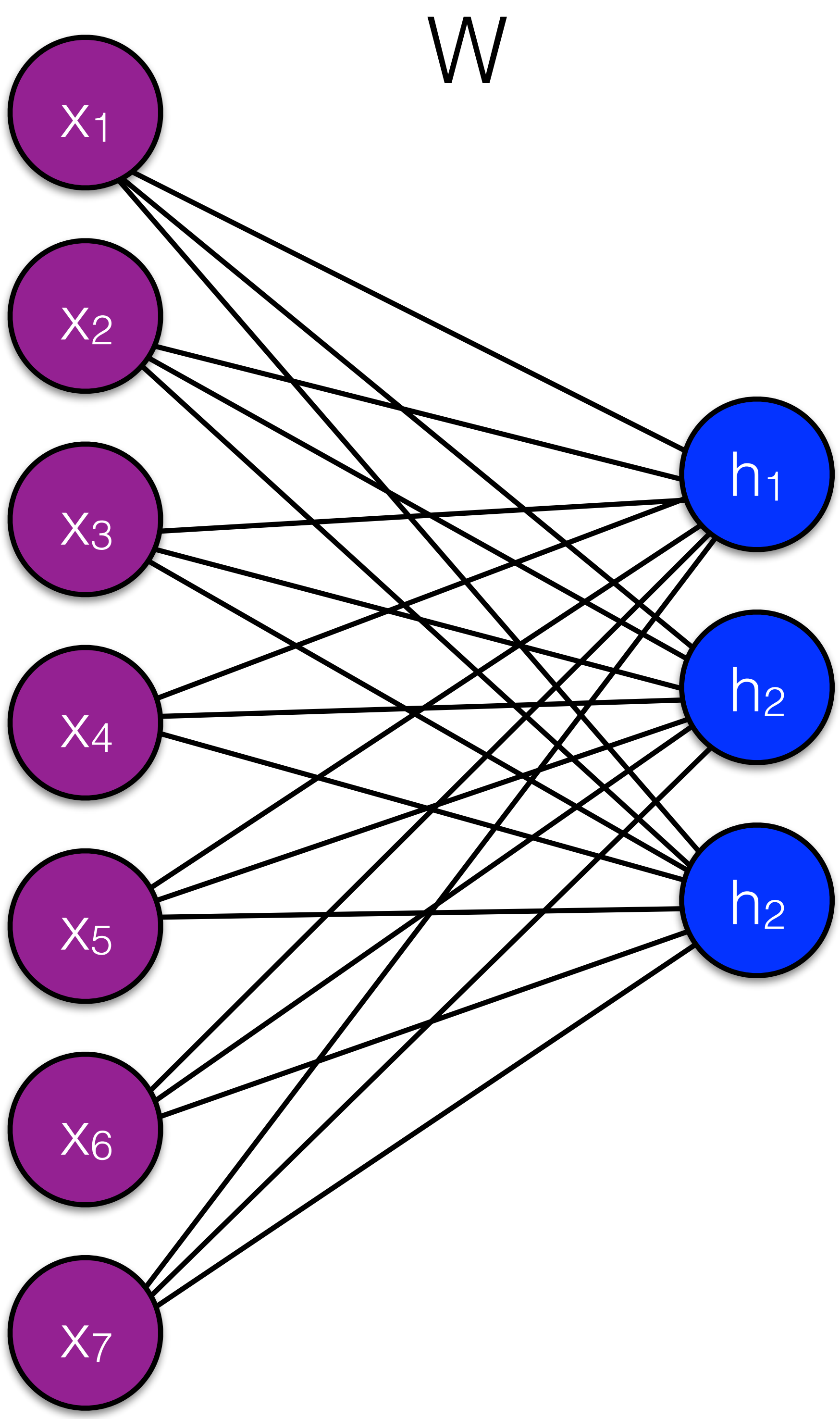
- Unsupervised neural network, where $y = x$
- Learns a low-dimensional representation of x

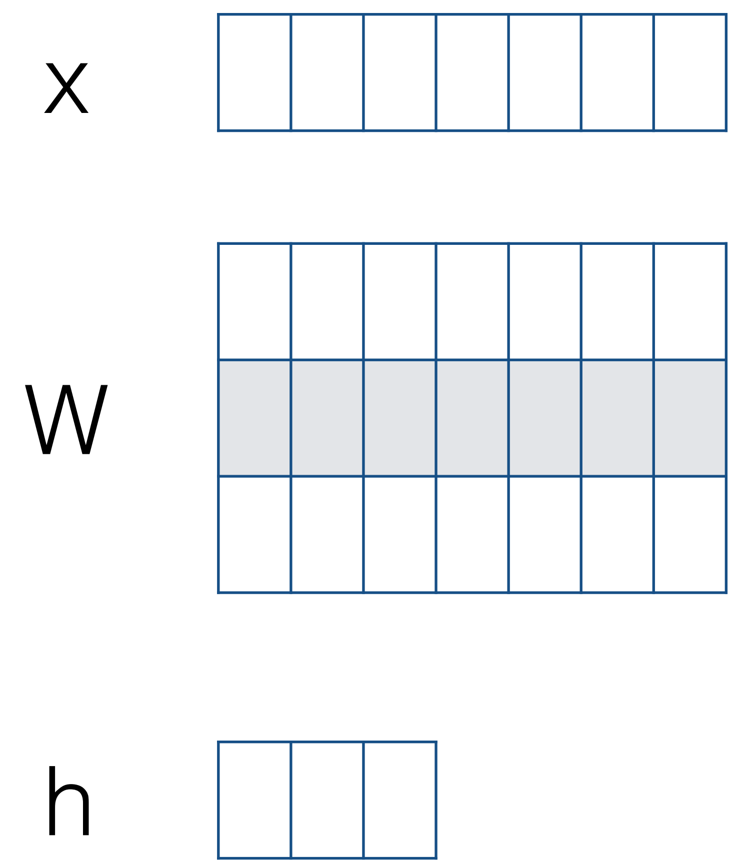


Feedforward networks



Densely connected layer



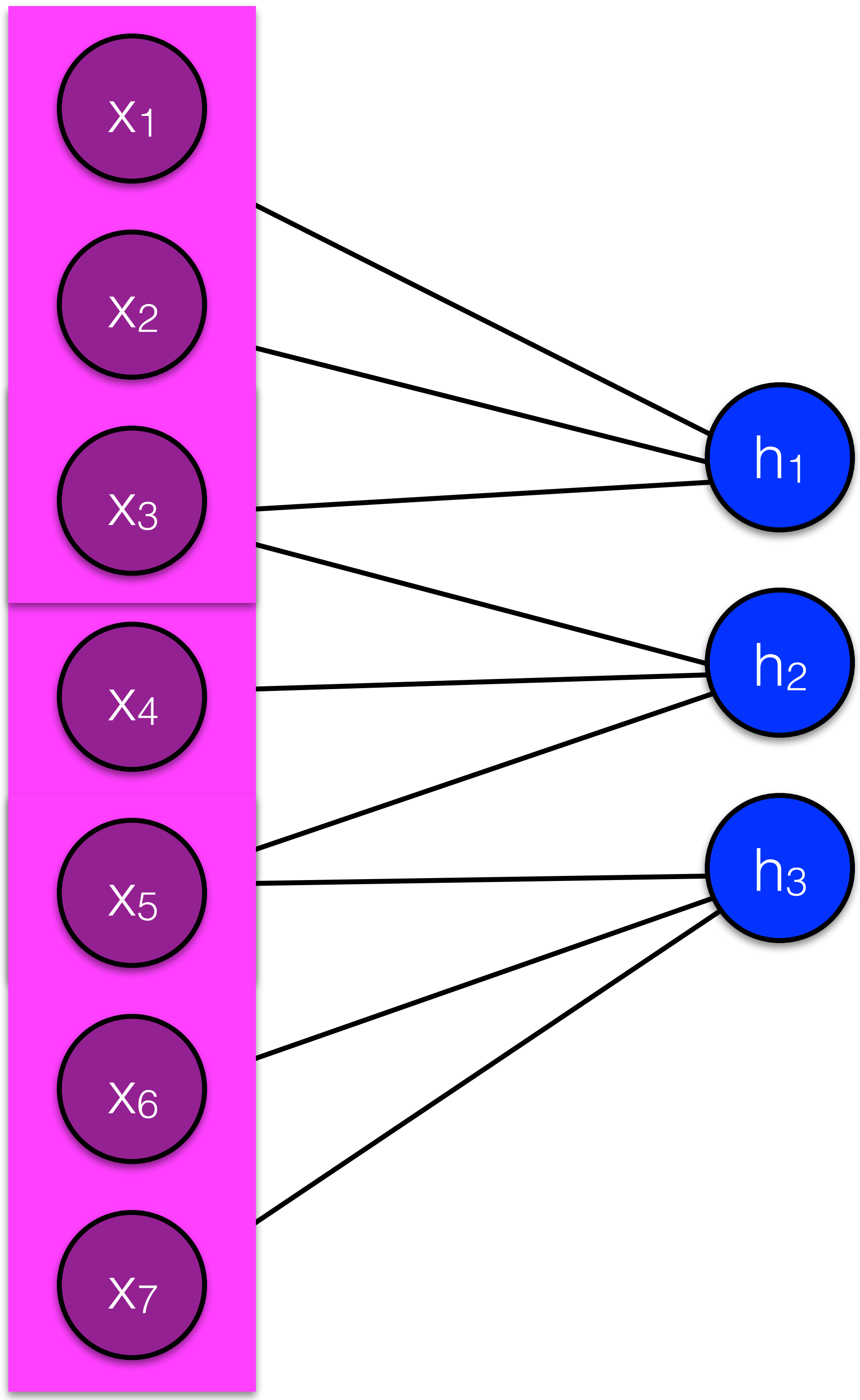


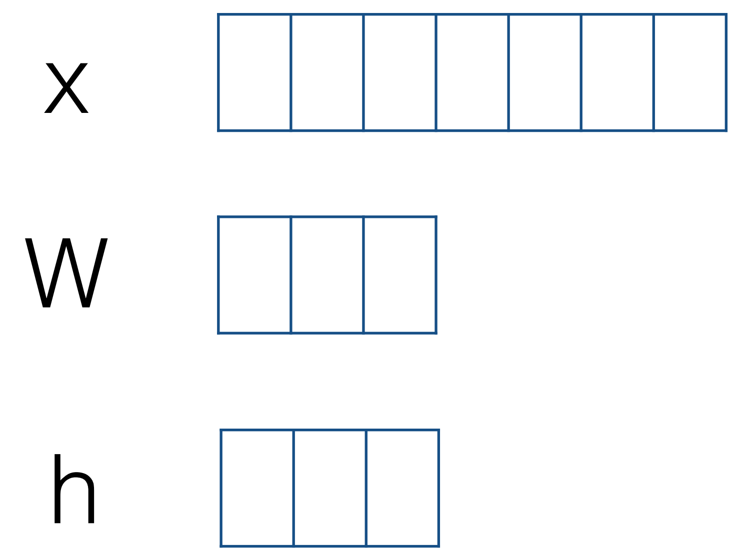
$$h = \sigma(xW)$$

Convolutional networks

- With convolution networks, the same operation is (i.e., the same set of parameters) is applied to different regions of the input

Convolutional networks





$$h_1 = \sigma(x_1W_1 + x_2W_2 + x_3W_3)$$

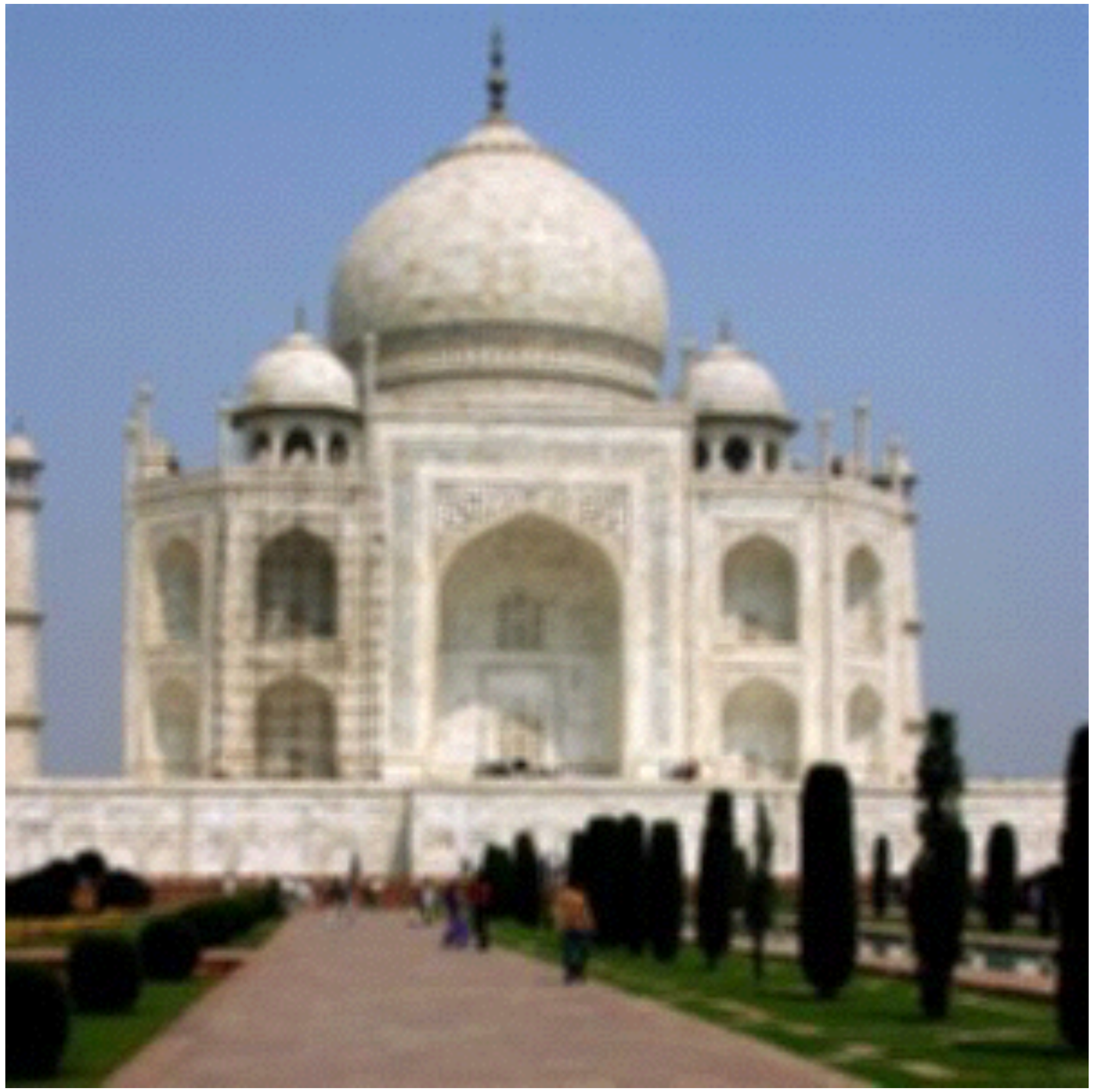
$$h_2 = \sigma(x_3W_1 + x_4W_2 + x_5W_3)$$

$$h_3 = \sigma(x_5W_1 + x_6W_2 + x_7W_3)$$

2D Convolution

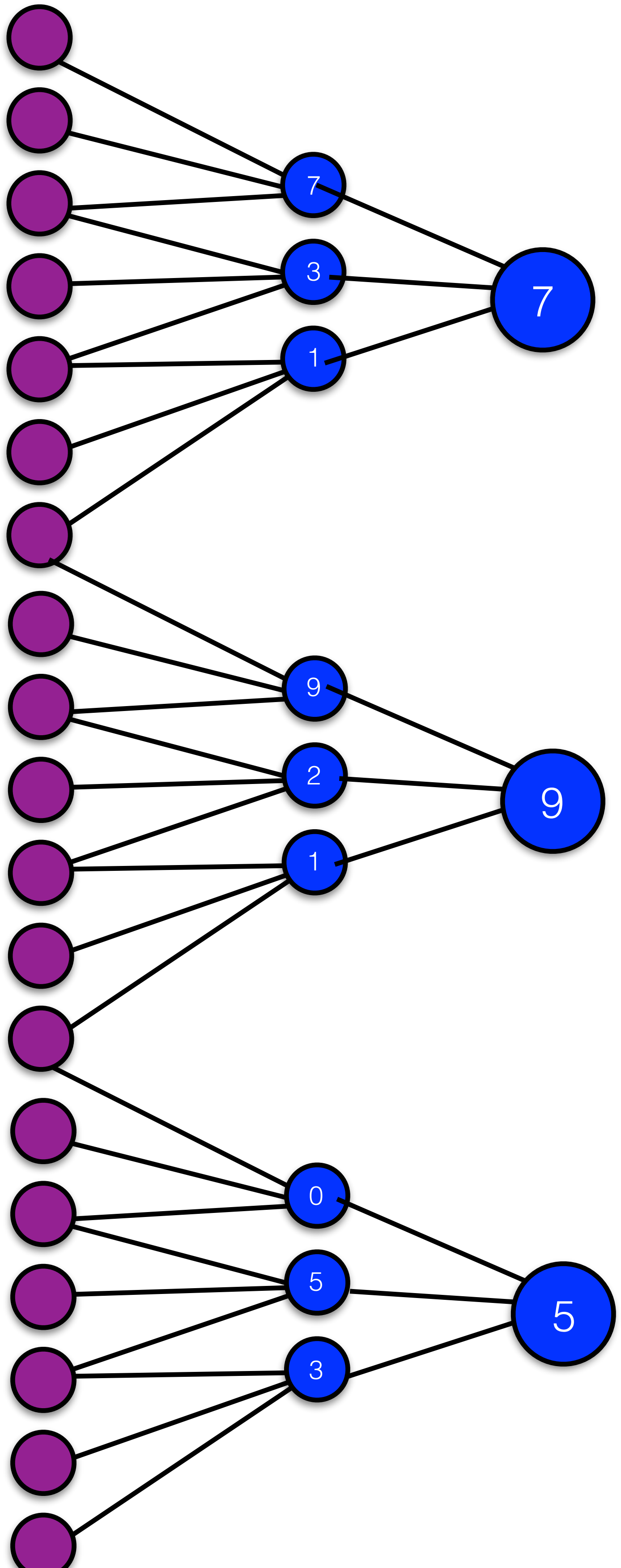
0	0	0	0	0
0	1	1	1	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

blurring



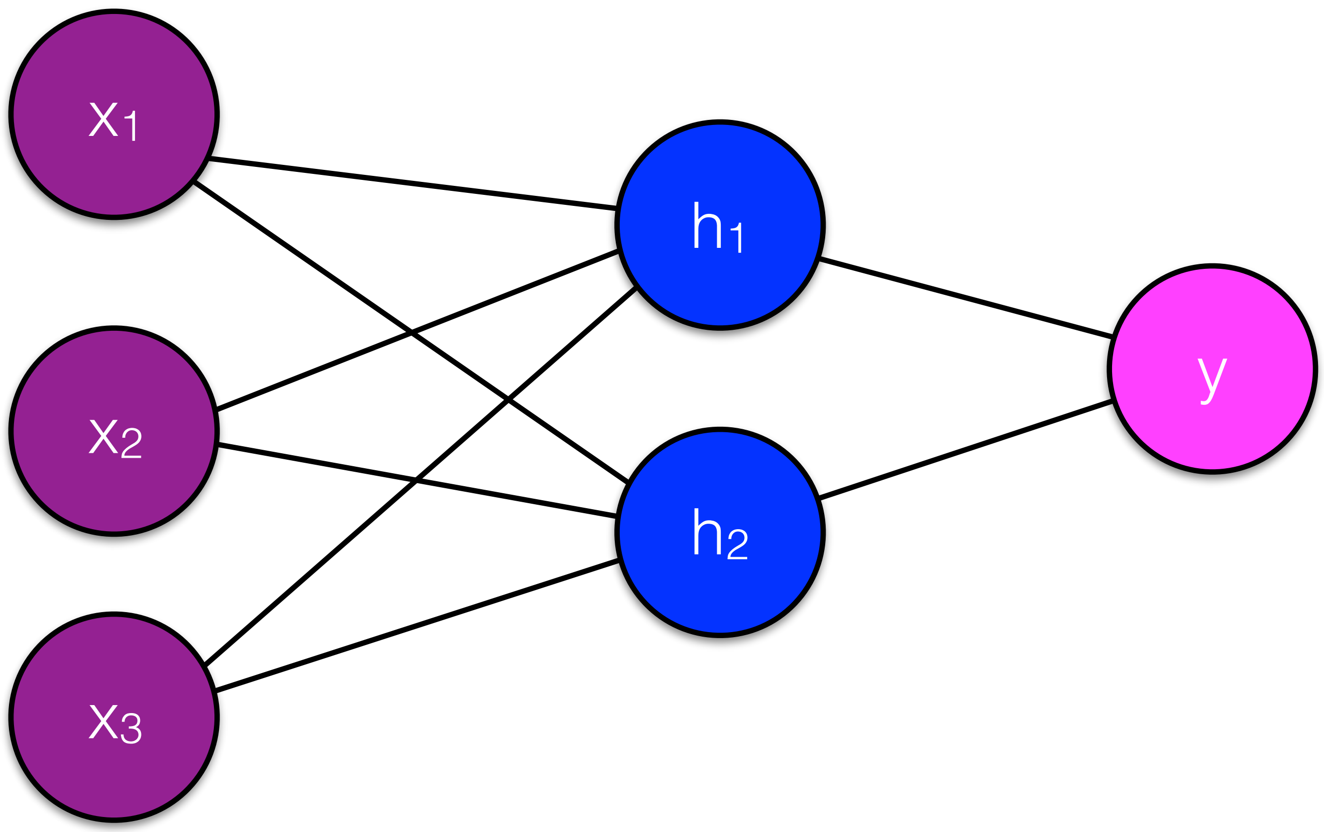
http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/
https://docs.gimp.org/en/plug-in-convmatrix.html

Pooling

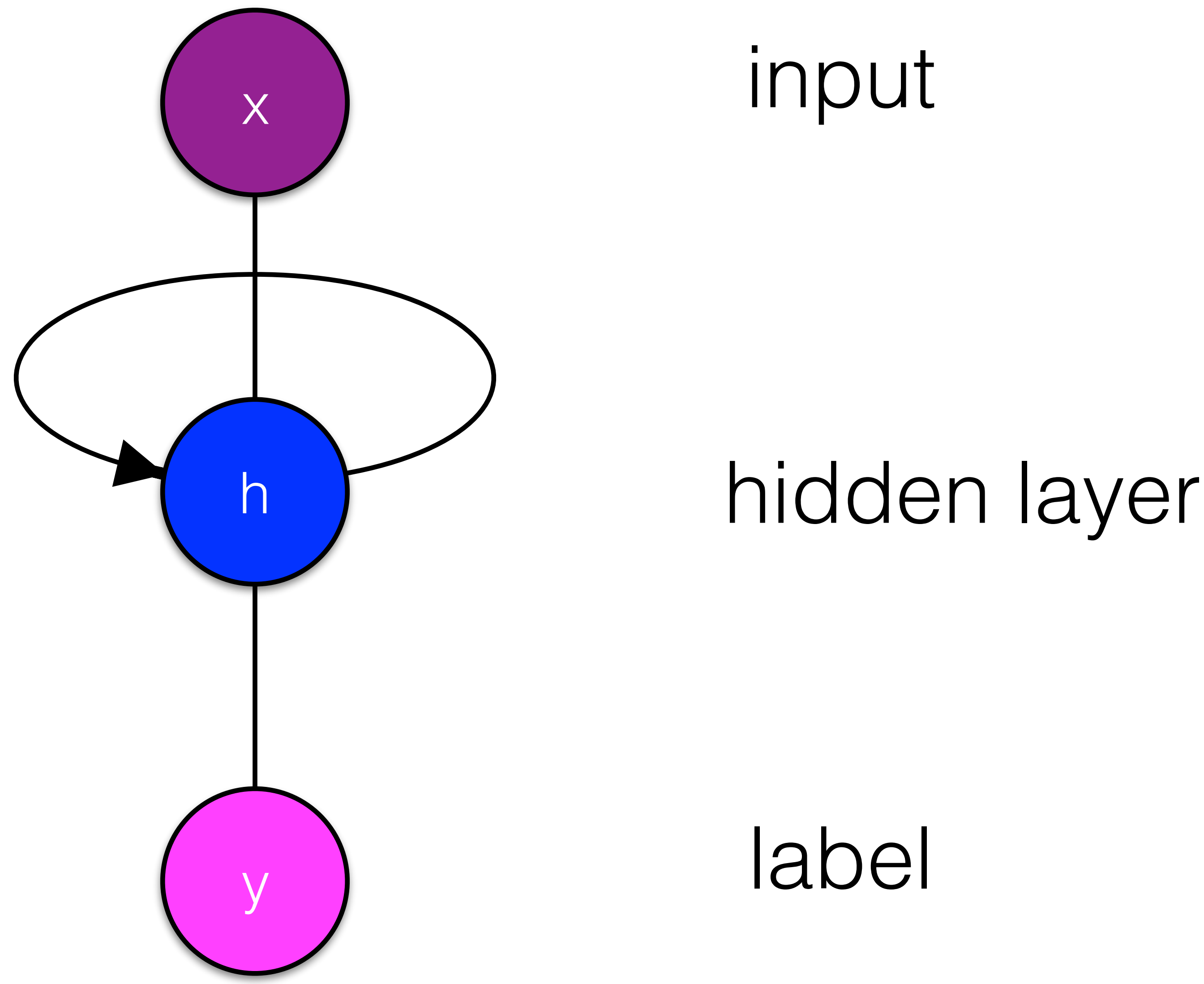


- Down-samples a layer by selecting a single point from some set
- Max-pooling selects the largest value

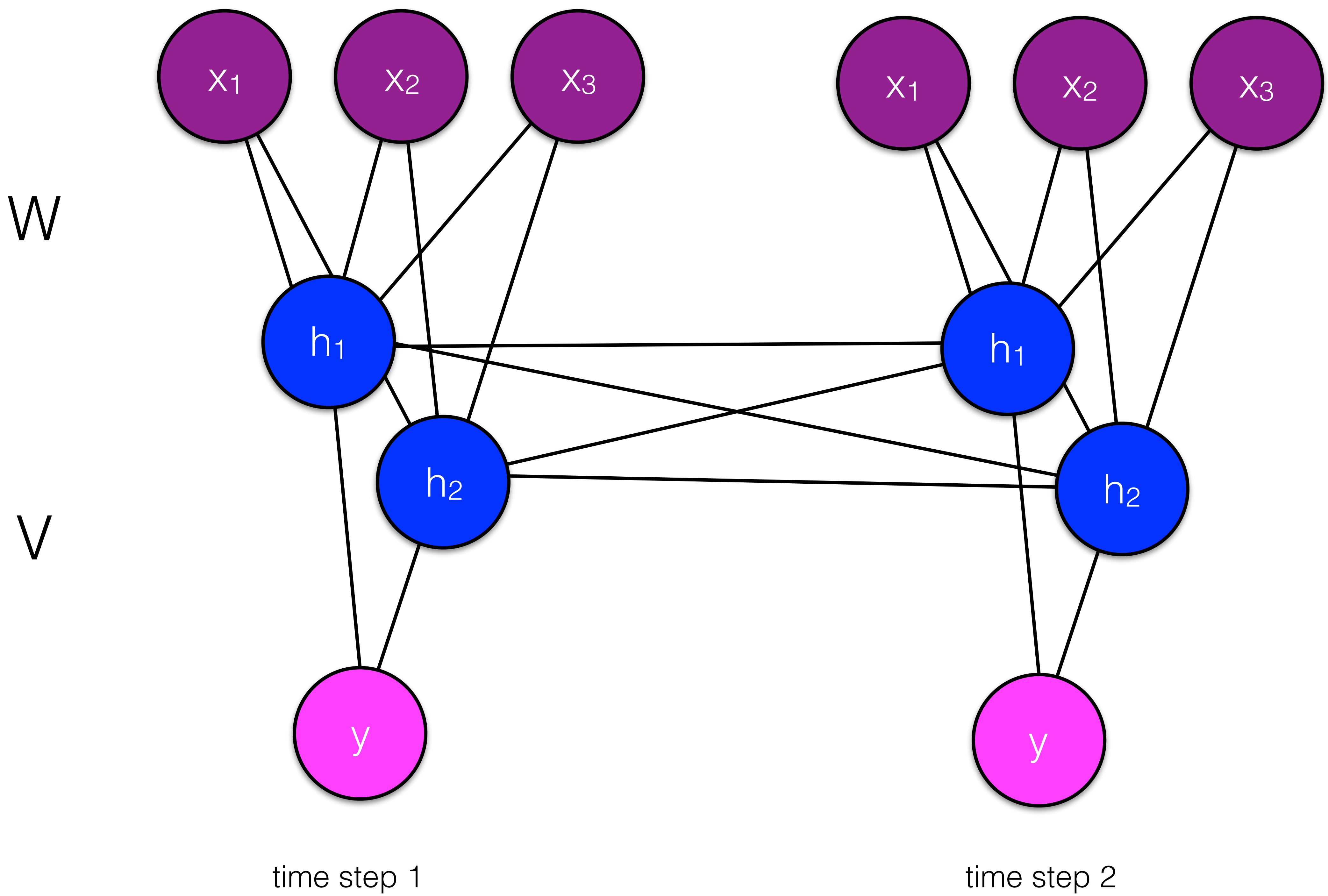
Feedforward networks



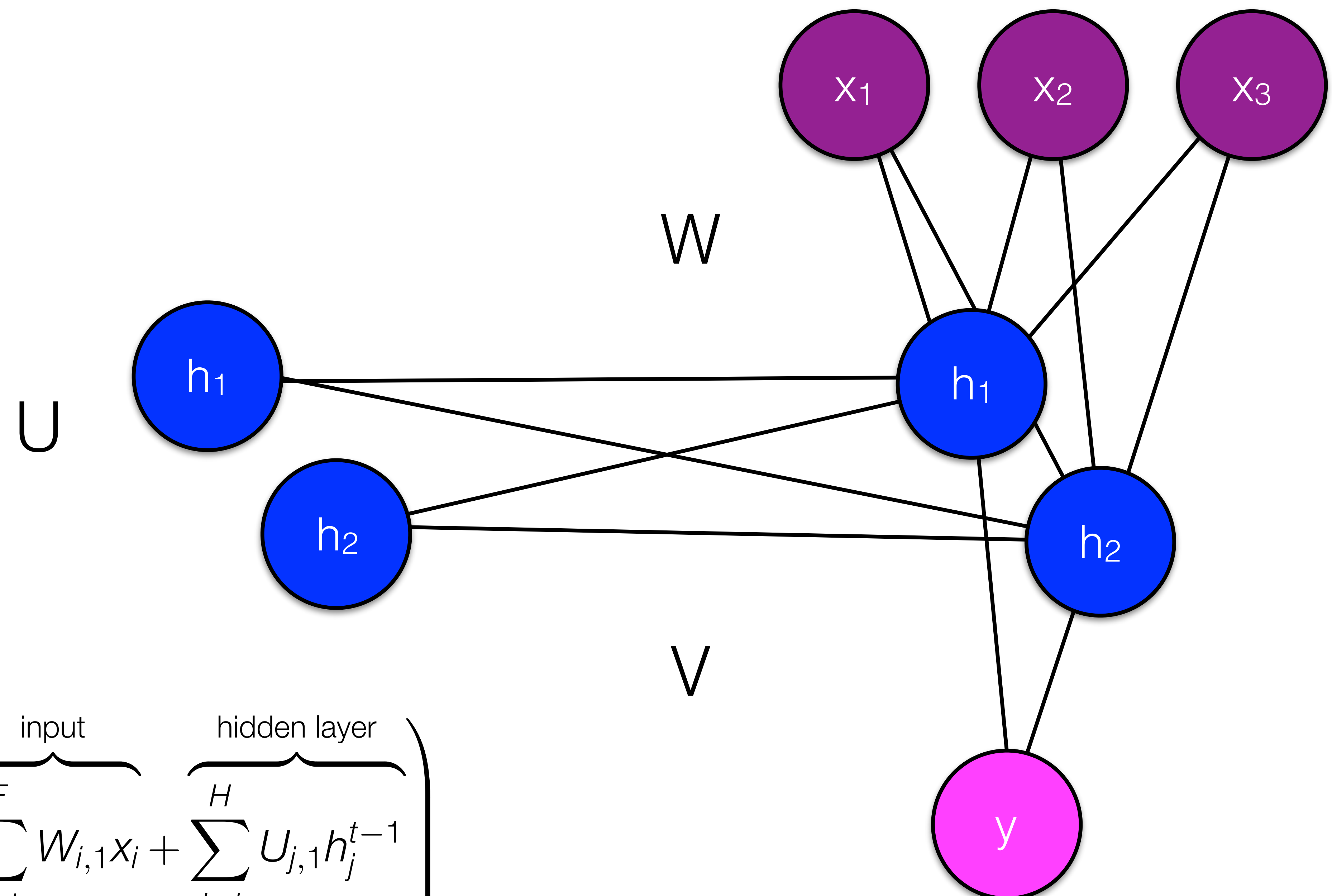
Recurrent networks



Recurrent networks



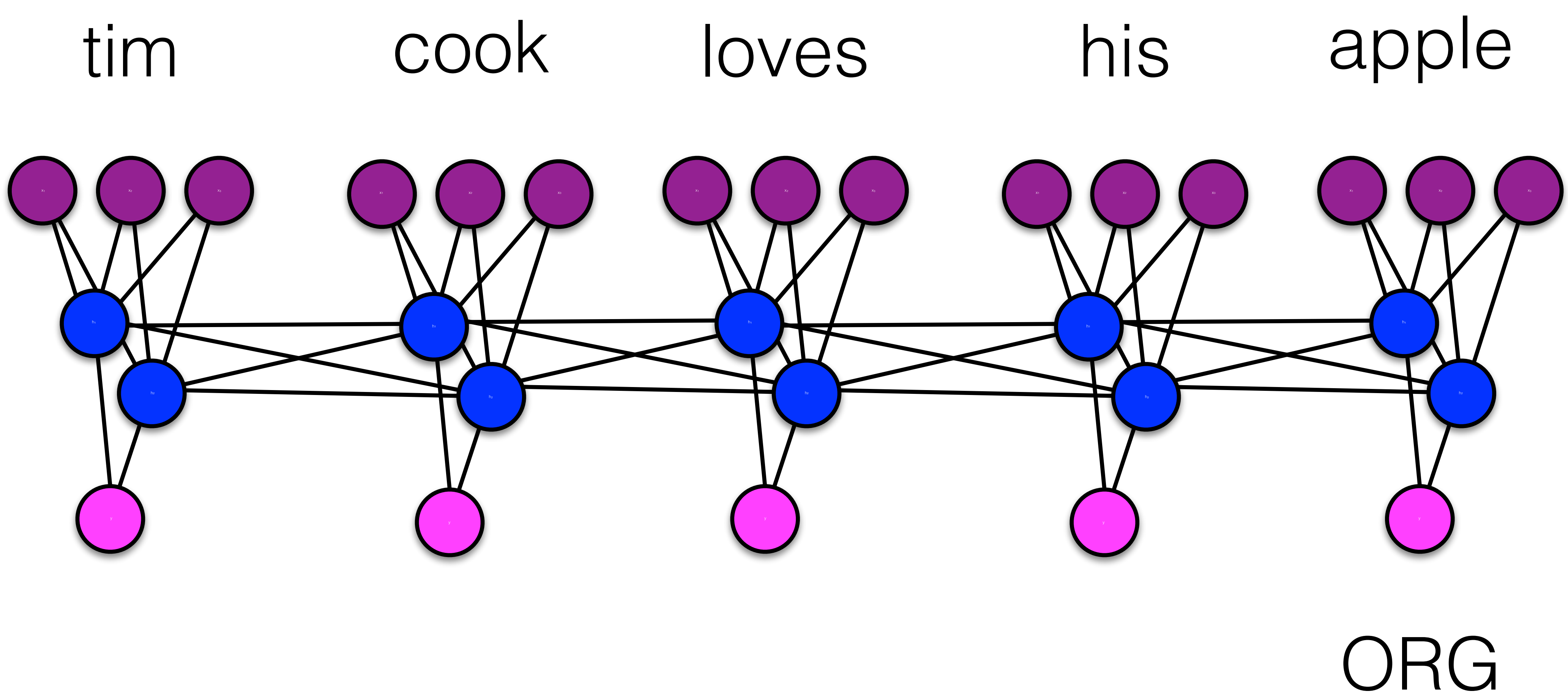
Recurrent networks



$$h_1 = \sigma\left(\overbrace{\sum_{i=1}^{F} W_{i,1}x_i}^{\text{input}} + \overbrace{\sum_{j=i}^{H} U_{j,1}h_j^{t-1}}^{\text{hidden layer}}\right)$$

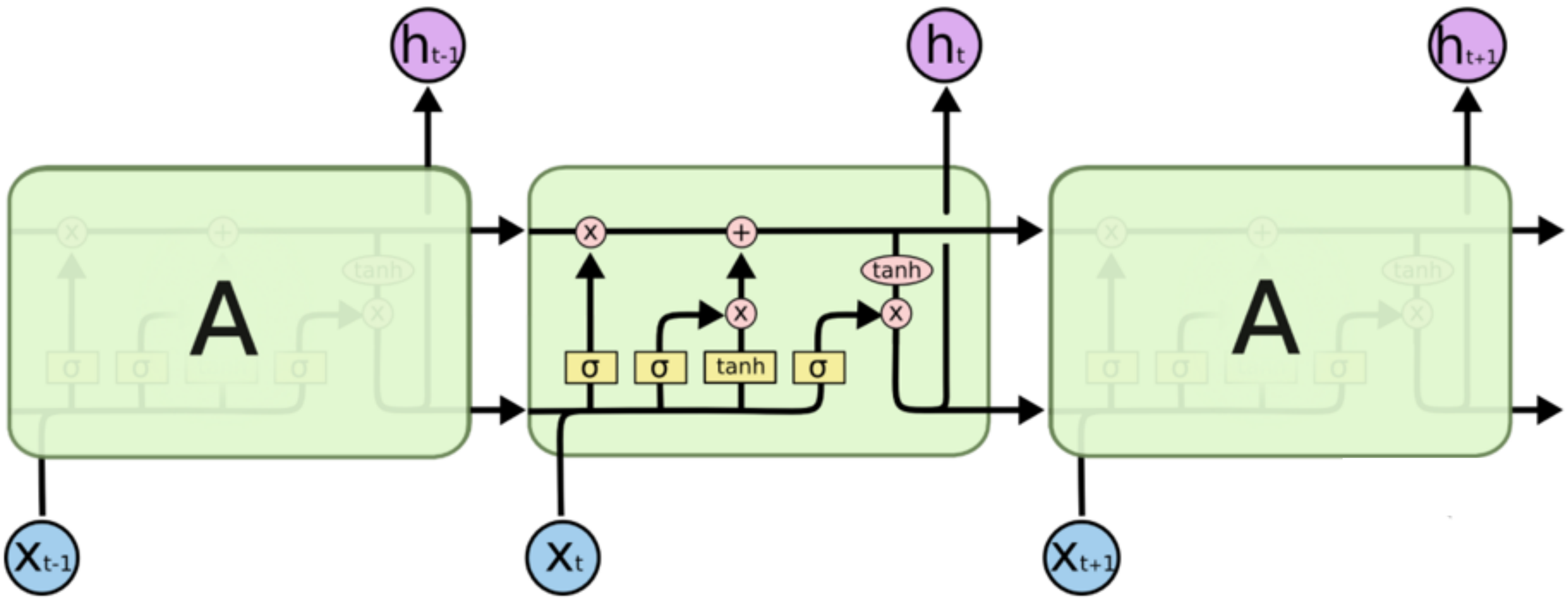
time step 2

Recurrent networks



RNNs often have a problem with long-distance dependencies.

LSTMs



http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Recurrent networks/LSTMs

task	x	y
language models	words in sequence	the next word in a sequence
part of speech tagging	words in sequence	part of speech
machine translation	words in sequence	translation

Midterm report, due Friday

- 4 pages, citing 10 relevant sources
- Be sure to consider feedback!
- Data collection should be completed
- You should specify a validation strategy to be performed at the end
- Present initial experimental results