Competition Results

Public leaderboard

Erica Chen	0.96961	1	9d
JTNT	0.96961	12	6d
Haas Analytics	0.96408	48	6d
InsertName	0.96132	25	6d
Stats Bears	0.96132	16	5d

Private leaderboard

—	Erica Chen	0.96213	1	9d
—	JTNT	0.96094	12	6d
▲ 2	Stats Bears	0.95739	16	5d
▼ 1	Haas Analytics	0.95621	48	6d
▲ 2	OG Lytics	0.94911	18	6d

Competition mini presentations (2-4 slides) this Thursday

Midterm

- Will be handed back the Tuesday after Spring break (April 2nd)
- We'll go over answers that day
- Requests for re-grading will be made that same day

Final Projects

- Dataset "pitches" (1 slide) due 2 weeks from this Thursday (April 4th)
- A pitch can be made by a group or an individual
- Eventual final project groups ***must*** be between 3 and 6 people
- Your project idea must be approved by instructors
- Only datasets pitched on that day are eligible for final projects

Skip-grams
"word2vec"

Data Mining & Analytics

Prof. Zach Pardos

INFO254/154: Spring '19

Skip-grams

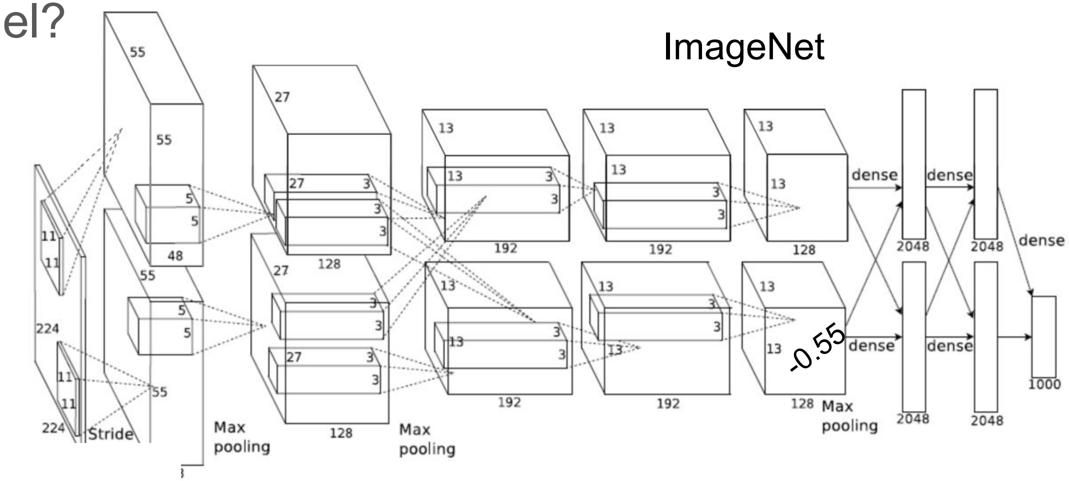
- Simple neural networks
- Learn vector representations of words from a large corpus of text.
- Can be used to explore the relationship between words in vector space
    - Similar words
    - Analogous relationships (Big is to Bigger as Small is to ___)
- Supervised objective, which learns unsupervised (unlabeled) structure

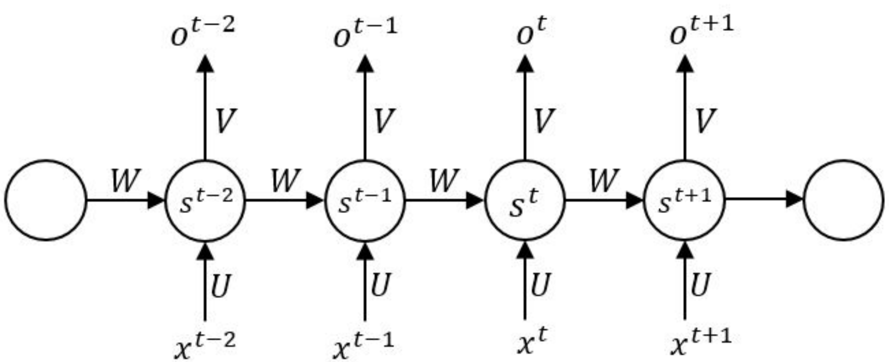
"Deep learning" vs Feed-forward neural networks

What is considered a "Deep" model?

- Many layers
  - Non-linearity
- Deep representations
- Many time slices

ImageNet





Recurrent neural network

Skip-grams

- Are not deep
- Use representation learning (embedding)
- Are big data models

What can they do?

vector[“KING”] - vector[“MAN”] + vector[“WOMAN”] ≈ vector[“QUEEN”]

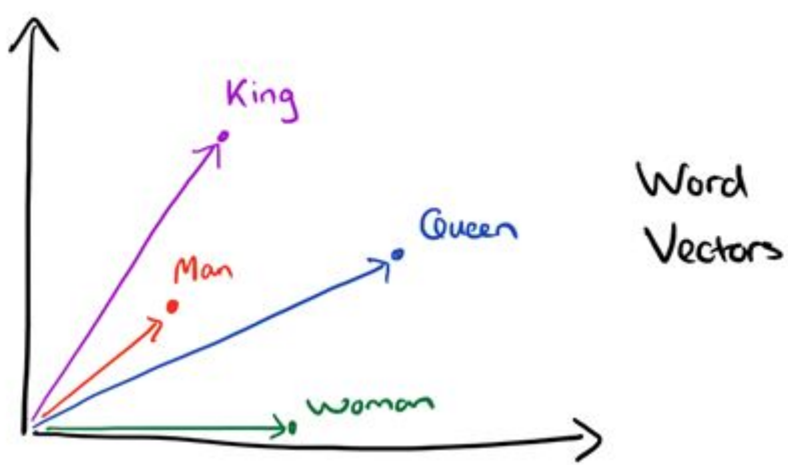
What concepts are involved in this arithmetic?

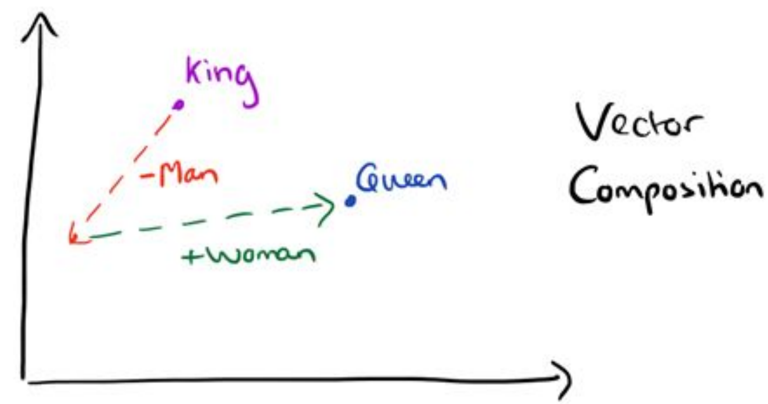
What can they do?

Similarity

$$\cos(\boldsymbol{a}, \boldsymbol{b}) = \frac{\boldsymbol{a} \cdot \mathbf{b}}{\|\boldsymbol{a}\|_2 \|\boldsymbol{b}\|_2}$$

vector[“KING”] - vector[“MAN”] + vector[“WOMAN”] ≈ vector[“QUEEN”]





Amazing power of word vectors (blog) - Linguistic regularities (original academic paper)

What can they do?

Complete a relationship pair based on an example relationship

Relationship	Example 1	Example 2	Example 3
France - Paris	Italy: Rome	Japan: Tokyo	Florida: Tallahassee
big - bigger	small: larger	cold: colder	quick: quicker
Miami - Florida	Baltimore: Maryland	Dallas: Texas	Kona: Hawaii
Einstein - scientist	Messi: midfielder	Mozart: violinist	Picasso: painter
Sarkozy - France	Berlusconi: Italy	Merkel: Germany	Koizumi: Japan
copper - Cu	zinc: Zn	gold: Au	uranium: plutonium
Berlusconi - Silvio	Sarkozy: Nicolas	Putin: Medvedev	Obama: Barack
Microsoft - Windows	Google: Android	IBM: Linux	Apple: iPhone
Microsoft - Ballmer	Google: Yahoo	IBM: McNealy	Apple: Jobs
Japan - sushi	Germany: bratwurst	France: tapas	USA: pizza

Amazing power of word vectors (blog) - Linguistic regularities (original academic paper)

What can they do?

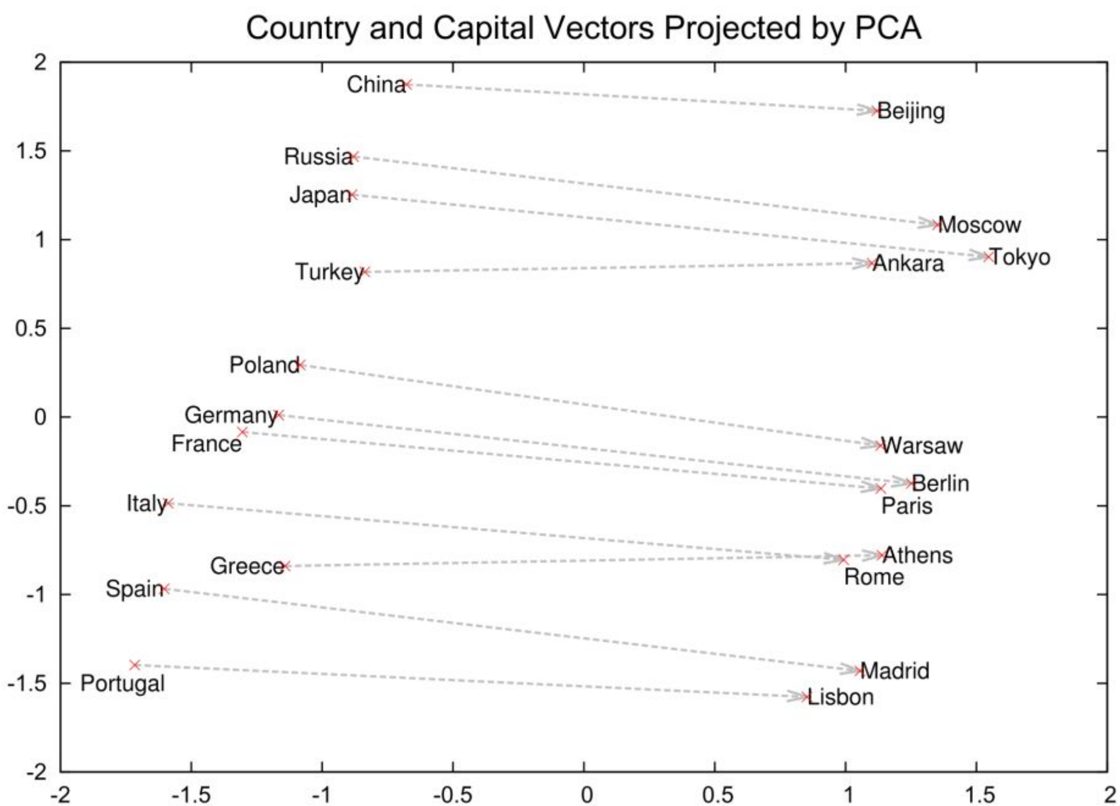
They complete analogies with about 60% accuracy
(results as high as 74% with modified approaches)

Model Architecture	Semantic-Syntactic Word Relationship test set		MSR Word Relatedness Test Set [20]
	Semantic Accuracy [%]	Syntactic Accuracy [%]	
RNNLM	9	36	35
NNLM	23	53	47
CBOW	24	64	61
Skip-gram	55	59	56

Amazing power of word vectors (blog) - Linguistic regularities (original academic paper)

What can they do?

PCA visualization of country and capital words



[switch to PPT]

What can they do?

They complete analogies with about 60% accuracy
(results as high as 74% with modified approaches)

Model Architecture	Semantic-Syntactic Word Relationship test set		MSR Word Relatedness Test Set [20]
	Semantic Accuracy [%]	Syntactic Accuracy [%]	
RNNLM	9	36	35
NNLM	23	53	47
CBOW	24	64	61
Skip-gram	55	59	56

Amazing power of word vectors (blog) - Linguistic regularities (original academic paper)

Feed-forward neural network

Multilayer Perceptron

Input size = 3
Output size = 2

$X =$

0	1	0

$W_{xh} =$

0.6948	0.0344
0.3171	0.4387
0.9502	0.3816

$h =$

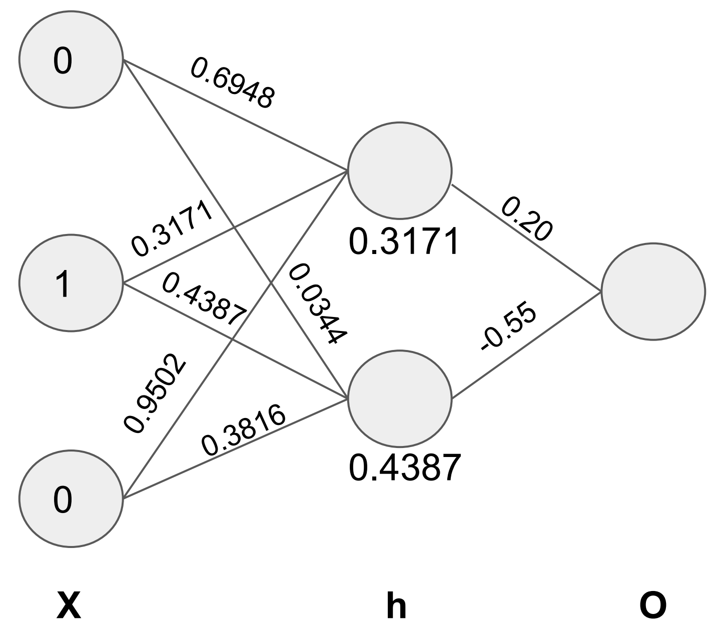
0.3171	0.4387

$W_{ho} =$

0.20
-0.55

$$O = h^*W_{ho} = -0.1779$$

No bias, no squashing (activation) function



Feed-forward neural network

Multilayer Perceptron / Skip-gram

Input size = 3
Output size = 3

$X =$

0	1	0

$W_{xh} =$

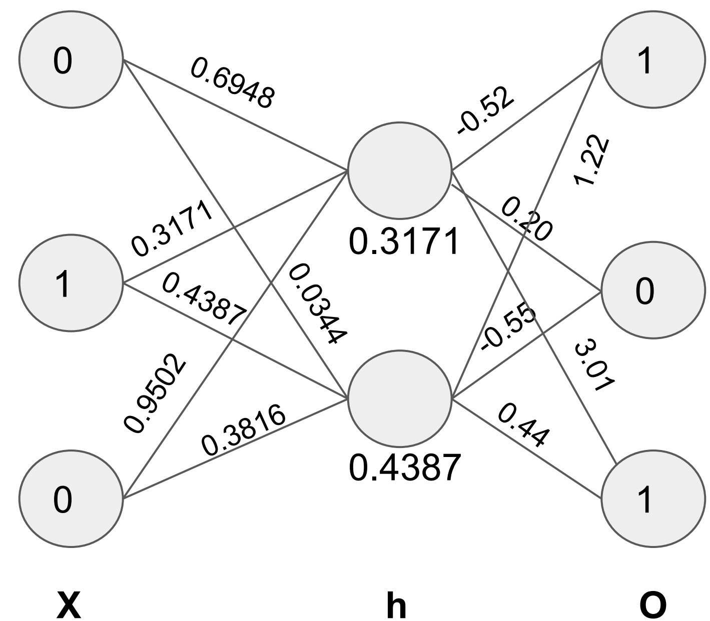
0.6948	0.0344
0.3171	0.4387
0.9502	0.3816

$h =$

0.3171	0.4387

$W_{ho} =$

-0.52	0.20	3.01
1.22	-0.55	0.44



Feed-forward neural network

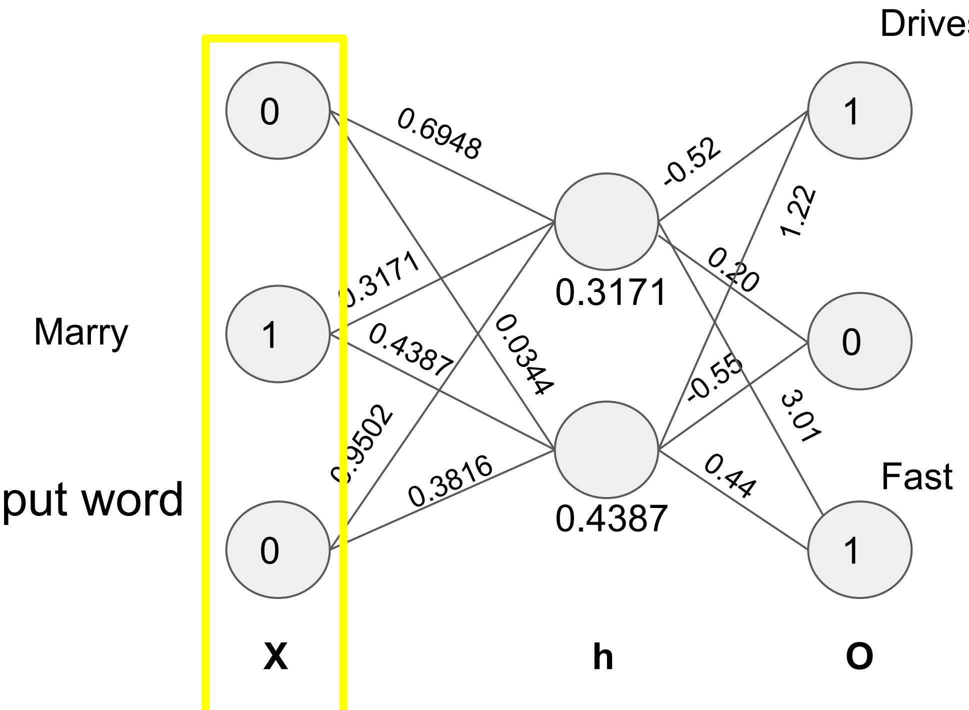
Multilayer Perceptron / Skip-gram

Input size = 3
Output size = 3

Word input one-hot

Consider a sentence:
Marry drives fast

Skip-gram predicts the context of the input word

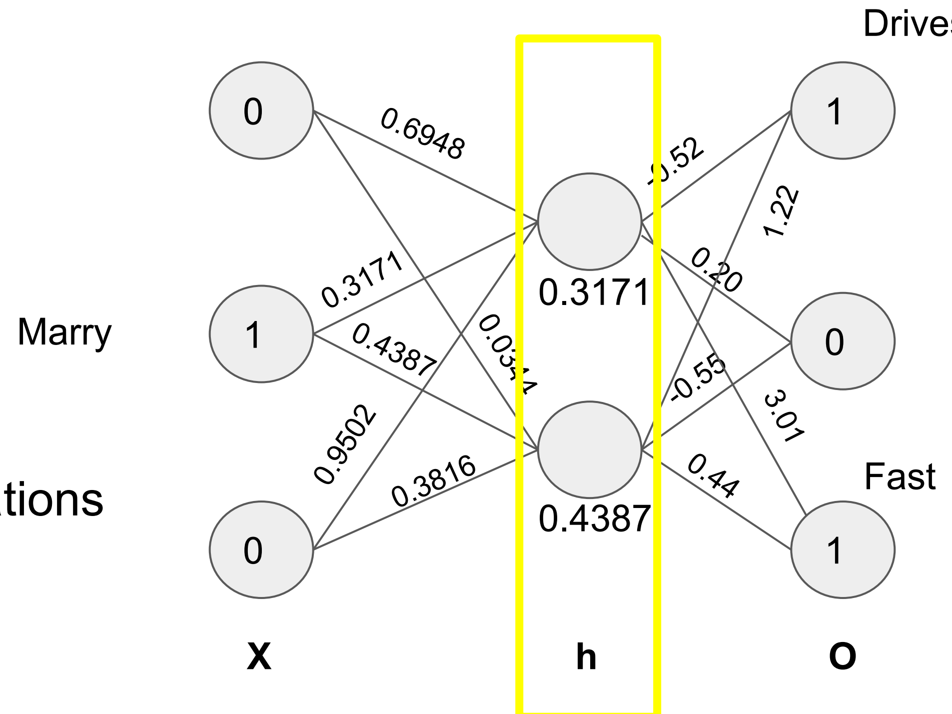


Feed-forward neural network

Multilayer Perceptron / Skip-gram

Input size = 3
Output size = 3

Continuous representation of word (embedding)



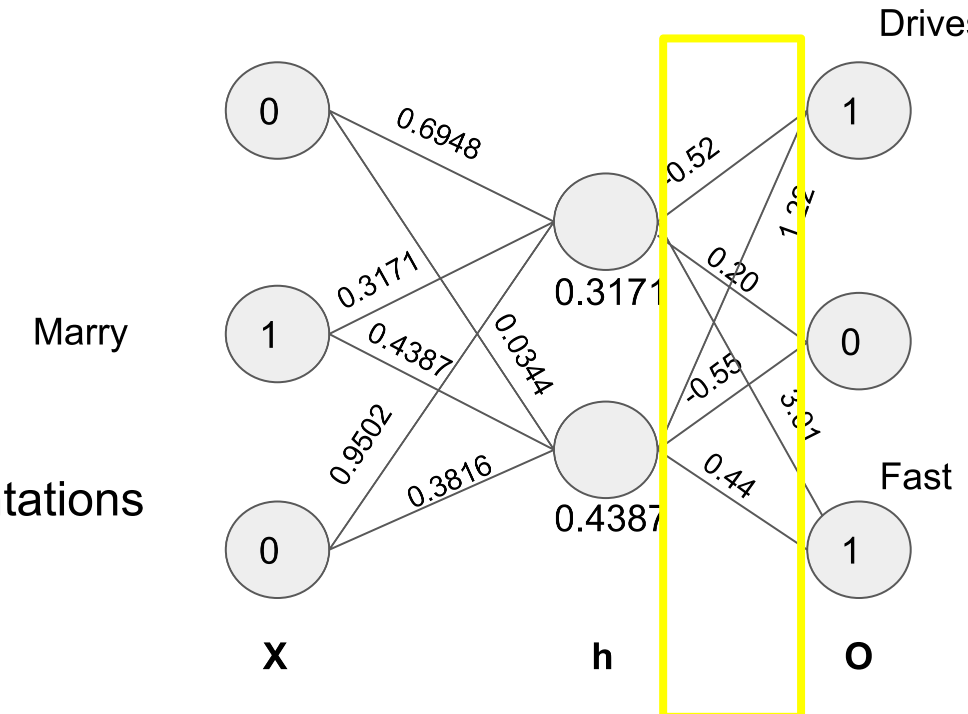
Wxh weights are the learned representations of the words in the vocabulary

Feed-forward neural network

Multilayer Perceptron / Skip-gram

Input size = 3
Output size = 3

Also a continuous representation of words (currently ignored)



Who weights are also learned representations of the words in the vocabulary

Feed-forward neural network

Multilayer Perceptron / Skip-gram

Input size = 3
Output size = 3

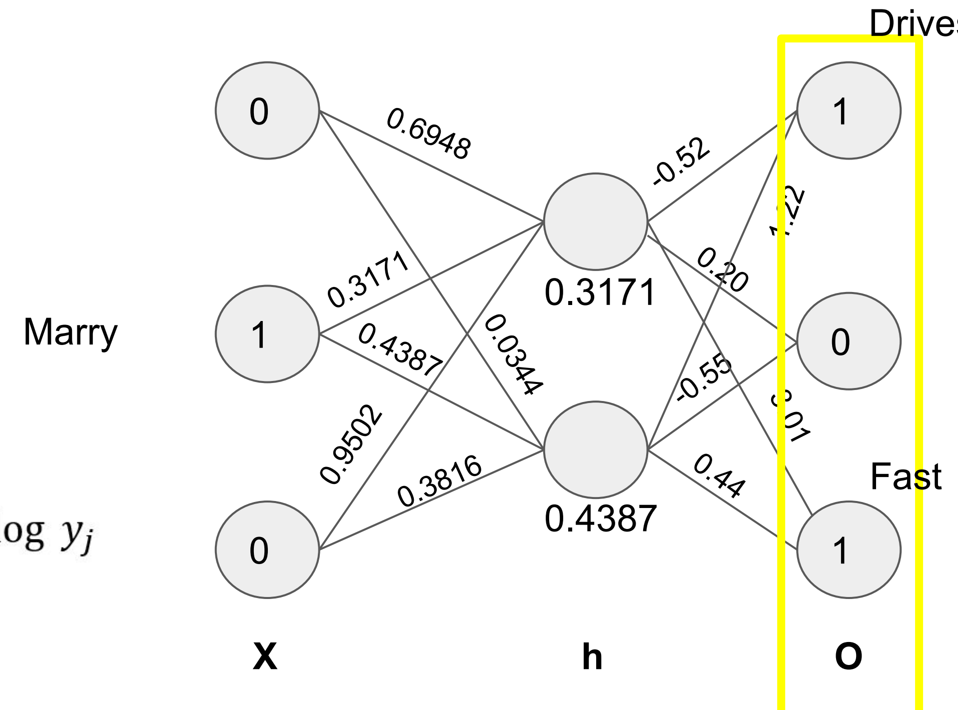
Activation: softmax

$$y_i = \frac{e^{z_i}}{\sum_{j \in Classes} e^{z_j}}$$

Loss: categorical cross-entropy

$$C = -\sum_{j \in Classes} t_j \log y_j$$

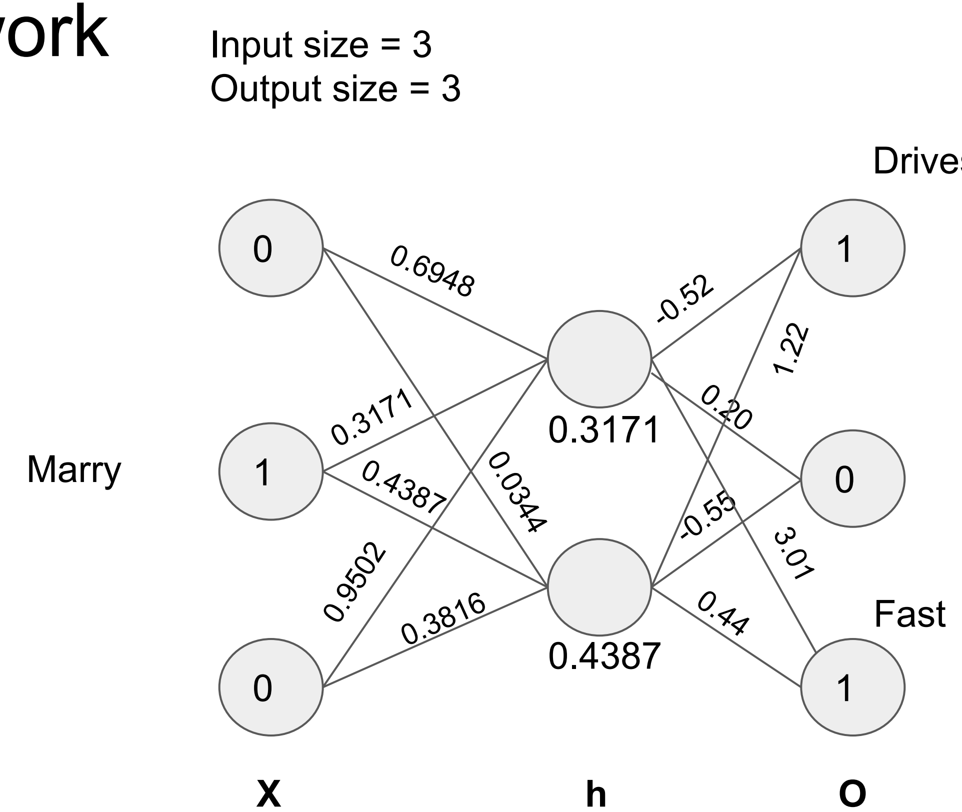
Alternative loss: binary cross-entropy for negative sampling variant



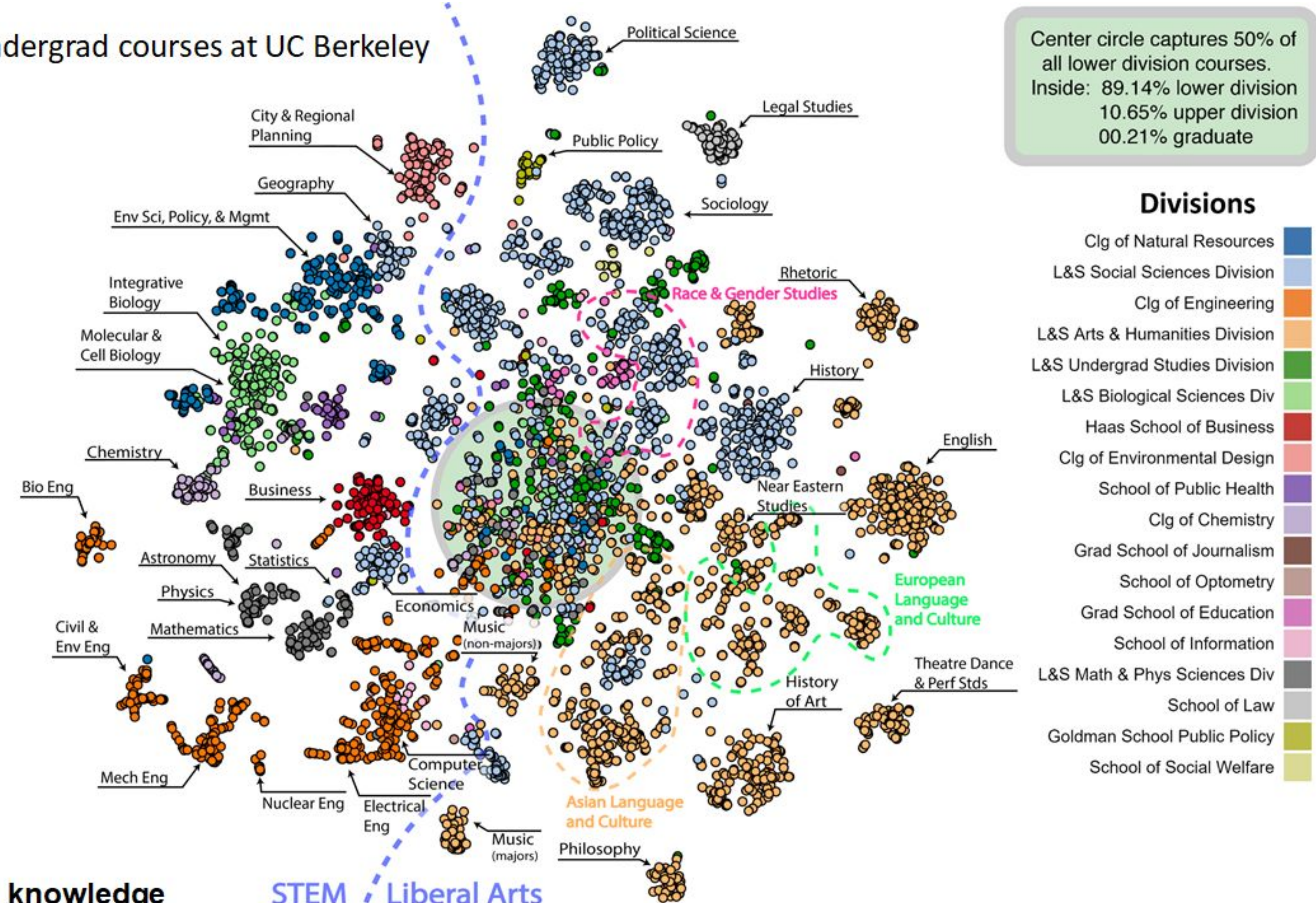
Feed-forward neural network

Multilayer Perceptron / Skip-gram

Skip-grams have a single word as input
context words as output



Visualization of all undergrad courses at UC Berkeley

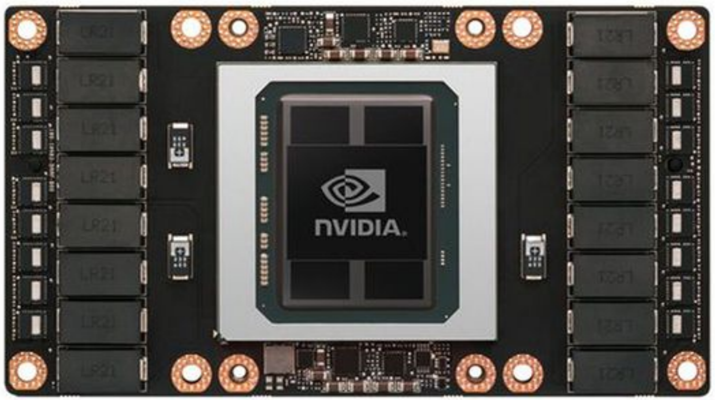


Pardos & Nam (2018)

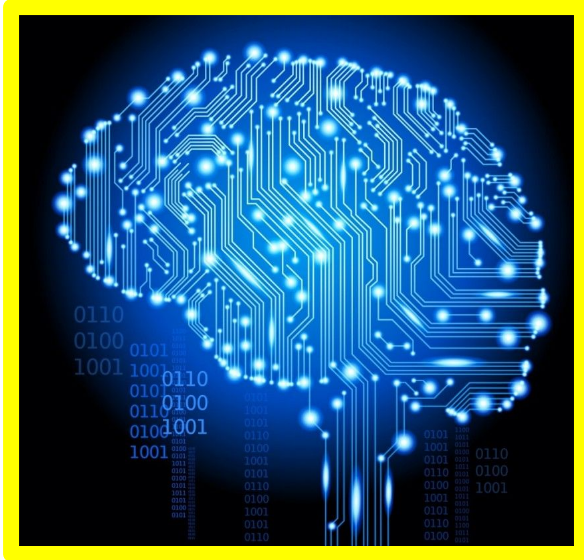
http://tiny.cc/**map_of_knowledge**

Neural Networks: Abstractions

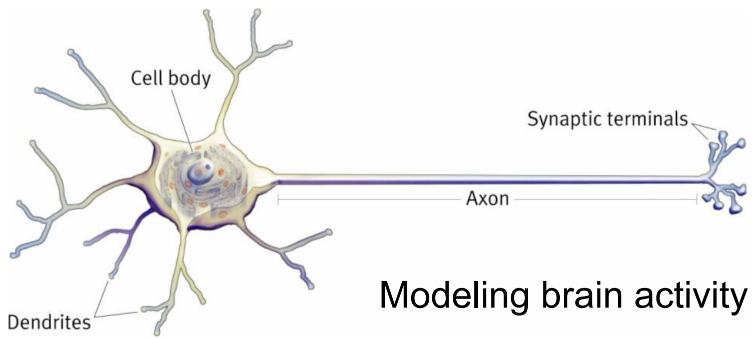
Hardware / Software optimization



Model of the mind



Office Hours starts now (migrating to BWW 4232)



Modeling brain activity