NLP and Word Embeddings

Word representation

deeplearning.ai

Word representation

V = [a, aaron, ..., zulu, <UNK>]

$|V| = 10,000$

1-hot representation

$$
\begin{array}{cccccc}
\text{Man} & \text{Woman} & \text{King} & \text{Queen} & \text{Apple} & \text{Orange} \\
(5391) & (9853) & (4914) & (7157) & (456) & (6257) \\
\begin{bmatrix}0\\0\\0\\0\\\vdots\\1\\\vdots\\0\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\1\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\\vdots\\1\\\vdots\\0\\0\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\1\\\vdots\\0\end{bmatrix} &
\begin{bmatrix}0\\\vdots\\1\\\vdots\\0\\0\\0\\0\\0\end{bmatrix} &
\begin{bmatrix}0\\0\\0\\0\\0\\\vdots\\1\\\vdots\\0\end{bmatrix} \\
O_{5391} & O_{9853} & & & &
\end{array}
$$

I want a glass of $\boxed{\text{orange juice}}$.

I want a glass of $\boxed{\text{apple ?}}$.

Andrew Ng

Featurized representation: word embedding

	Man (5391)	Woman (9853)	King (4914)	Queen (7157)	Apple (456)	Orange (6257)
Gender	-1	1	-0.95	0.97	0.00	0.01
Royal	0.01	0.02	0.93	0.95	-0.01	0.00
Age	0.03	0.02	0.7	0.69	0.03	-0.02
Food	0.04	0.01	0.02	0.01	0.95	0.97
⋮	⋮	⋮				

300

size
cost
alive
verb

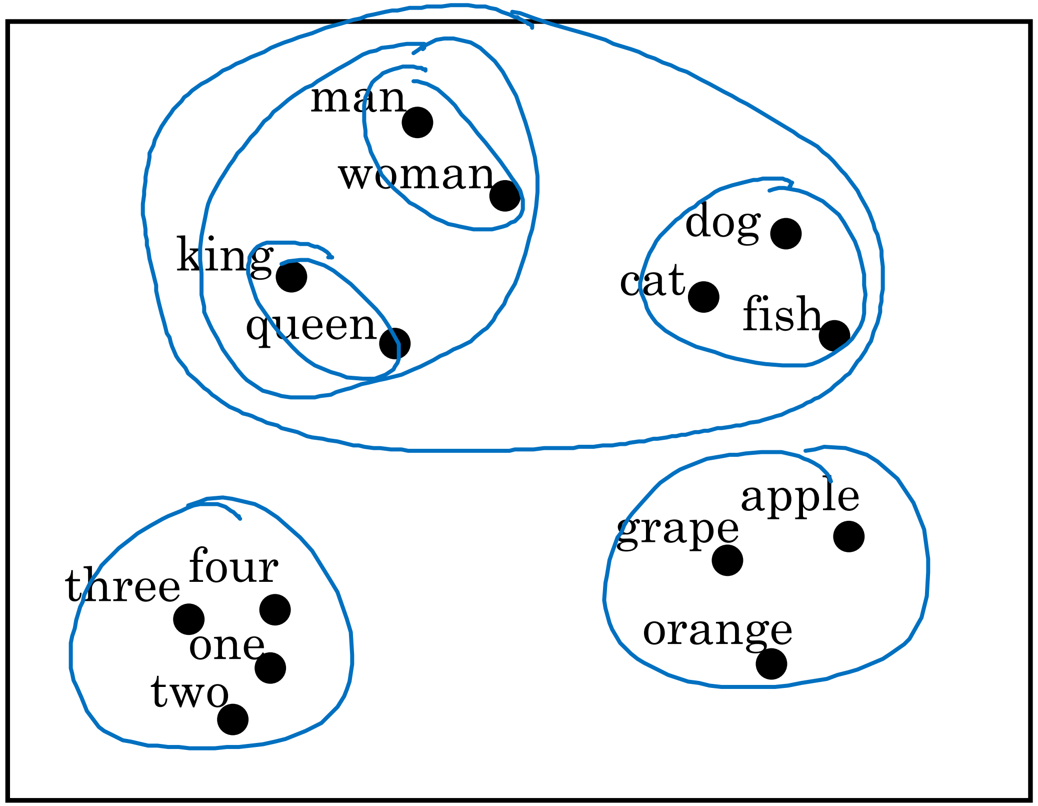
e_{5391} e_{9853}

I want a glass of orange juice.

I want a glass of apple juice.

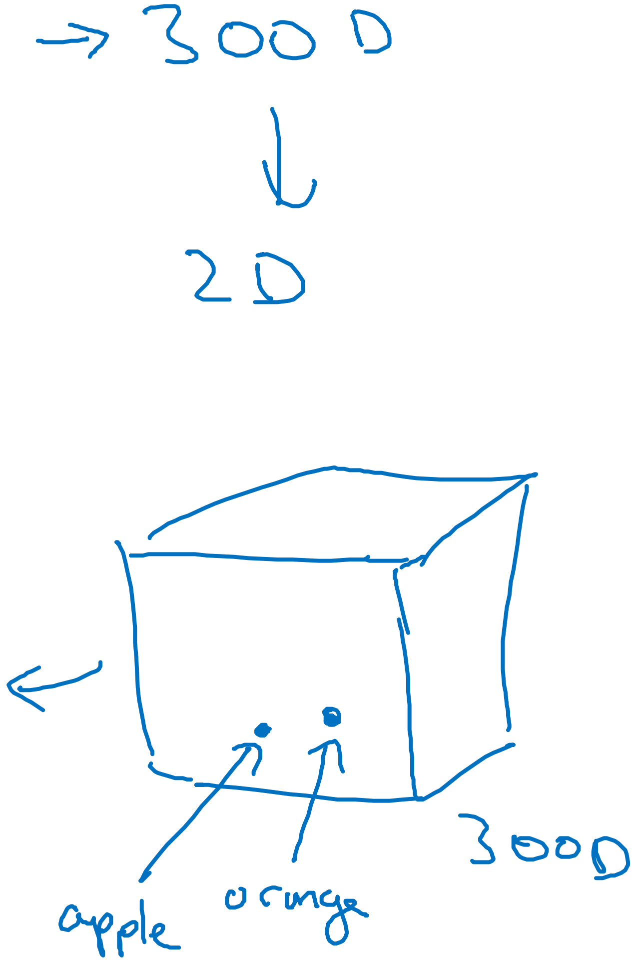
Andrew Ng

Visualizing word embeddings



t-SNE

A non-linear dimensionality reduction technique



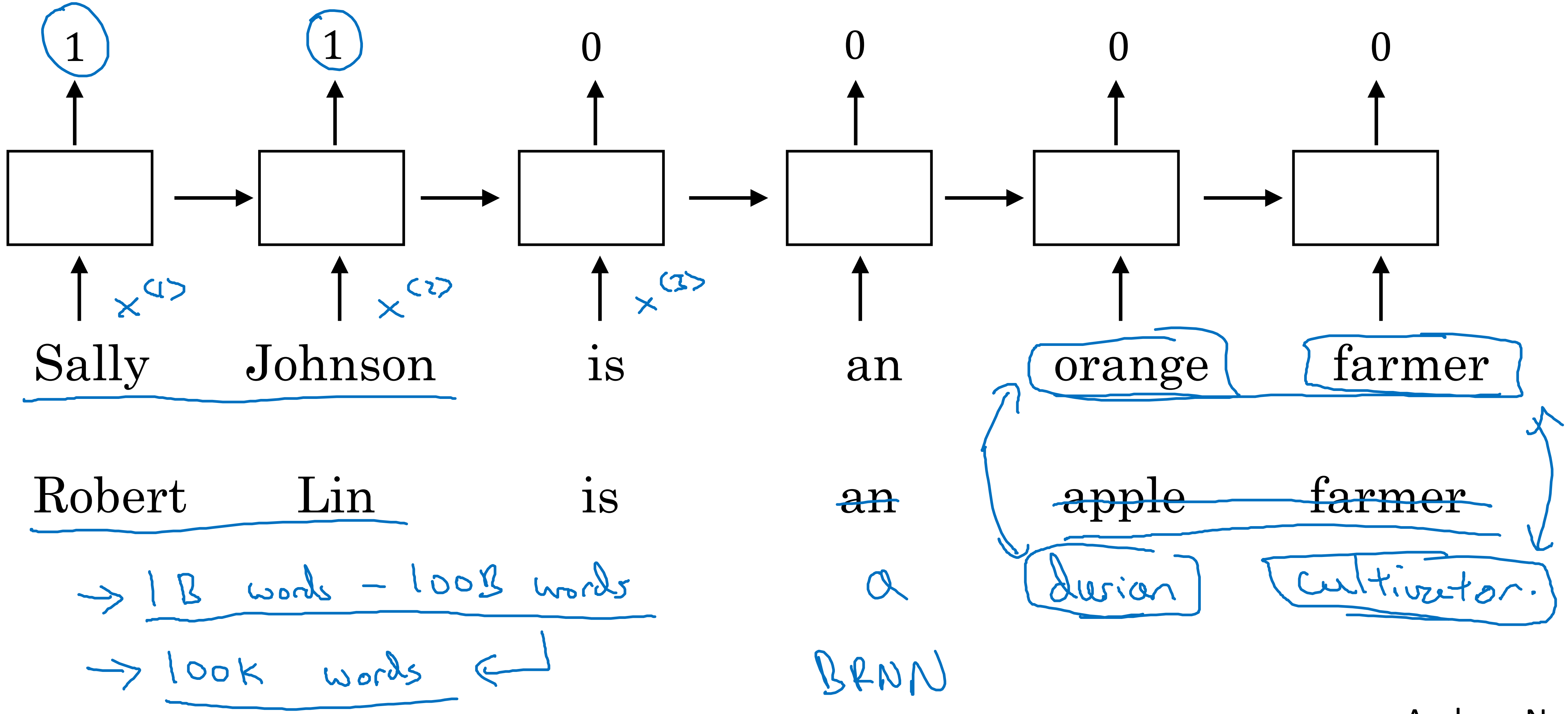
[van der Maaten and Hinton., 2008. Visualizing data using t-SNE]

deeplearning.ai

NLP and Word Embeddings

Using word embeddings

Named entity recognition example



Transfer learning and word embeddings

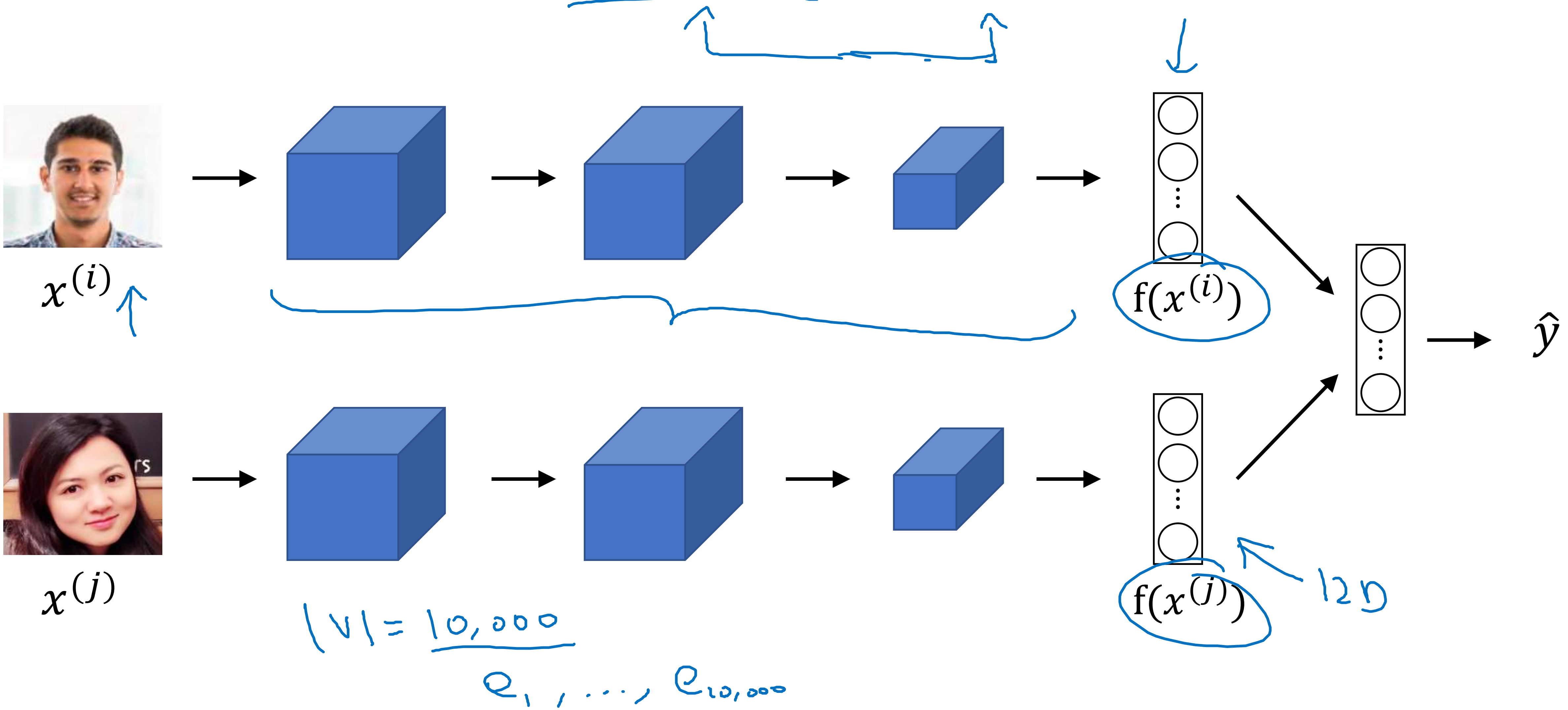
A
1. Learn word embeddings from large text corpus. (1-100B words)

   (Or download pre-trained embedding online.)

B
2. Transfer embedding to new task with smaller training set. (say, 100k words) → 10,000 → 300

3. Optional: Continue to finetune the word embeddings with new data.

Relation to face encoding (embedding) 128D



[Taigman et. al., 2014. DeepFace: Closing the gap to human level performance]

Andrew Ng

NLP and Word Embeddings

deeplearning.ai

Properties of word embeddings

Analogies

	Man (5391)	Woman (9853)	King (4914)	Queen (7157)	Apple (456)	Orange (6257)
Gender	−1	1	-0.95	0.97	0.00	0.01
Royal	0.01	0.02	0.93	0.95	-0.01	0.00
Age	0.03	0.02	0.70	0.69	0.03	-0.02
Food	0.09	0.01	0.02	0.01	0.95	0.97

e_{5391}

e_{man}

e_{woman}

Man → Woman as King →? Queen

$e_{man} - e_{woman} \approx e_{king} - e_?$

? ← queen

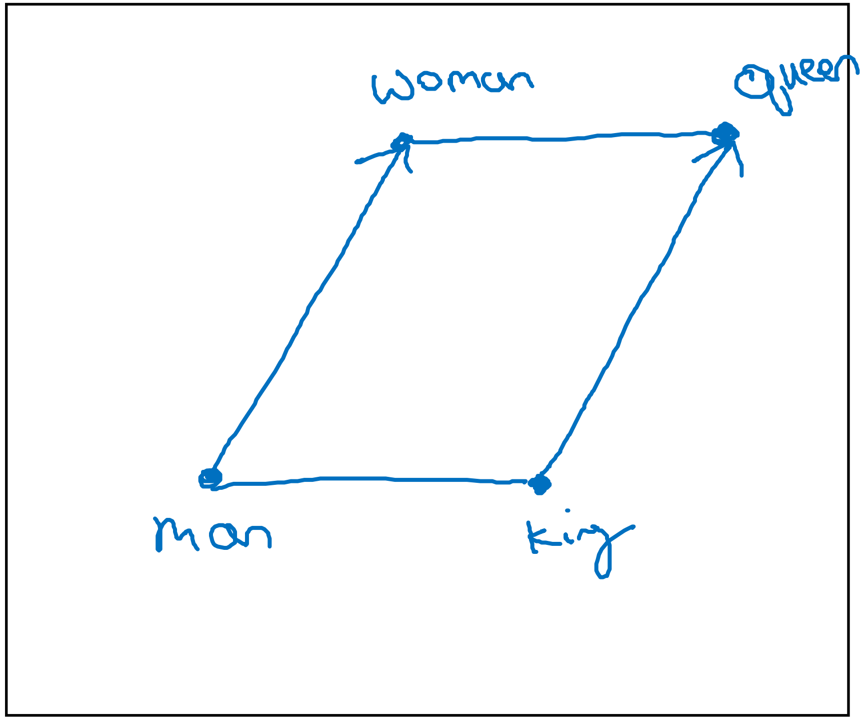
$e_{man} - e_{woman} \approx \begin{bmatrix} -2 \\ 0 \\ 0 \\ 0 \end{bmatrix}$

$e_{king} - e_{queen} \approx \begin{bmatrix} -2 \\ 0 \\ 0 \\ 0 \end{bmatrix}$

[Mikolov et. al., 2013, Linguistic regularities in continuous space word representations]

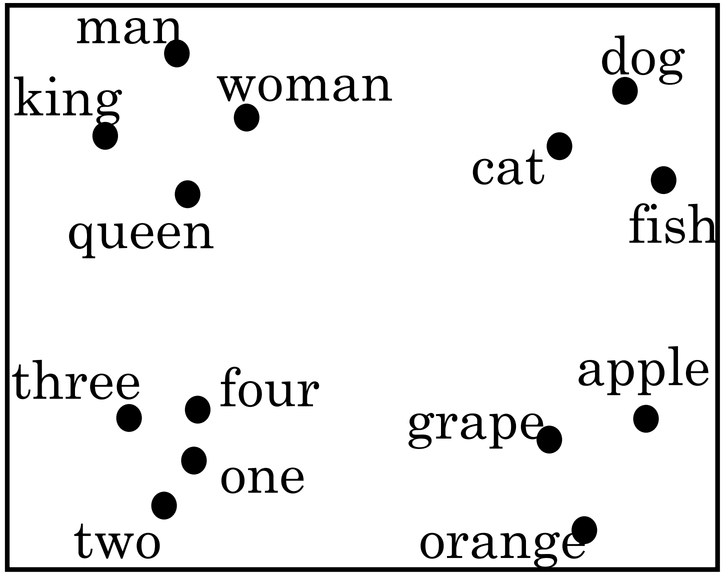
Andrew Ng

Analogies using word vectors



300 D

300D → 2D



t-SNE

$$\underbrace{e_{man} - e_{woman} \approx e_{king} - \cancel{e_?}}\ e_w$$

Find word w: $\arg\max_w \ \boxed{sim(e_w, \ e_{king} - e_{man} + e_{woman})}$

Similarity

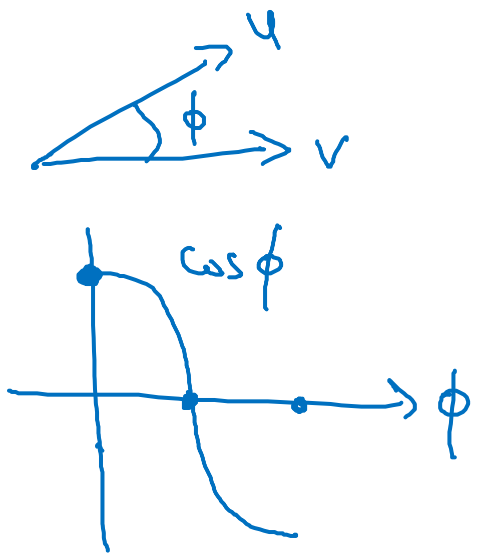
30 - 75%

Andrew Ng

Cosine similarity

$$\rightarrow \boxed{sim(e_w, e_{king} - e_{man} + e_{woman})}$$

$$\underline{sim(u,v)} = \frac{u^T v}{\|u\|_2 \|v\|_2}$$



$$\|u - v\|^2$$

Man:Woman as Boy:Girl

Ottawa:Canada as Nairobi:Kenya

Big:Bigger as Tall:Taller

Yen:Japan as Ruble:Russia

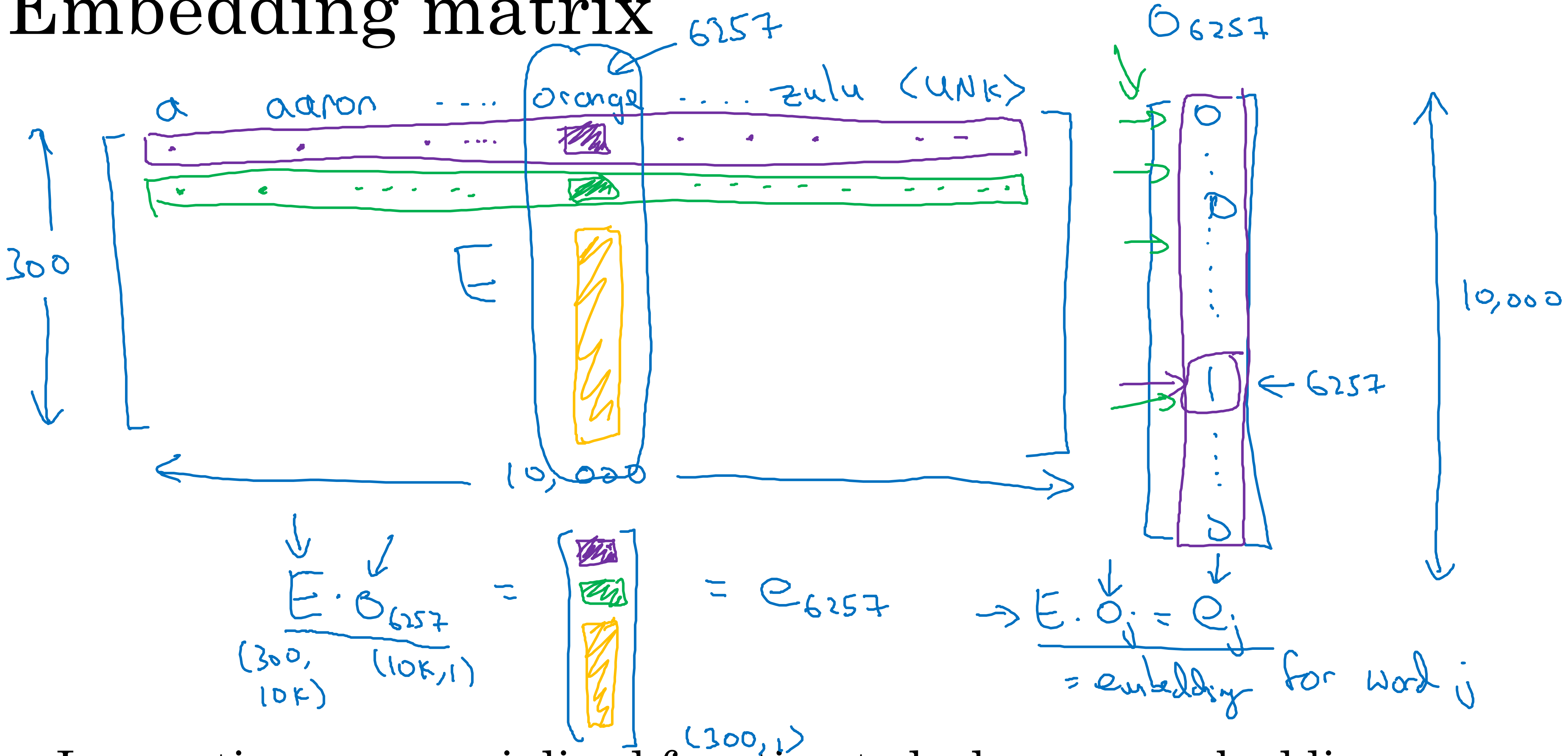
Andrew Ng

NLP and Word Embeddings

Embedding matrix

deeplearning.ai

Embedding matrix



In practice, use specialized function to look up an embedding.

→ Embedding

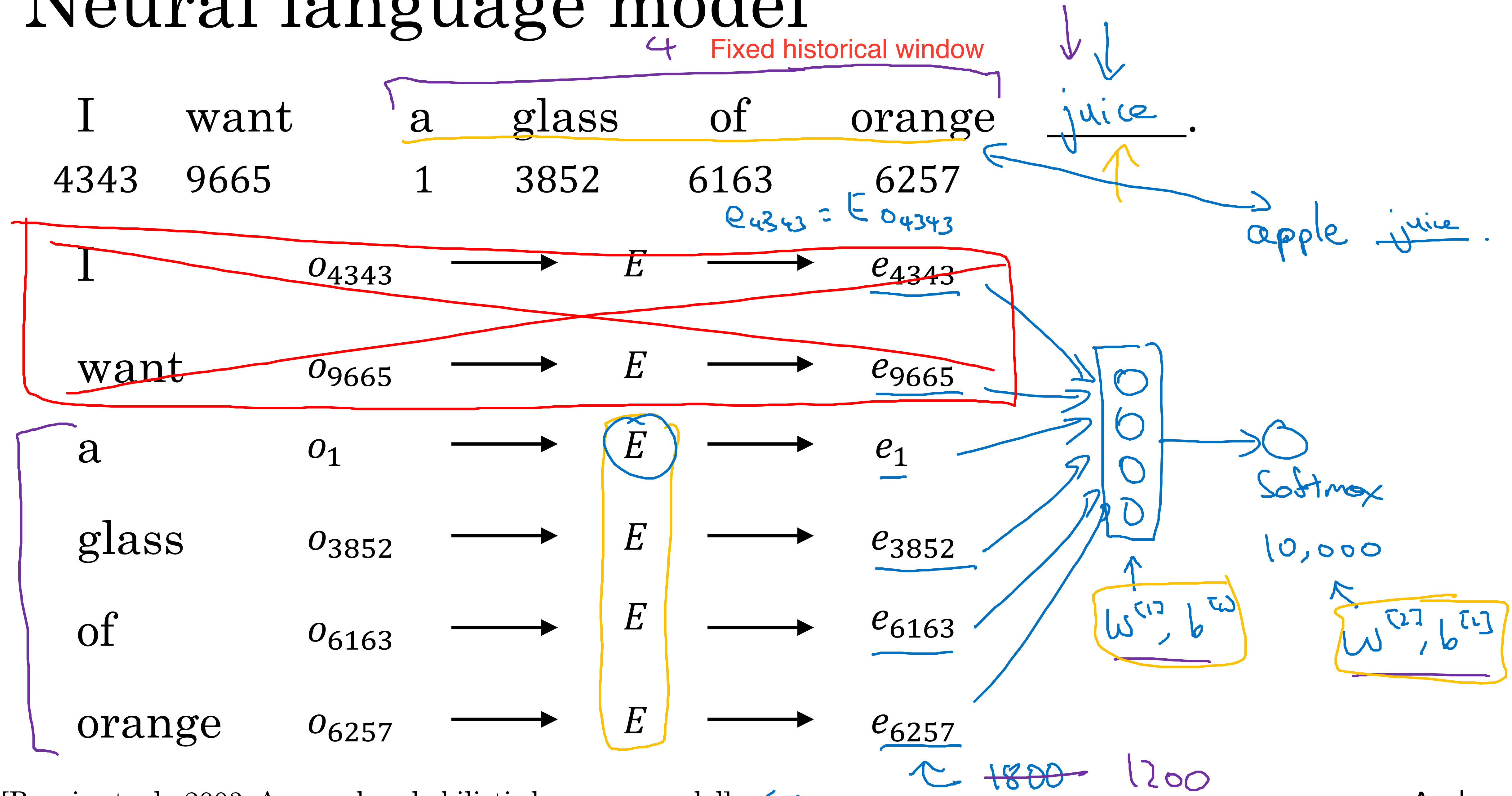
Andrew Ng

deeplearning.ai

NLP and Word Embeddings

Learning word embeddings

Neural language model



[Bengio et. al., 2003, A neural probabilistic language model]

Andrew Ng

Other context/target pairs

I want a glass of orange juice to go along with my cereal.

context

target

Context: Last 4 words.

4 words on left & right
of the blank

a glass of orange ? to go along with

Last 1 word before the blank

orange ?

Nearby 1 word

glass ?

skip gram model

simipler algorithm

NLP and Word Embeddings

deeplearning.ai

Word2Vec

Skip-grams

I want a glass of orange juice to go along with my cereal.

randomly choose a target word

Context	Target
orange	juice
orange	glass
orange	my

[Mikolov et. al., 2013. Efficient estimation of word representations in vector space.]

Model

Vocab size = 10,000k

based on supervised learning

$x \longrightarrow y$

Context c ("orange") 6257 $\longrightarrow$ Target t ("juice") 4834

$O_c \longrightarrow E \longrightarrow e_c \longrightarrow \underset{\text{softmax}}{O} \longrightarrow \hat{y}$

one-hot vector

$e_c = E O_c$

e^T = e’s transpose

Softmax: $p(t|c) = \dfrac{e^{\theta_t^T e_c}}{\sum_{j=1}^{10,000} e^{\theta_j^T e_c}}$

θ_t = parameter associated with output t

associated

$\mathcal{L}(\hat{y}, y) = - \sum_{i=1}^{10,000} y_i \log \hat{y}_i$

$$y = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} \leftarrow 4834$$

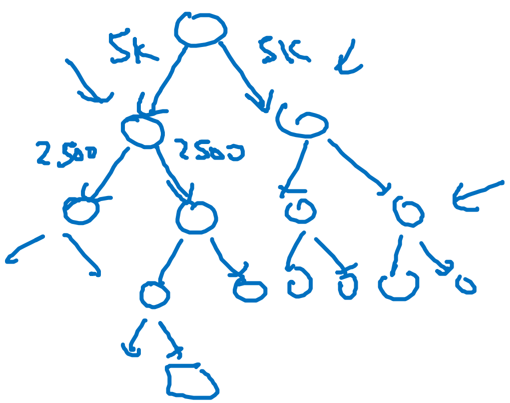
Andrew Ng

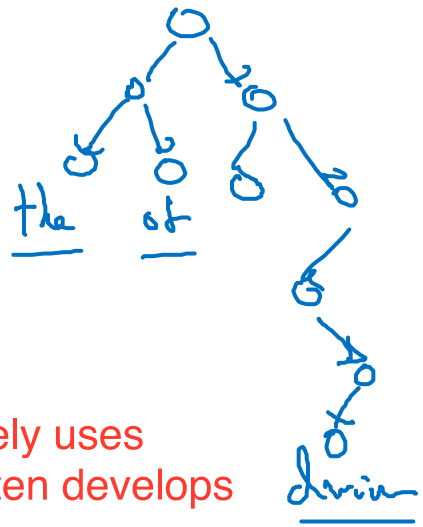
Problems with softmax classification

$$p(t|c) = \frac{e^{\theta_t^T e_c}}{\boxed{\sum_{j=1}^{10,000} e^{\theta_j^T e_c}}}$$

Hierachial softmax to address computational cost

log |V| rather than linear vocab size





In practice, Hierachical softmax rarely uses balanced (symmetric) tree. Instead, it often develops trees of common words

How to sample the context c?

t

→ the, of, a, and, to, ...

$c \rightarrow t$

→ orange, apple, durian

$P(c)$

durian

Andrew Ng

NLP and Word Embeddings

deeplearning.ai

Negative sampling

Defining a new learning problem

I want a glass of orange juice to go along with my cereal.

x → y

context	word	target?
orange	juice	1
orange	king	0
orange	book	0
orange	the	0
orange	of	0

k

k = 5-20 smaller datasets

k = 2-5 larger dataset

[Mikolov et. al., 2013. Distributed representation of words and phrases and their compositionality]

Model

Softmax: $$p(t|c) = \frac{e^{\theta_t^T e_c}}{\sum_{j=1}^{10,000} e^{\theta_j^T e_c}}$$

10,000-way softmax

$$P(y=1 \mid c, t) = \sigma(\theta_t^T e_c)$$

basically logistic regression model

context	word	target?
orange	juice	1
orange	king	0
orange	book	0
orange	the	0
orange	of	0

x: context, word; y: target?

c, t, y

k

possible target

Orange 6257

$o_{6257} \rightarrow E \rightarrow e_{6257}$

juice?

king

10,000

10,000 binary classification problem

$k+1$

k negative examples.
+ 1 positive example

instead of having a giant 10,000 way softmax, turn it into 10,000 binary logistic regression problems

Andrew Ng

Selecting negative examples

$$\overbrace{\text{context} \quad \text{word}}^{x} \quad \overbrace{\text{target?}}^{y}$$

context	word	target?
orange	juice	1
orange	king	0
orange	book	0
orange	the	0
orange	of	0

t

the, of, and, ...

One extreme case is to sample the words in the middle (whatever was the observed distribution in the training set) but it's not representative; Another extreme, which is to take uniformly random (1/v), is also not representative of english words. So, the authors take the heuristic value (between two extremes of sampling from the empirical frequencies). They sampled proportional to the frequency of a word to the power of three forth. f(Wi) is the observed frequency of a particular word in the english language

$$P(w_i) = \frac{f(w_i)^{3/4}}{\sum_{j=1}^{10{,}000} f(w_j)^{3/4}}$$

$$\frac{1}{|V|}$$

Andrew Ng

NLP and Word Embeddings

deeplearning.ai

GloVe word vectors

GloVe (global vectors for word representation)

I want a glass of orange juice to go along with my cereal.

c, t

X_{ij} = # times i appears in context of j.

$\underbrace{X_{ij}}$ = X_{ji}

X_ij is a count that captures how often do words i and j appear with each other, or close to each other.

Depending on the definition of context and target words, X_ij might be equal X_ji. However, if the choice of context is always the word immdeiately before the target word, X_ij and X_ji may not be symmetric. For the purpose of the GloVe algorithm, we can define context and target as whether or not the two words appear in close proximity. Say within plus or minus 10 words of each other.

[Pennington et. al., 2014. GloVe: Global vectors for word representation]

Model

$$\text{minimize} \sum_{i=1}^{10{,}000} \sum_{j=1}^{10{,}000} \underbrace{f(X_{ij})}_{\text{weighting term}} (\theta_i^T e_j + b_i + b_j' - \log X_{ij})^2$$

$t \quad c$

"$\theta_t^T e_c$"

Think of i and j as playing the role of c and t.

If X_ij is zero, its log is undefined and negatively infinity.. So we add an extra weighting term too sum over the terms where X_ij is zero.

How related are words i and j (measured by how often they occur with each other) is affected by X_ij.

$\rightarrow 0?$

$f(X_{ij}) = 0$ of $X_{ij} = 0$. "$0 \log 0$" $= 0$

by using a convention

If X_ij =0, 0log0 is not relevant. The sum is sum only over the pairs of words that have co-occurred at least once in that context-target relationship.

this, is, of, a, ...

durion

Stop words: some words that appear very often in the English language.

θ_i, e_j are symmetric

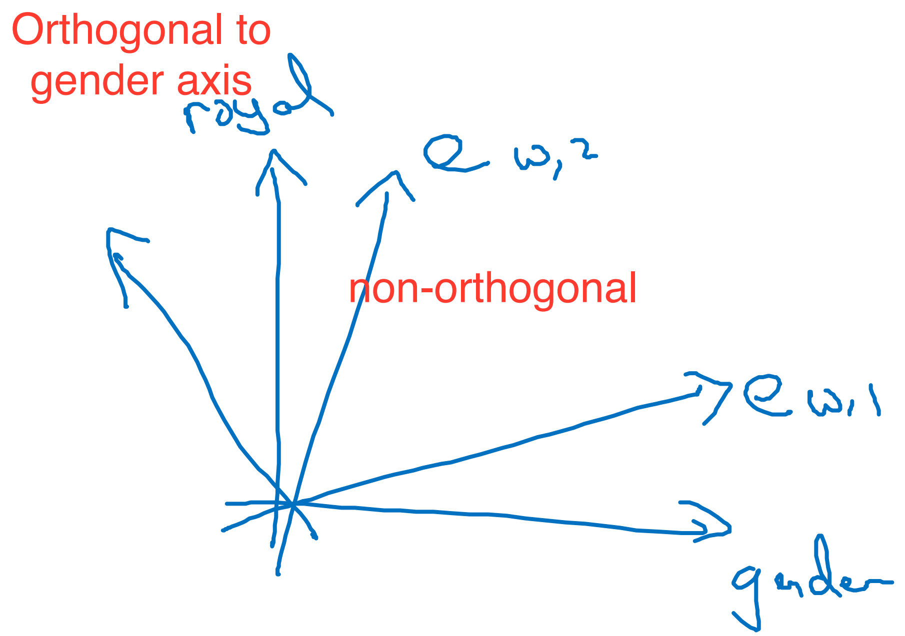
$$e_w^{(final)} = \frac{e_w + \theta_w}{2}$$

One way to train the algorithm is to initialize theta and e both uniformly around gradient descent to minimize its objective, and then take the average when being done for every word. It is possible because theta and e play symetric roles in this particular forumulation.

Andrew Ng

A note on the featurization view of word embeddings

	Man (5391)	Woman (9853)	King (4914)	Queen (7157)
Gender	−1	1	-0.95	0.97
Royal	0.01	0.02	0.93	0.95
Age	0.03	0.02	0.70	0.69
Food	0.09	0.01	0.02	0.01



We cannot guarantee that individual components of the embeddings are interpretable: the axis used to represent the features may not be well-aligned with what might be easily humanly interpretable axis.

$$\text{minimize} \sum_{i=1}^{10,000} \sum_{j=1}^{10,000} f(X_{ij})\left(\theta_i^T e_j + b_i + b'_j - \log X_{ij}\right)^2$$

If there was some invertible matrix A, then this could easily be replace with the following:

$$(A\theta_i)^T (A^{-T} e_j) = \theta_i^T \cancel{A^T} \cancel{A^{-T}} e_j$$

Despite this type (potentially arbitrary) of linear transformation, the parallelogram map still works

Andrew Ng

NLP and Word Embeddings

deeplearning.ai

Sentiment classification

Sentiment classification is the task of looking at a piece of text and telling if someone likes or dislikes the thing they are talking about.

Sentiment classification problem

One of the challenges for sentiment classification is you might not have a huge label training set .

With word embeddings, you can build good sentiment classifiers even with only modest-size lable training sets.

$x \longrightarrow y$

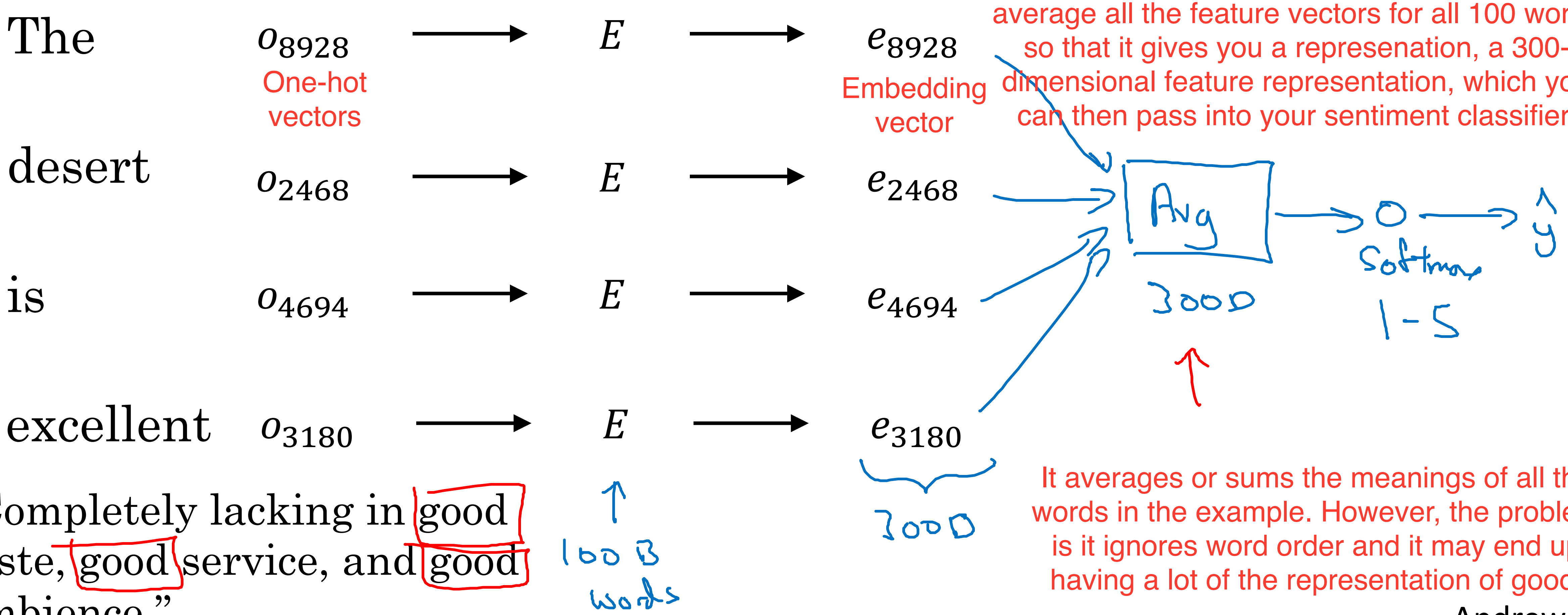
The dessert is excellent.	★★★★☆
Service was quite slow.	★★☆☆☆
Good for a quick meal, but nothing special.	★★★☆☆
Completely lacking in good taste, good service, and good ambience.	★☆☆☆☆

10,000 → 100,000 words

Andrew Ng

Simple sentiment classification model

The	dessert	is	excellent
8928	2468	4694	3180



Notice that by using the average operation here, this particular algorithm works for reviews that are short or long because ou can just sum or average all the feature vectors for all 100 words so that it gives you a represenation, a 300-dimensional feature representation, which you can then pass into your sentiment classifier.

"Completely lacking in good taste, good service, and good ambience."

It averages or sums the meanings of all the words in the example. However, the problem is it ignores word order and it may end up having a lot of the representation of good.

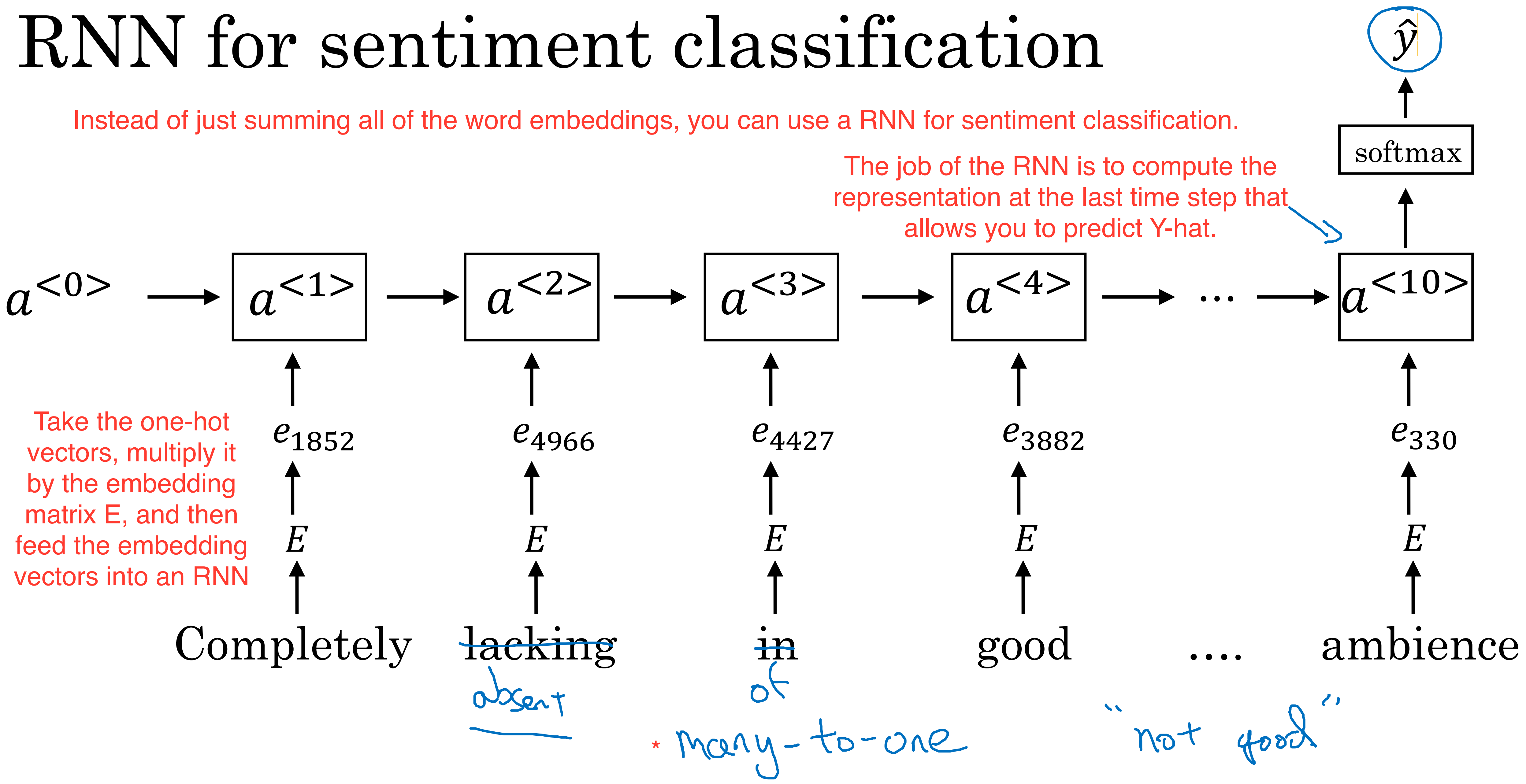
Andrew Ng

RNN for sentiment classification

Instead of just summing all of the word embeddings, you can use a RNN for sentiment classification.

The job of the RNN is to compute the representation at the last time step that allows you to predict Y-hat.

Take the one-hot vectors, multiply it by the embedding matrix E, and then feed the embedding vectors into an RNN



Andrew Ng

deeplearning.ai

NLP and Word Embeddings

Machine learning and AI algorithms are increasingly trusted to help with, or to make important decisions.

Debiasing word embeddings

The problem of bias in word embeddings

Bias here is not bias variants but gender bias or ethnicity bias.

Man:Woman as King:Queen

Unhealthy gender stereotype

Man:Computer_Programmer as Woman: Homemaker

Father:Doctor as Mother: Nurse

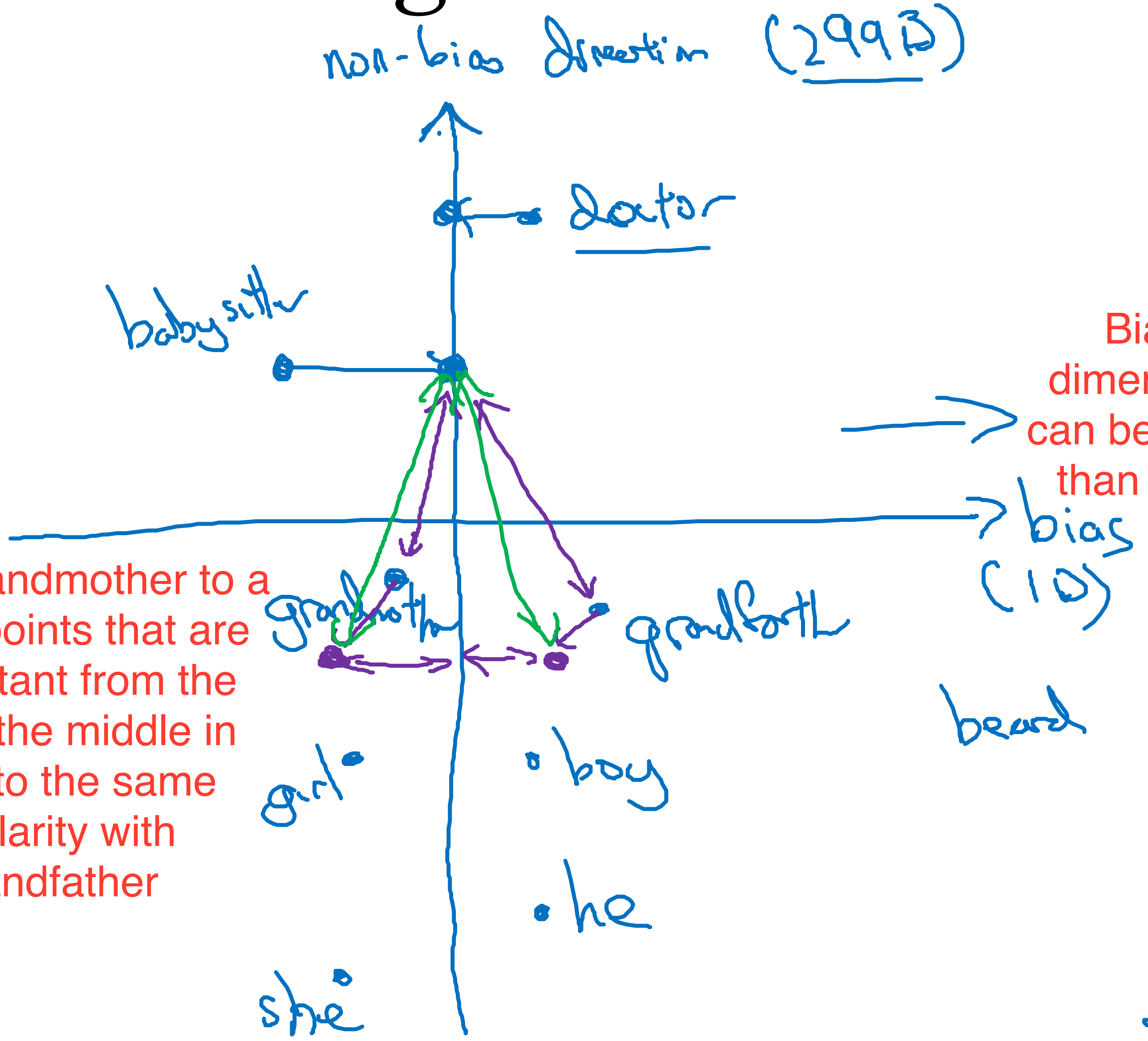
Word embeddings can reflect gender, ethnicity, age, sexual orientation, and other biases of the text used to train the model.

Bias relating to socioeconomic status

[Bolukbasi et. al., 2016. Man is to computer programmer as woman is to homemaker? Debiasing word embeddings]

Andrew Ng

Addressing bias in word embeddings



Bias dimension can be more than one.

Move grandmother to a pair of points that are equidistant from the axis in the middle in order to the same similarity with grandfather

1. Identify bias direction.

$e_{he} - e_{she}$

$e_{male} - e_{female}$

$\vdots$

average

Taking a few of these differences and average them allows you to figure out what looks the gender direction or the bias direction.

SVU: singular vlaue decomposition, which is close to PCA

2. Neutralize: For every word that is not definitional, project to get rid of bias.

3. Equalize pairs.

→ grandmother — grandfather

girl boy

[Bolukbasi et. al., 2016. Man is to computer programmer as woman is to homemaker? Debiasing word embeddings]

Andrew Ng