# Week 2

## Video 1: Word Representation

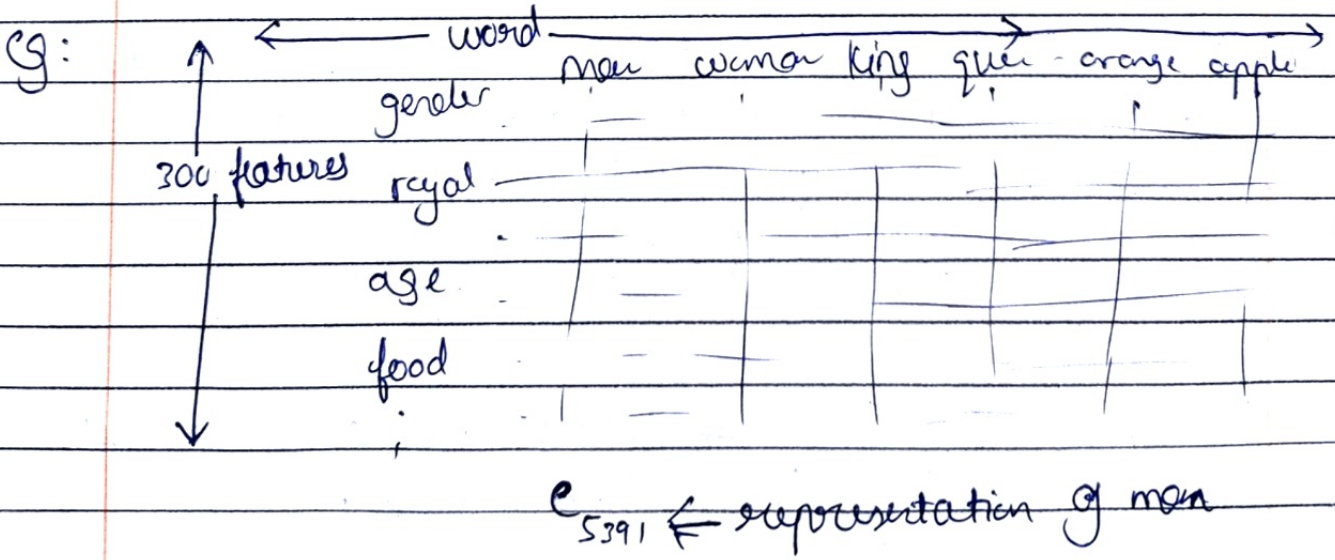
* Word embedding is a way of representing words
* We define language using vocabulary and represent our words with a one hot vector
* Weakness - words are treated as objects and can't be generalised

Example:

"I want a glass of orange ____" — predicts juice if trained

I want a glass of apple ____" — can't 100% predict juice

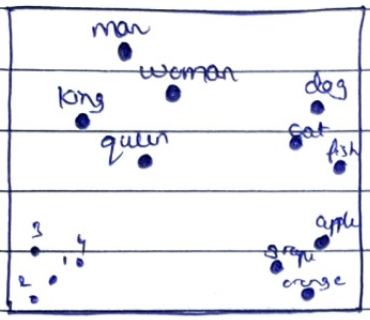
* product of one hot encoding vectors is 0.
* featured representation (word embeddings)




orange & apple share similar, hence algo can generalise between them.

DATE
PAGE

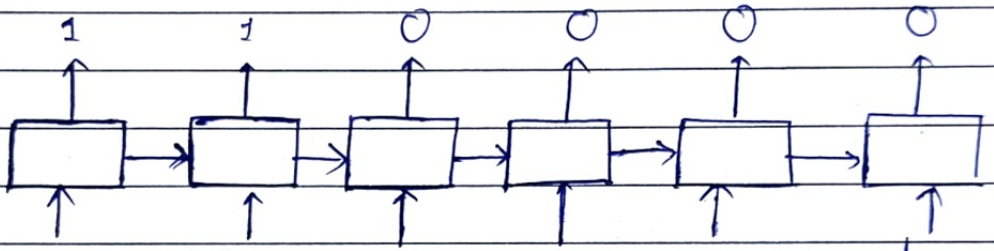
* To visualise word embeddings we use a t-SNE algorithm to reduce features to 2 dimensions




more related words are closer

Video 2: **Using Word Embeddings**

**Named Entity Recognition**




| Sally | Johnson | is | an | orange | farmer |
|---|---|---|---|---|---|
| Robert | Lin | is | an | apple | farmer |
| ~~Mahmoud Sarah~~ | ~~Badary Smith~~ | is | a | durian | ~~farmer~~ cultivator |

If the algorithm is trained on a smaller set of words it might not recognise durian as a fruit. But even if it does

durian : fruit : orange
cultivator : farmer

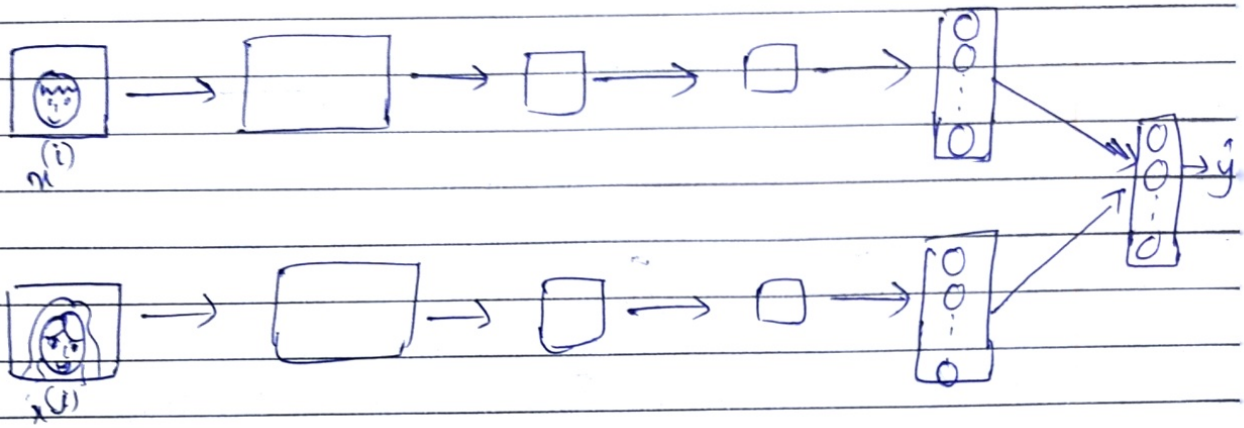
Word embeddings are ~~to~~ learnt from a large text corpus probably 100 billion words.

These are transferred to a task which requires smaller

training set

You must fretune the word embeddings with new data.

*




face → vector → check for similarity

Video 3: **Properties of Word Embeddings**

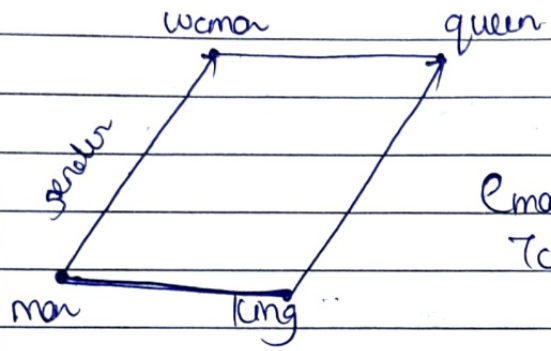
* Analogy Reasoning

| | 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 | 0.09 | 0.01 | 0.02 | 0.01 | 0.95 | 0.97 |

$e_{5391}$ $e_{man}$ $e_{woman}$

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

~~$e_{king} - e_{?}$~~ man : woman :: king : ?




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

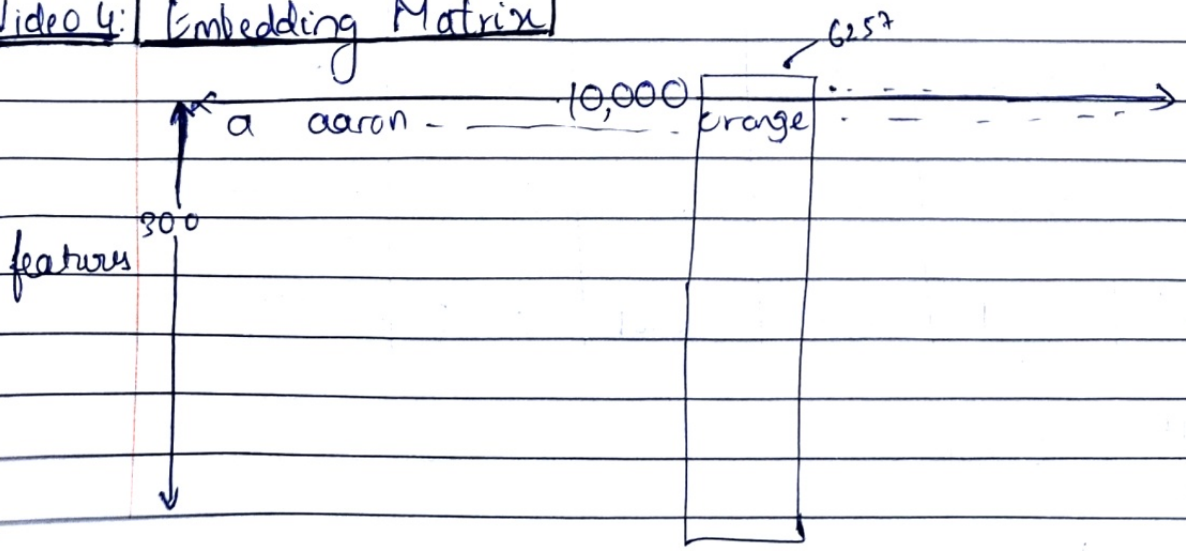
To find word $w$:

$\arg\max_{w} \; sim(e_w, e_{king} - e_{man} + e_{woman})$

↑ 30-75% accurate

$sim(u, v) = \frac{u^T v}{\|u\|_2 \|v\|_2} = \|u - v\|^2$

## Video 4: Embedding Matrix



E - Embedding matrix
$O_{6257}$ - One hot vector

$E \cdot O_{6257} = \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix} = e_{6257}$ (300,1)

## Learning Word Embeddings

Video 4:

### Neural Language Model

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

I $O_{4345} \rightarrow E \rightarrow e_{4345}$

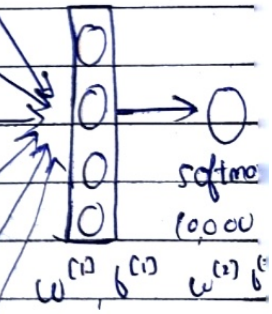
want $O_{9665} \rightarrow E \rightarrow e_{9665}$

a $O_1 \rightarrow E \rightarrow e_1$

glass $O_{3852} \rightarrow E \rightarrow e_{3852}$

of $O_{6163} \rightarrow E \rightarrow e_{6163}$

orange $O_{6257} \rightarrow E \rightarrow e_{6257}$




Context:

(i) last 4 words

4 → hyperparameter

(ii) 4 words on L & R

(iii) Last one word

(iv) nearby 1 word

i.e skip grams model

Video 5: **Word to Vec**

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

| context | target |
|---|---|
| orange | juice |
| " " | glass |
| " " | my |

given a context word, you are asked to predict a randomly chosen word within +- 5 / 10 window of input context word.

Model:

Vocab size = 10000

$x \longrightarrow y$

context c ("orange") $\longrightarrow$ target t ("juice")

6257 4834

$O_c \rightarrow E \rightarrow e_c \rightarrow O \rightarrow \hat{y}$

softmax

#PragyaNotebook

DATE
PAGE

$e_c = EO_c$

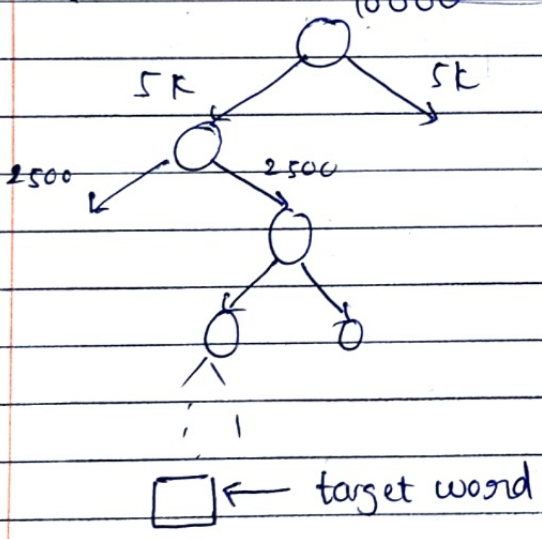
softmax : $p(t|c) = \frac{e^{\theta_t \cdot e_c}}{\sum_{j=1}^{10000} e^{\theta_j \cdot e_c}}$ $\theta_t$ = parameter associated with output

$\mathcal{L}(\hat{y}, y) = \sum_{i=1}^{10000} y_i \log \hat{y}_i$

$\hat{y}$ = possibility/probability of target output by softmax

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

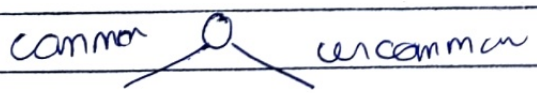
Hierarchical softmax: uses log




where does word lie? first 5K or latter 5K

If first 5K:
then in first 2500/latter 2500

so on till it reaches the leaf of the tree - word




How to sample context c?

- choose context randomly from corpus
- frequent words can dominate over uncommon words
- we don't do this normally.

## Video 6: Negative Sampling

| context (c) | word (t) | target (y) |
|---|---|---|
| orange | juice | 1 |
| orange | king | 0 |
| orange | book | 0 |
| orange | the | 0 |
| orange | of | 0 |

define a logistic regression model

The chance of $y=1$, given input c, t pair

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

$\theta_t$: parameter; $e_c$: embedding

$O_{6257} \rightarrow E \rightarrow e_{6257}$ → 10k binary logistic regression classifiers

#PragyaNotebook

DATE
PAGE

## Video 7: GloVe Word Vectors

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

C, t :

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

($X_{ij}$: $i$ = c, $j$ = t)

$X_{ij} = X_{ji}$ - count that measures how often i & j appear close to each other.

GloVe Model:

~~minimise : $(\theta_i^T e_j - \log X_{ij})^2$~~

minimise : $\sum_{i=1}^{10000} \sum_{j=1}^{10000} f(X_{ij}) (\theta_i^T e_j + b_i + b_j' - \log X_{ij})^2$

($\theta_i^T e_j + b_i + b_j' - \log X_{ij}$ → 0)

$f(X_{ij})$ : weighing term

$f(X_{ij}) = 0$ if $X_{ij} = 0$

$0 \log 0 = 0$

Weighing factor gives meaningful amount of computation even to the less freqeuent words like durian & gives more weight to words like this, i, a, of which appear too many times

$\theta_i$ & $e_j$ are symmetric

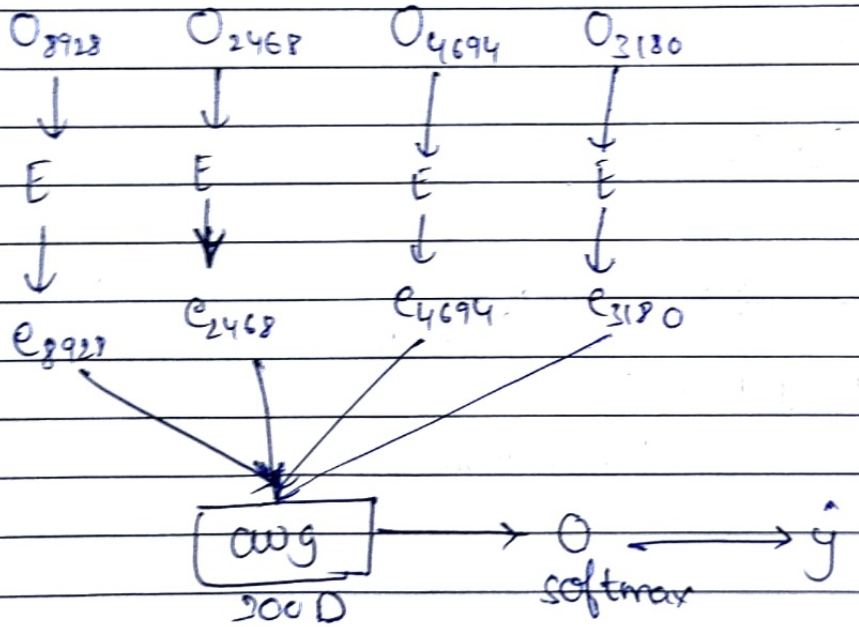
Video 8: **Sentiment Classification**

* Huge label data set may not be available
* Training sets are from 10000 to 100000.
* That's why we use word embeddings

Example

The dessert is excellent

8928 2468 4694 3180




Drawback: ignores word order.

eg. "Completely lacking in good service, good taste & good ambience."

algo will measure 3 "good" and ~~thanks~~ evaluate the review to be good

DATE
PAGE

# Video 9: Debiasing Word Embeddings

* bias here refers to undesirable bias like gender bias, ethnicity bias & soon.

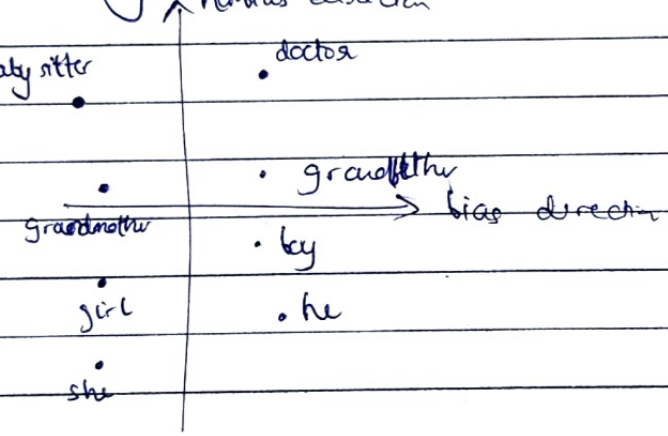
Example:

man: Computer programmer as woman: homemaker

father: doctor as mother: nurse

* biases originate from the biases in text written by people.

## Addressing bias in word embeddings




1. Identifying bias direction

$$\begin{pmatrix} e_{he} - e_{she} \\ e_{male} - e_{female} \\ \vdots \end{pmatrix}$$

avg.

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