Date ..../..../.....

# Chapter 2

## # Vector Space Models for NLP

### ⇒ 2-D Vector Space

A 2-D vector-space is defined as a set of linearly independent basis vectors with 2 axis. Each axis corresponds to a dimension in the vector space

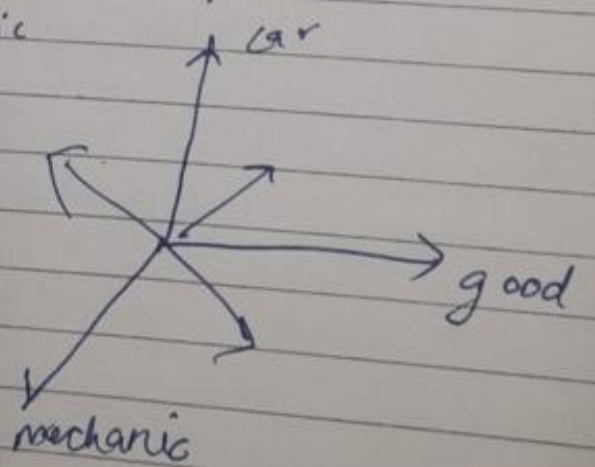
### ⇒ 3-D Vector Space

A 3-D vector-space is defined a set of linearly independent basis vector with 3 axis. Each axis corresponds to a dimension in the vector space.

Linearly independent vectors of size $N$ will result in $N$ dimensional axes which are mutually orthogonal to each other

Ex :- Binary Incidence Matrix

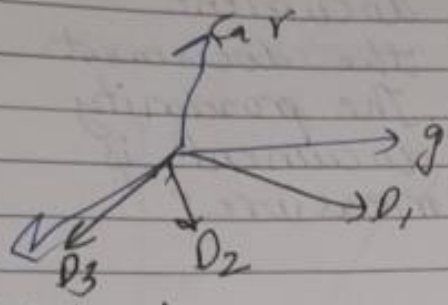
| | good | car | mechanic |
|---|---|---|---|
| $D_1$ | 1 | 1 | 1 |
| $D_2$ | 1 | 0 | 1 |
| $D_3$ | 0 | 1 | 1 |



Date ..../..../.....

Ex:- TF - IDF Incidence Matrix

| | good | car | mechanic |
|---|---|---|---|
| $D_1$ | 0.91 | 0 | 0.0011 |
| $D_2$ | 0.21 | 0 | 0.1 |
| $D_3$ | 0.15 | 0 | 0.921 |




# Document - Term Matrix

| | $d_1$ | $d_2$ | $d_3$ | $d_4$ | $d_5$ | ... | $d_{12}$ |
|---|---|---|---|---|---|---|---|
| $t_1$ | 0.1 | 0.0 | | | | | |
| $t_2$ | .1 | 0.0 | | | | | |
| $t_3$ | 0.0 | 0.9 | | | | | |
| $t_4$ | 0.0 | 0.8 | | | | | |
| ⋮ | | | | | | | |
| $t_{10}$ | 0.6 | 0.7 | | | | | |

→ The columns of the matrix represent the documents as vectors. A document vector is represented by the term present in the document

# Query Modelling

Each query is modelled as a vector using the same attribute space of documents.

$$q = [q_1 \; q_2 \; q_3 \; \ldots \; q_{tn}]$$

The relevancy ranking of a document depends on the distance of the document with respect to the query. The proximity of the query with every document is computed using distance measures

# Document similarity

Earlier using the Binary Incidence Matrix, a query returned a set of documents whether the query keywords were found in documents or absent. It did not give any ranking for the retrieved documents. A similarity measure is a real-value function that qualifies the similarity between two objects. Some of the methods are given below.

Euclidian distance = $\sqrt{q_1^2 - d_2^2}$

Cosine similarity - $\frac{d_1 \cdot d_2}{(||d_1|| \; ||d_2|| \; ||d_1|| \; ||d_2||} = \frac{\vec{d_1} \cdot \vec{d_2}}{||\vec{d_1}|| \; ||\vec{d_2}||}$

Cosine distance = $1 - \cos(\vec{d_1}, \vec{d_2})$

cluster similarity = $\frac{\vec{d_1} \cdot d_2}{||\vec{d_2}||}$

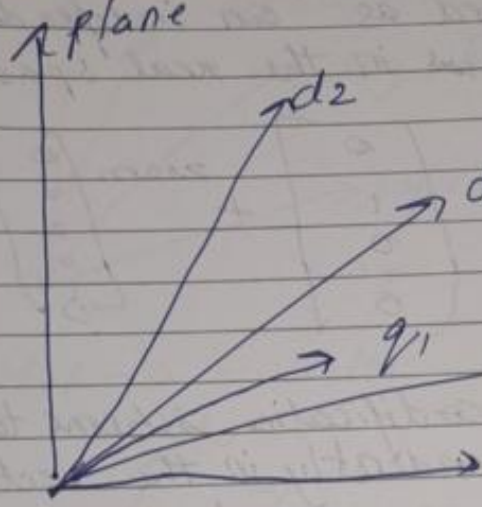
Jaccard similarity = $\left| \frac{\vec{d_1} \cap \vec{d_2}}{\vec{d_1} \cup \vec{d_2}} \right|$

Spiral

Date ..../..../.....

Euclidean measure does not work well for unequal sized vectors as the vectors are not normalized. We often use a normalized correlation coefficient, cosine distance for the similarity measure.

cosine distance $= 1 - \cos(\vec{d_1} \cdot \vec{d_2})$




$$\cos\theta_1 = \frac{\vec{q_1} \cdot \vec{d_3}}{|q_1||d_3|}$$

$$\cos\theta_2 = \frac{\vec{q_1} \cdot \vec{d_2}}{|q_1||d_2|}$$

$$\cos\theta_3 = \frac{\vec{q_1} \cdot \vec{d_3}}{|q_1||d_3|}$$

# Proximity Score

A query is considered as a document vector. The proximity of the query with every document is computed using a distance measure. Cosine distance is preferred and it is easy to compute if the document vector distances are normalized. Proximity score (angle) will be considered as relevant and retrieved

# Vector Representation of words

Let V be the unique terms and |V| be the size of the vocabulary. Then every vector representing the word $\mathbb{R}^{|V| \times 1}$ would point to a vector in the V dimensional space

# One hot Vector

We can represent each word as an independent vector quantity as follows in the real space $\mathbb{R}^{|V| \times 1}$

$$t^{a} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \end{bmatrix} \quad t^{ab} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad t^{zoom} \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ \vdots \\ 1 \end{bmatrix}$$

This is a very simple codification scheme to represent words independently in the vector space. This is known as one hot vector.

# one Hot Vector - 2

In one-hot vector, every word is represented independently. The terms, home, house, flats, apartments are independently coded. With one-hot vector based model. the dot product

$$(t^{House})^{T} \cdot t^{apartment} = 0$$

With one-hot vector, there is no notion of similarity or ~~Synonyms~~ Synonyms.

Spiral

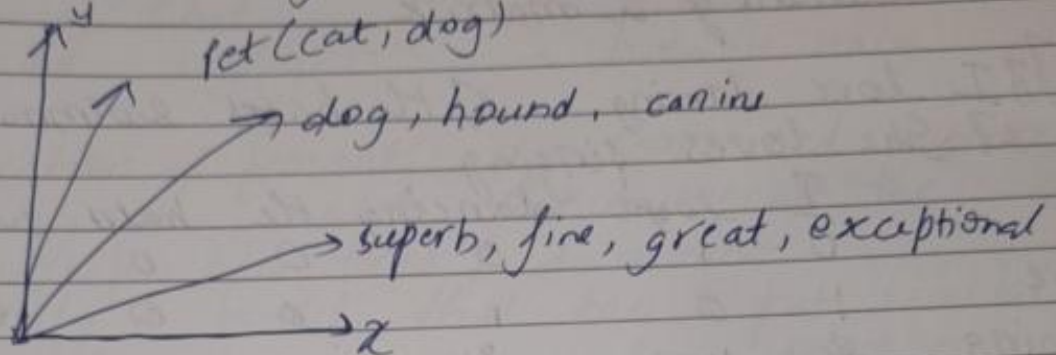
Date ..../..../.....

Goal of word vector :-

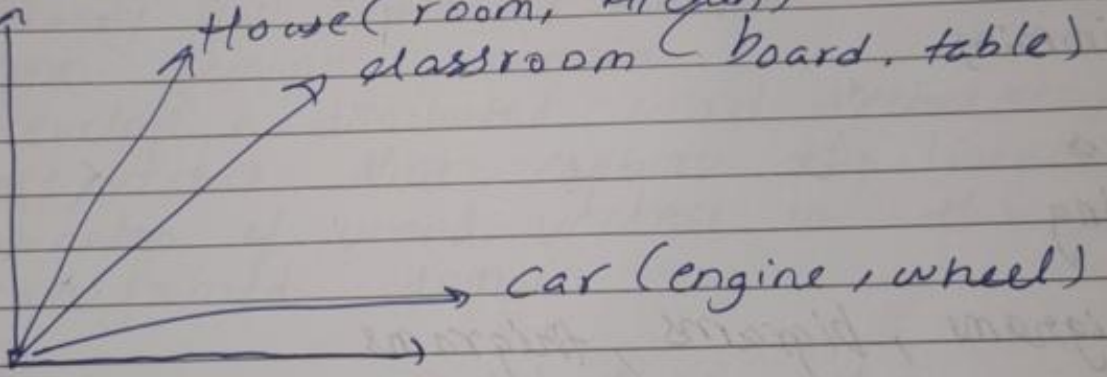
→ Reduce word - vector space into a smaller sub-space

→ encode the relationship among words.

→ Relationship among terms - synonyms




→ Has-a-vector - composition




## Contextual understanding of text

→ In order to understand the word and its meaning, its not enough if we consider only the individual word

→ The meaning & context should be central in understanding word / text

→ exploit the context department of words

→ co-occurance of words

Spiral

# Co-occurence Matrix

Date ..../..../.....

A co-occurence is a combination of terms that are likely to be used in the same context. A co-occurence matrix stores co-occurences of words. The count of a pair of words that appears in a content window is represented as an element of a matrix

[x] 1] I love dancing 2] He hates swimming 3] She loves singing

| | I | love | dancing | He | hates | swim... |
|---|---|---|---|---|---|---|
| I | 0 | 1 | 0 | 0 | 0 | 0 |
| love | 1 | 0 | 1 | 0 | 0 | 0 |
| dancing | 0 | 1 | 0 | . | . | . |
| He | . | . | . | | | |
| hates | . | . | . | | | |
| swimming | . | . | . | | | |
| she | . | . | . | | | |
| loves | . | . | . | | | |
| singing | . | . | . | | | |

# Unigrams, bigrams, trigrams

→ A sequence of two words is called a bigram
→ A three-word sequence is called trigram
→ N-gram means a sequence of words of length n.

Spiral

Date ..../..../.....

## Collocations

Collocations is a juxtaposition of two or more words the more often occur together than by chance

- powerful computer → major problem
- brief chat → pitch dark
- broad daylight → heavy rain

## Creation of semantically connected vectors

- Identify a model that enumerate the relationship b/w terms & documents
- Identify a model that tries to put similar items closer to each other in some space or structure
- A model that discovers/uncovers the semantic similarity b/w words & documents in the latent semantic domain
- Develop a distributed word vectors or dense vectors that captures the linear combination of word vectors in the transformed domain.

## Methods to create dense vector

- Latent semantic analysis or latent semantic Indexing
- Neural networks using skip grams & CBOW
- Skip grams use center of words to predict the surrounding words.
- Brow Clustering - statistical algorithms for assigning words to class based on the frequency of their co-occurence with other words.

Spiral

# Why dense Vectors?

→ sparse vectors are too long & ↓ convenient as features in machine learning.
→ abstracts more than just frequency count.
→ It captures neighborhood words that are connected as synonyms

# Singular Value Decomposition

Singular value decompisition is a method to factorize a rectangular / square matrix into three matrices.

$$A = U \Sigma V^T$$

$M \times N$ = $M \times K$ (left singular vectors) $K \times K$ (singular vector) $K \times N$ (right singular matrix)

Row vectors of U are called as the left-singular vector
Rows of $V^T$ form an orthonormal set
$\Sigma$ is diagonal matrix and its values are arranged in descending order

# Singular Values

→ It is a diagonal matrix
→ Singular value are arranged in descending order
→ Singular values reflect the major associative patterns in the data, & ignore the smaller less important influences
→ Highst order dimension captures the most variance in the original dataset or most of the information related to term document matrix.

Spiral

Date ..../..../.....

→ The next higher dimension captures the next higher variance in the original data set
→ arranged in descending order

# Dimensionality Reduction
→ SVD is better suited for measuring the similarity b/w documents by exploiting the similarity patterns that exist in the word co-occurance.
→ The co-occuring terms are mapped in the same dimension thereby reducing the dimension.
→ Increases the similarity Matrix A in the m-dimensional space & transforms it as $\hat{A}$ in the reduced dimensional space
$k \leq m$

$\Delta = \| A - \hat{A} \|_2$ Should be as less as possible

where $\|.\|$ is the $L_2$ norm for the matrices.

Similar documents will be brought to one hyper plane.
Similar words will be brought to same axis.

# Important equation in SVD
Since U & V are orthonormal matrices

$U^T U = V^T V = I$

$A^T A = V \Sigma^T U^T U \Sigma V^T = V \Sigma^2 V^T$

$A A^T = U \Sigma V^T V \Sigma^T U^T = U \Sigma^2 U^T$

$U^T A = U^T U \Sigma D^T = \Sigma V^T$

$U_q = U_q^T U \Sigma^{-1}$

Spiral

Date ..../..../.....

$$\Sigma_3^{-1} = \begin{bmatrix} \frac{1}{\sigma_{11}} & 0 & 0 \\ 0 & \frac{1}{\sigma_{22}} & 0 \\ 0 & 0 & \frac{1}{\sigma_{33}} \end{bmatrix}$$

$$q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \rightarrow u_q = u_q^T U \Sigma^{-1}$$

$$\cos\theta \quad \frac{u_q \cdot d_i}{\|u_q\| \; \|d_i\|}$$