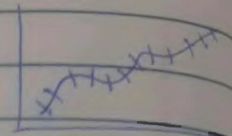
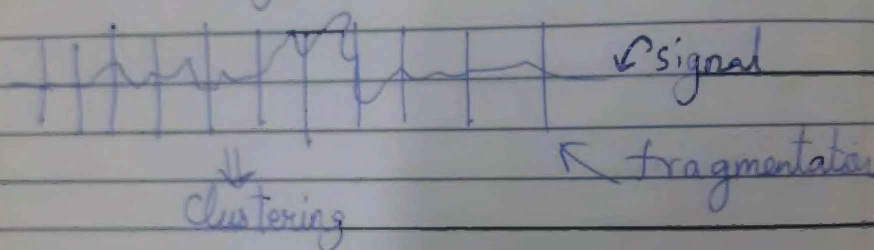
→ Kew words

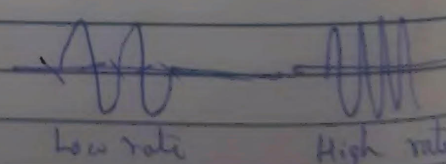
- STFT - Short-term Fourier Transform
- Piece wise linear Transformation



(plication NLP) Speaker - Diarization
input: multiple voice signal
output: transcript
solution: framing → frame by frame feature comparison



- Sampling
- zero-crossing rate



- MFCC - Mel frequency Cepstral Coefficients
- letters → sentences → paragraphs → documents → corpus

* NLTK - natural language toolkit (Python library)

PAGE NO.
DATE :

→ Processing Text Data.

D1 I read a book daily

D2 I booked ticket

D3 Read and write daily

(1) Tokenization – word level or character level
(2) Stop word removal – domain specific

(3) Stemming – transform word to root form

eg: crying, cries, cried } cry — remove suffix.

machines → machine

Note: butterflies → butterflie
solution: Lemmatization : preserves meaning

(4) Lower case conversion.
(5) Extract vocabulary : list of terms.

V = {book, daily, read, ticket, write}

Note : Not necessarily in alphabetic order.

→ Document-term matrix

	book	daily	read	ticket	write
D_1	1	1	1	0	0
D_2	1	0	0	1	0
D_3	0	1	1	0	1

Also known as boolean representation of corpus. or bag-of-words (BOW)

*→ TF = Term-frequency representation
TFIDF = alternative to doc-term matrix

> Problems with document-term matrix

· Assume 1 billion docs x 50 billion terms

↓

very large matrix

· And sparse matrix - most values are zeros.

Drawbacks - Frequency of word count
- order of words

PAGE NO 1
DATE. Lec 3

→ Inverted Indexing Term - Frequency Representation

D_1		D_2		Relevance
t_1 -	180	t_1 -	134	
t_2 -	180	t_2 -	5	
t_3 -	5	t_3 -	8	

t_1 A term is present in most of the documents in the corpus.

t_2 A term is present in a single document several times and rarely found in others.

t_3 A term is present in very few documents of the corpus.

Drawbacks - TF cannot seperate between t_2 and t_1.

→ TF - IDF approach.

DF = document frequency.
Number of documents in which this term present

IDF (t_1) = Kind of weight - such that its captures difference of t_1, t_2

$$= \log\left(\frac{N}{df_i}\right)$$

The intution behind this is that if a term appears in a large number of documents in a corpus it is probably not important or discriminative.

PAGE NO.
DATE :

▷ Vector-space model. (TF-IDF)

	book	daily	read	ticket	write
D_1	$\log \frac{3}{2}$	$\log \frac{3}{2}$	$\log \frac{3}{2}$	0	0
D_2	$\log \frac{3}{2}$	0	0	$\log 3$	0
D_3	0	$\log \frac{3}{2}$	$\log \frac{3}{2}$	0	$\log 3$

Consider $D_1, D_2 \dots D_n$ as vectors of dimension 5.

This allows us to perform vector operation on these and perform document classification, document similarity, document clustering etc.

▷ There are more advanced representations. word2Vec, doc2Vec, Wav2vec, GLoVe .....

* App of the week - SCIGen.

/10/2023 Wed

PAGE NO.
DATE: Lec 4

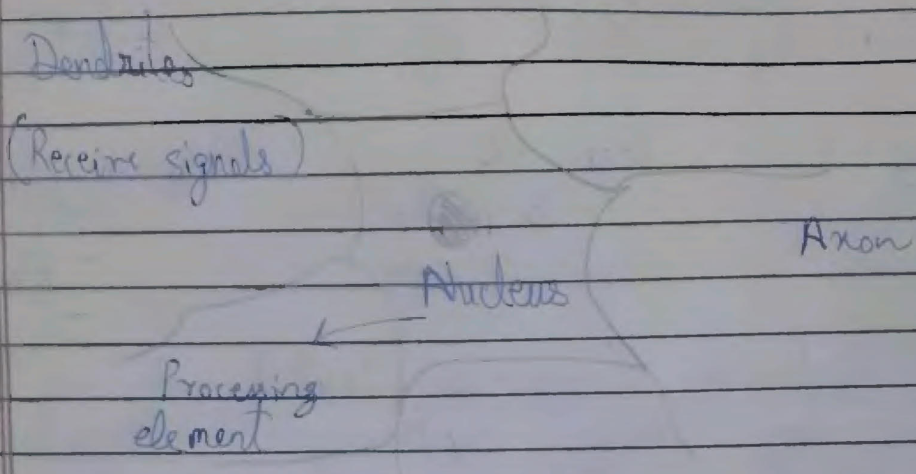
Innovative Assignment (30 marks)

- → Research article / Review Paper
- → Implemention of GUI based project.

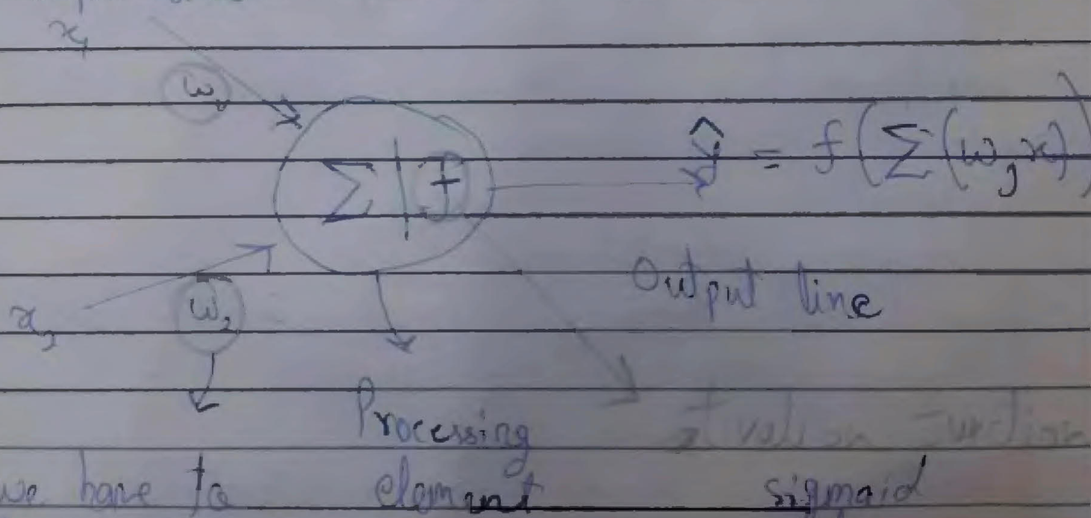
Trending Areas	Team of 1 - 3 people	Prepare Report
		- Survey - Existing sol^n^ - Implementati

- Image Captioning (image to text)
- Reverse of image captioning
- Text - based question - answering system
- Visual question - answer system.
- Text - to - speech synthesis
- Speech - to - text (speech recognition)
- Document classification
- Information retrieval from PDF / Documents
- Voice Biometrics.
- Speaker Diarization
- Singer Identification
- Sentiment Analysis from text data
- Music composer recognition
- ChatBot / Voice Assistant.
- Text or Audio Summerization
- Generating speech or music (GAN)
- Intent detection from audio.
- Speech data collection and some experiments.
- Bird species classification (BirdClef)
- Text detection and conversion to speech.
- Speech corpus development for gujarati language
- Voice transformation.
- Audio signal classification.
- Tagging Audio data
- Talking Calculator

Artificial Neural Networks



Biological Neuron



we have to learn these parameters

sigmoid
relu
linear
tanh
elu
hardlimit

$w_1' = w_1 + \Delta w_1$

PAGE NO. 1
DATE: Dec 6

01/02/2023 Wednesday

Keywords -

Vanishing Gradient Problem,

MNIST Dataset - 70,000 images of B&W digits
10 classes

Flattening 28×28 7000 image /class

Word Embeddings

App of the week -

03/02/2023 Wednesday

Text Representation.

Text - Bag of words.

- One hot encoding

labels = {book, daily, read, ticket, write}

$$= \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \right\}$$

Problem = sparsity, meaning

- Embeddings

PAGE NO.
DATE :

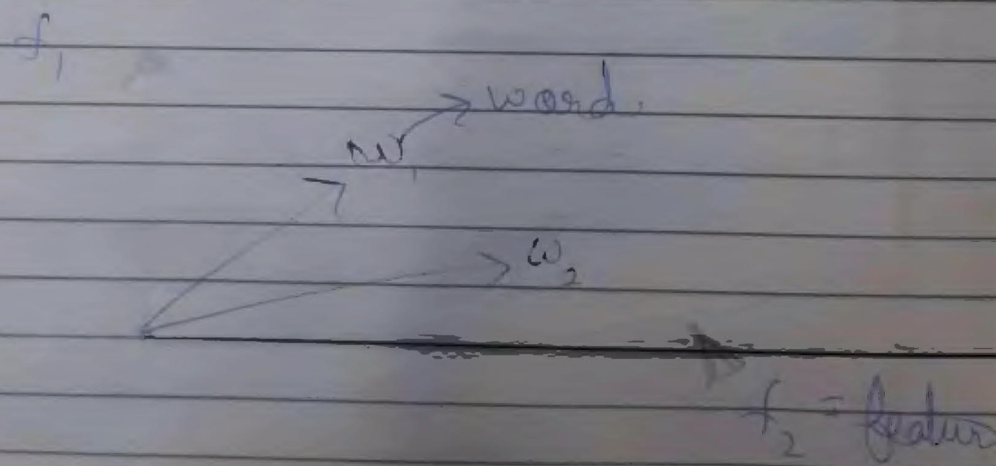
Dimentionality reduction
Attribute Subset selection / feature ranking

Principle Component analysis
linear Discriminent analysis → Meth
Auto encoder.
Singular value dicomposition.
→ Ref 24/02/2023

→ Embeddings

Embeddings aim to represent the word in a dense vetor form.

Making sure that similar words are close to eachother in the embedding space.



Latent semantic Indexing
↑ hidden Implicit
↑ Capture meaning context

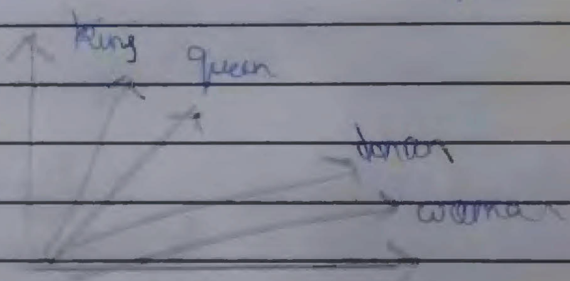
PAGE NO.
DATE :

- Topic modeling - similar words come closer in embedding space.

- Cosine similarity is prefered as compared to euclidean distance.

- Vector / arithmetic operations on words -

  King - man + woman = queen



(i) Word 2 Vec - Word to vector
(ii) GLoVe - global vectors

Ref
Q # 29/03/2023

→ Language models

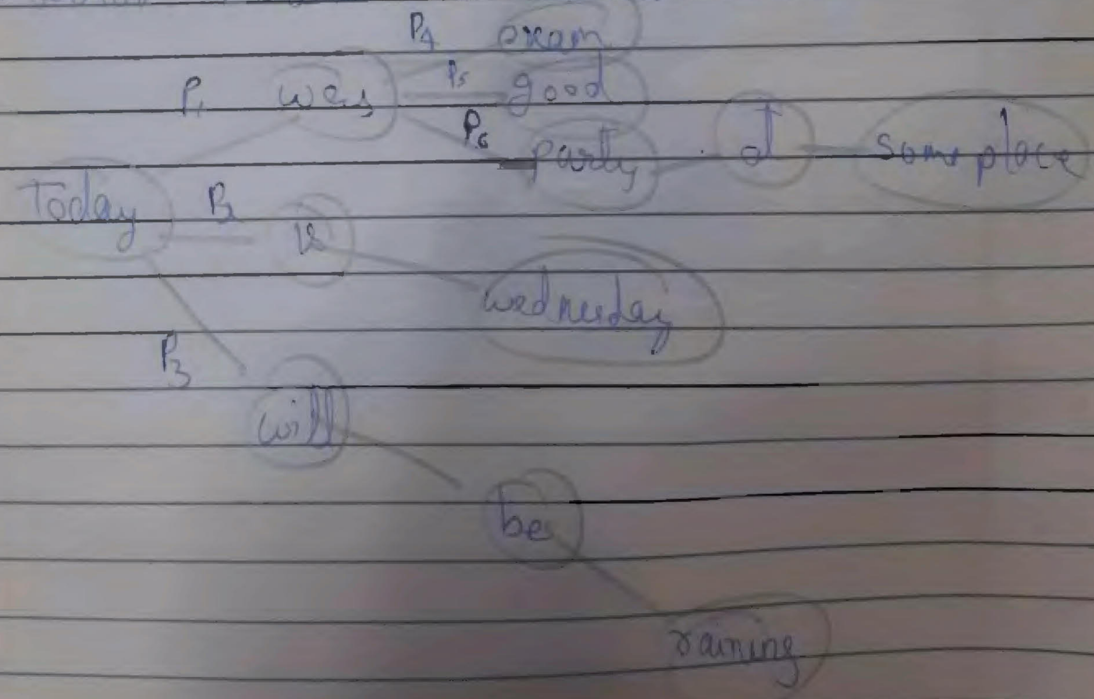
Unigrams = { book , read , ticket .... }

Bigrams = { book daily , daily read , ... }

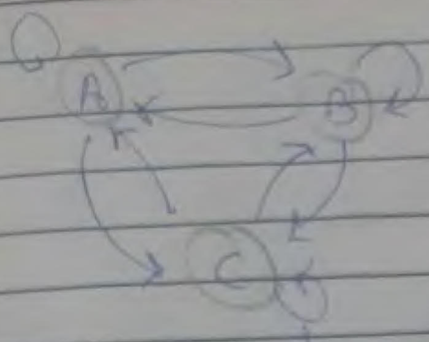
Order is important.

In general N-gram is a representation of language model.

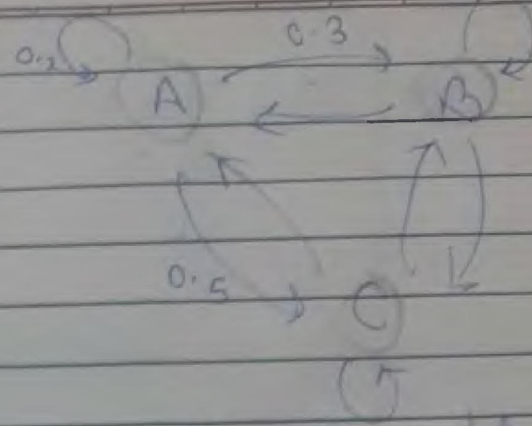
→ Markov models / Markov chains



* Markov models are state-transition diagrams with probabilities.



State transition diagram.



Markov model.

If there are n states, maximum n^2 transition possible.

Summation of probability for each outgoing transition is equal to 1.

Can be represented as $N \times N$ matrix

Diagonals represent probability self loops.

Known as state transition probability matrix.

Deterministic state

$$\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Non deterministic State

$$\begin{bmatrix} 0.2 \\ 0.6 \\ 0.5 \end{bmatrix}$$

$$P = \begin{bmatrix} 0.5 & 0.1 & 0.4 \\ 0.1 & 0.7 & 0.2 \\ 0.3 & 0.3 & 0.4 \end{bmatrix}$$

10/02/2023 Friday

PAGE NO.
DATE: Lec 7

$$T = \begin{array}{c|ccc} & A & B & C \\ \hline A & 0.5 & 0.1 & 0.4 \\ B & 0.1 & 0.7 & 0.2 \\ C & 0.1 & 0.3 & 0.4 \end{array}_{3\times 3}$$

$$P_0 = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}_{1\times 3}$$

$$P_1 = \begin{bmatrix} 0.1 & 0.7 & 0.2 \end{bmatrix}_{1\times 3}$$

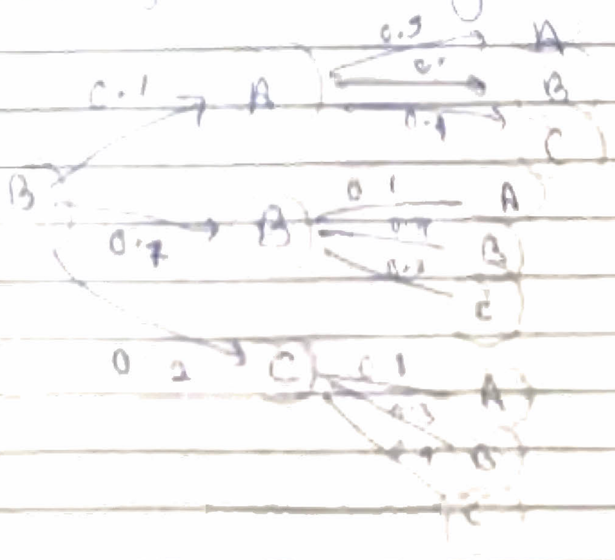
$$P_2 = \begin{bmatrix} 0.18 & 0.56 & 0.26 \end{bmatrix}_{1\times 3}$$

$$P(B\_A) = P(BAA) + P(BBA) + P(BCA)$$

$$P_3 = P_2 \cdot T = \begin{bmatrix} 0.224, & 0.488 & 0.288 \end{bmatrix}$$

$$P_n = P_{n-1} \cdot T = P_0 T^n$$

Probability Tree Diagram



$P_0 \quad P_1 \quad P_2 \quad \ldots$

PAGE NO.
DATE :

Q) Calculate probability of following state sequence

A C B A C B =

Q2) Calculate probability of achieving state C assuming you start with same state

P(C -- C)

A1) P_0 A = [1 0 0]

P_1 AC = [0.3 0.3 0.4]

P_2 ACB = [0.34 0.34 0.26]

P_3 ACBA = [0.262 0.346 0.292]

P_4 ACBAC = [0.2532 0.356 0.2908]

P_5 ACBACB = [0.25 0.35 0.3]

A1) A → C → B → A → C → B

= 0.00144

P_4 = [0.2472 0.4504 0.3024]

P_5 = [0.25936.

0.5 0.1 0.4
0.1 0.7 0.2
0.3 0.3 0.4

A 2>

$$P(C \_ \_ C) = P_0 = [0 \quad 0 \quad 1]$$

$$P_1 = [0.3 \quad 0.3 \quad 0.4]$$

$$P_2 = [0.3 \quad 0.31 \quad 0.34]$$

$$P_3 = [0.288 \quad 0.384 \quad 0.328]$$

— α — α — α — α —

$$T = \begin{array}{c} \\ A \\ B \\ C \end{array} \begin{array}{c} \begin{array}{ccc} A & B & C \end{array} \\ \begin{bmatrix} 0.5 & 0.1 & 0.4 \\ 0.1 & 0.7 & 0.2 \\ 0.3 & 0.3 & 0.4 \end{bmatrix} \end{array}$$

$$P_0 = [0 \quad 1 \quad 0]$$

$$P_1 = [0.1 \quad 0.7 \quad 0.2]$$

$$P_2 = [0.18 \quad 0.56 \quad 0.26]$$

$$P_3 = [0.224 \quad 0.488 \quad 0.288]$$

$$P_4 = [0.2472 \quad 0.4504 \quad 0.3024]$$

$$P_5 = [0.25936 \quad 0.43072 \quad 0.30992]$$

$$P_6 = [0.265728 \quad 0.420416 \quad 0.313856]$$

$$P_7 = [0.2690624 \quad 0.4150208 \quad 0.3159168]$$

$$P_8 = [0.27080832 \quad 0.41219584 \quad 0.31699584$$

$$P_9 = [0.271722496 \quad 0.410716672 \quad 0.31756081$$

PAGE NO.
DATE : 12c8

05/02/2022 Wednesday

2) A markov chain is used to model the status of equipment and having following possible states.

idle and awaiting work — I
working on a task — W
broken — B
in repair. — R

The machine is monitered at regular intervals to determine its status,

Assume that status is monitored every hour.

		I	W	B	R
T =	I	0.05	0.93	0.02	0
	W	0.10	0.86	0.04	0
	B	0	0	0.8	0.2
	R	0.5	0.1	0	0.4

(i) Prob that machine is working after 1 hour if it is idle now.

$P(W|I) = P(IW) = 0.93$

(ii) $P(I|W) = 0.1$

(iii) $P(B-R) = 0.24$

$P_0 = [0 \quad 0 \quad 1 \quad 0]$
$P_1 = [0 \quad 0 \quad 0.8 \quad 0.2]$
$P_2 = [0.1 \quad 0.02 \quad 0.64 \quad 0.24]$

(iv) In next five hours, states is

IWNIB

$= (0.93)(0.86)(0.1)(0.02)$

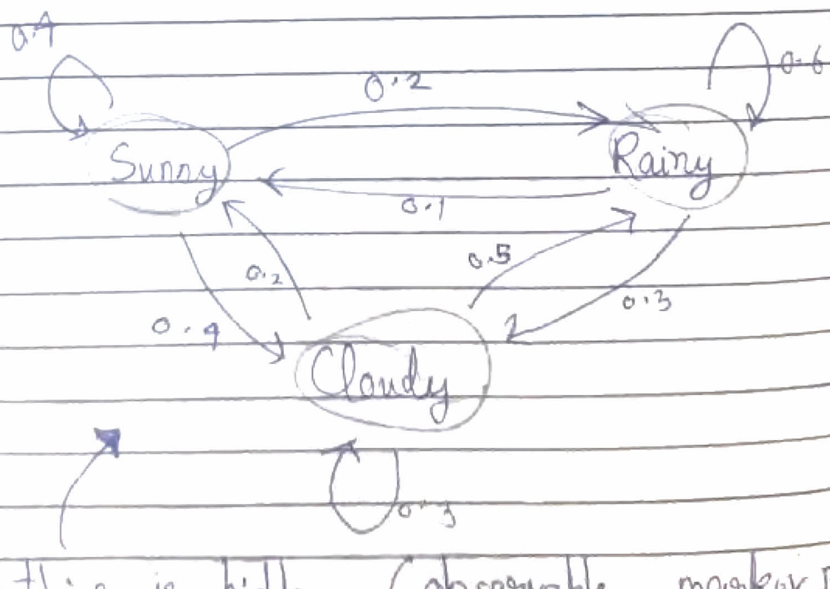
$= 0.0015$

→ Hidden Markov Models. (HMM)

Applications – Next word prediction.

- Any data which has sequence of states can be modeled as HMM.

→ Weather Example.



Nothing is hidden (observable markov models)

- In hidden markov models – state is not given. Only probability is available

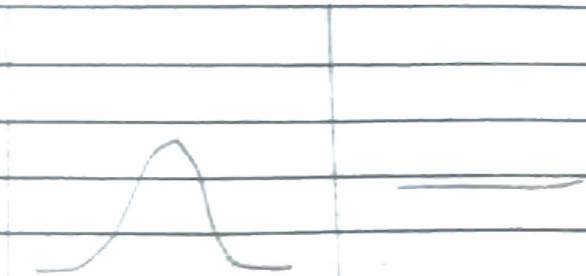
- Example : Tomorrow is cloudy – 0·3
  ↳ what is current state ?

- Related to Bayes's Theorem / Probability distribu
  ↓
  Conditional probability

- Posterior = $\frac{\text{likelihood} \times \text{prior}}{\text{evidence}}$

- $P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$

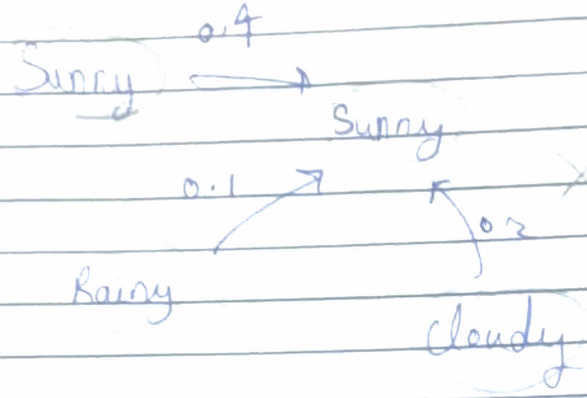
- Types of distribution.



Gaussian uniform.

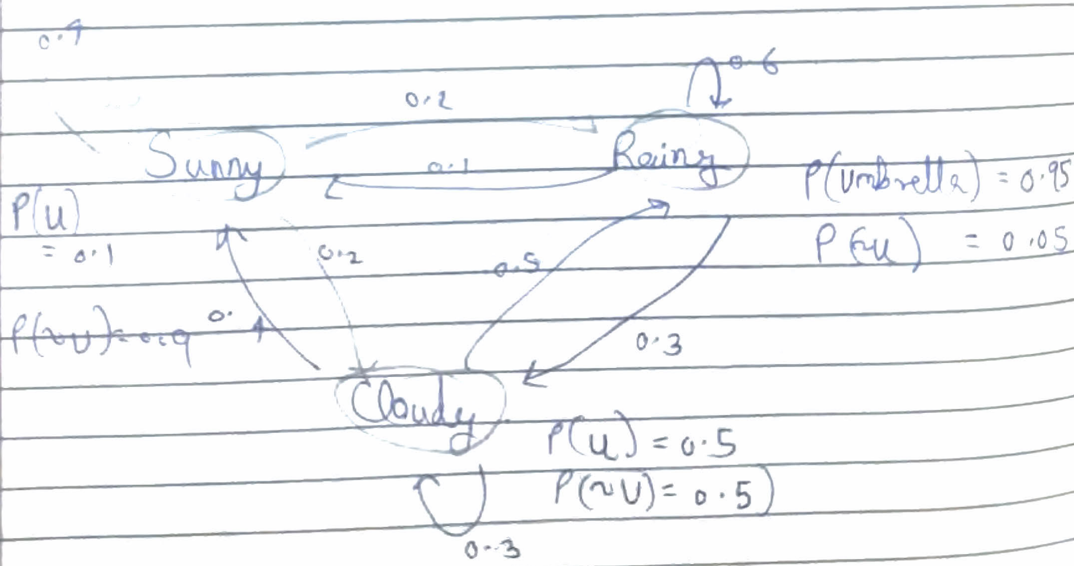
Types of ML models

- → Discriminative
- → Generative ← HMM is useful he



Emmission Probability for cloudy

- HMM : If today is sunny, which state could have resulted to today



PAGE NO. 1
DATE: Dec 9

19/02/2022 Friday

(Q) HMM has 3 parameter.

B (i) Emmision probability matrix (can be more than 1)

π (ii) Initial probability vector.

A (iii) Transition probability matrix.

$$\lambda = \{A, B, \pi\}$$

19/02/2022 Friday

Hidden Markov Models (Continued)

- State sequence notation.

$q_1, q_2 \ldots\ldots$ where $q_i \in \{$Sunny, Rainy, Cloudy$\}$

- In order to compute prob. for tomorrow's state we can use markov property.

- $$P(q_1 \ldots q_n) = \prod_{i=1}^{n} P(q_i | q_{i-1})$$

Not dependent on all previous state

But only on imediate previous state.

$P(q_3 | q_2, q_1) = P(q_3 | q_2)$

PAGE NO.
DATE :

- It is sthochastic model which models temporal or sequential data.
- A way to model dependencies of current information with previous information.
- Sthochastic = Non deterministic / Probabilistic

Example :

- O_i is observation on ith day
- We have observation sequence $O_1 O_2 \cdots O_t$
- we need to find probability that corresponding state sequence is $q_1 q_2 \cdots q_t$

$$\therefore P(q_1 q_2 \ldots q_t \mid O_1 O_2 \ldots O_t) = \frac{P(O_1 O_2 \ldots O_t \mid q_1 q_2 \ldots q_3) P(q_1 q_2 \ldots q_t)}{P(O_1 O_2 O \ldots O_t)}$$

$$\text{i.e.}\quad P(q_i \mid O_i) = \frac{P(O_i \mid q_i)\, P(q_i)}{P(O_i)}$$

$$P(q_{1:t} \mid O_{1:t}) \propto \prod_{i=1}^{t} P(O_i \mid q_i) \prod_{i=1}^{t} P(q_i \mid q_{i-1})$$

HMM Parameters

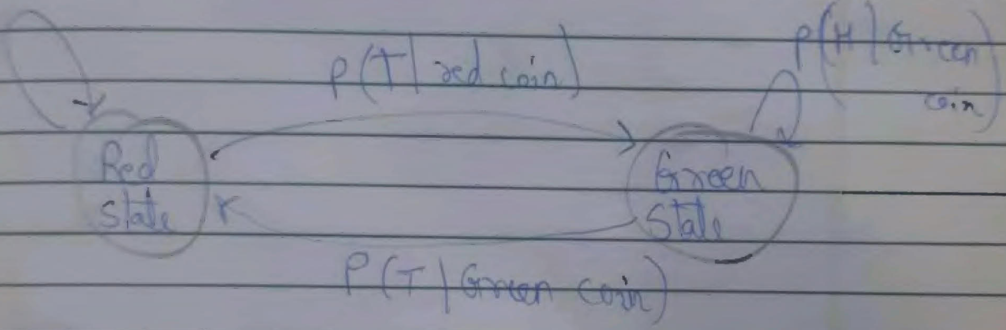
Transition Probabilities

- Red Coin
  - $P(H) = 0.9$
  - $P(T) = 0.1$
- Green Coin
  - $P(H) = 0.95$
  - $P(T) = 0.05$

Emissions

- Red dice
  - $\{1, 2, 3, 4, 5, 6\}$
- Green Dice
  - $\{1, 1, 1, 1, 1, 1, 2, 3, 4, 5, 6\}$

- $P(H \mid \text{Red coin})$



- $A = \begin{bmatrix} 0.9 & 0.1 \\ 0.05 & 0.95 \end{bmatrix}_{2\times2}$

- W = max no. of emissions
  $W = 6$

- $\pi = \begin{bmatrix} 1 \\ 0 \end{bmatrix}_{2\times1}$

- N = no. of state
  $N = 2$

- $B = \begin{bmatrix} 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 7/12 & 1/12 & 1/12 & 1/12 & 1/12 & 1/12 \end{bmatrix}_{2\times6}$

- In realword, we cannot determine exact value of parameters.
- Hence, we need to estimate the parameters.
- Parameter Estimation
  - Viterbi algorithm { " }
  - Forward algorithm (Ref # 15/03/2023)
  - Baum - Welch Algorithm - (")

$$\vec{U} = \begin{array}{cccc} \vec{U}_{c_1} & \vec{U}_{c_2} & \vec{U}_{c_3} & \vec{U}_{c_4} \\ \end{array}$$
$$\begin{bmatrix} 2 & 1 & 0 & 7 \\ 3 & 4 & 5 & 4 \\ 2 & 8 & -3 & 6 \end{bmatrix}_{3\times 4}$$

- $\|\vec{U}_{c_1}\| = \sqrt{2^2+3^2+2^2} = \sqrt{17}$

→ The Euclidean norm of vector
2nd Norm

Orthogonal matrix → Each column vector is perpendicular to each other

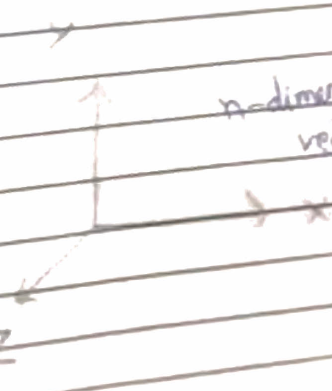
$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

unit vector

- Normalization $[2, 4, 1] \longrightarrow \left[\frac{2}{\sqrt{21}}, \frac{4}{\sqrt{21}}, \frac{1}{\sqrt{21}}\right]$

- Ortho Normalization
(ortho + normal)

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \Longrightarrow$$



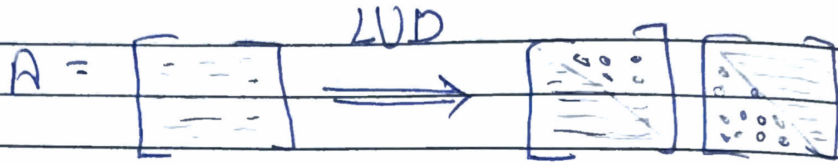
n-dimensional vector space

→ Matrix factorization.

24 → 1, 2, 3, 4, 6, 8, 12, 24

★ LU Decomposition

- (Lower - upper) Decomposition.



Lower Upper
triangular matrix

Ref #08/02/2023

★★★ Singular value decomposition,

- $A = \underline{U}\, S\, \underline{V}^T$

S → Diagonal matrix (Square)

U, V → Orthonormal matrix

- Used for Dimensionality reduction
- Used for latent semantic indexing

$A = U \; S \; V^T$

A: 500 × 4000 (documents × vocabulary)

U: 500 × 1000 (Feature matrix)

S: 1000 × 1000

V^T: 4000 × 1000

hyperparameter

PAGE NO.
DATE:

K = 1000, 100% .... ↑ ↓, reconstruct A,

- find score, rank of SVD.

- $S = \begin{bmatrix} \sigma_1 & 0 & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 \\ 0 & 0 & \sigma_{..} & \\ 0 & 0 & 0 & \sigma_n \end{bmatrix}$ ← most relevant

1000×1000

least relevant factor

singular values.

- Lower rank approximation.

- Eigan value (prerequisite)

- $A\vec{v} = \lambda \vec{v}$

Eigan value — Eigen vector

- $\begin{bmatrix} 1 & 11 \\ 11 & 1 \end{bmatrix}$ $\begin{bmatrix} 11 & 1 \\ 1 & 11 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} \lambda v_1 \\ \lambda v_2 \end{bmatrix}$

- $11v_1 + v_2 = \lambda v_1$ — $(11-\lambda) v_1 + v_2 = 0$

$v_1 + 11v_2 = \lambda v_2$ — $v_1 + (11-\lambda) v_2 = 0$

- $121 - 22\lambda + \lambda^2 - 1 = 0 \Leftarrow \begin{vmatrix} 11-\lambda & 1 \\ 1 & 11-\lambda \end{vmatrix} = 0$

$\lambda = 10, 12$

PAGE NO.
DATE :

- For $\lambda = 12$,

$$(11-\lambda)\,v_1 + v_2 = 0 \quad \Big| \quad v_1 - v_2 = 0 \Big\}$$
$$-v_1 + v_2 = 0$$

$\boxed{v_1 = v_2}$ Infinite $[1,1]$, $[2,2]$ ...

- For $\lambda = 10$,

$$v_1 + v_2 = 0 \quad \Big| \quad v_1 + v_2 = 0 \Big\}$$

$\boxed{v_1 = -v_2}$ ∞ $[1,-1]$ $[2,-2]$ .....

- Process of SVD

- The columns of $\vec{U}$ are orthonormal Eigenvectors of

$$A \cdot A^T$$

- Columns of $\vec{V}$ are orthonormal Eigenvectors of

$$A^T \cdot A$$

- S is a diagonal matrix which contains square roots of Eigen values from U or V in descending order.

01/03/2023
Wednesday

PAGE NO. 1
DATE: lec 11

Q)
$$A = \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix}_{2\times 3}$$

$$A^T = \begin{bmatrix} 3 & -1 \\ 1 & 3 \\ 1 & 1 \end{bmatrix}_{3\times 2}$$

$$A \cdot A^T = \begin{bmatrix} 9+1+1 & -3+3+1 \\ -3+3+1 & 1+9+1 \end{bmatrix}_{2\times 2} = \begin{bmatrix} 11 & 1 \\ 1 & 11 \end{bmatrix}_{2\times 2}$$

U > $\lambda = 10, 12$ ~~12, 10~~ 12, 10

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \xrightarrow{\text{norm}} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}$$

$$A^T \cdot A = \begin{bmatrix} 9+1 & 3-3 & 3-1 \\ 3-3 & 1+9 & 1+3 \\ 3-1 & 1+3 & 1+1 \end{bmatrix}_{3\times 3}$$

$$= \begin{bmatrix} 10 & 0 & 2 \\ 0 & 10 & 4 \\ 2 & 4 & 2 \end{bmatrix}_{3\times 3}$$

$10V_1 + 2V_3 = \lambda V_1$
$10V_2 + 4V_3 = \lambda V_2$
$2V_1 + 4V_2 + 2V_3 = \lambda V_3$

V > $\lambda = 12, 10, 0$

$$\begin{bmatrix} 1 & -2 & 1 \\ 2 & 1 & 2 \\ 1 & -2 & -5 \end{bmatrix} \xrightarrow{\text{norm}} \begin{bmatrix} \frac{1}{\sqrt{6}} & 0 & \frac{1}{\sqrt{30}} \\ \frac{2}{\sqrt{6}} & \frac{1}{\sqrt{5}} & \frac{2}{\sqrt{30}} \\ \frac{1}{\sqrt{6}} & -\frac{2}{\sqrt{5}} & -\frac{5}{\sqrt{30}} \end{bmatrix}$$

PAGE NO.
DATE :

$$S = \begin{bmatrix} \sqrt{12} & 0 \\ 0 & \sqrt{10} \end{bmatrix}_{2\times 2}$$

$$S = \begin{bmatrix} \sqrt{12} & 0 & 0 \\ 0 & \sqrt{10} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Calculation for V

$$10v_1 + 0v_2 + 2v_3 = \lambda v_1$$

$$0v_1 + 10v_2 + 4v_3 = \lambda v_2$$

$$2v_1 + 4v_2 + 2v_3 = \lambda v_3$$

$$\begin{vmatrix} 10-\lambda & 0 & 2 \\ 0 & 10-\lambda & 4 \\ 2 & 4 & 2-\lambda \end{vmatrix} = 0$$

$$(10-\lambda)(20+\lambda^2-12\lambda-16) + -2(20-2\lambda) = 0$$

$$(10-\lambda)\left[(\lambda^2-12\lambda+\cancel{4}) + \cancel{4}\right] = 0$$

$$(10-\lambda)(\lambda^2-12\lambda+\cancel{8}) = 0$$

$$\lambda(10-\lambda)(\lambda-12) = 0$$

$$\lambda = \frac{12 \pm \sqrt{144-32}}{2}$$

$$\lambda = 12 \pm \sqrt{}$$

$$\lambda = 0, 10, 12$$

$$\lambda = 12, 10, 0$$

$\underset{2\times 2}{U} \cdot \underset{2\times 2}{S} \cdot \underset{3\times 3}{V^T}$ = should give $A_{2\times 3}$

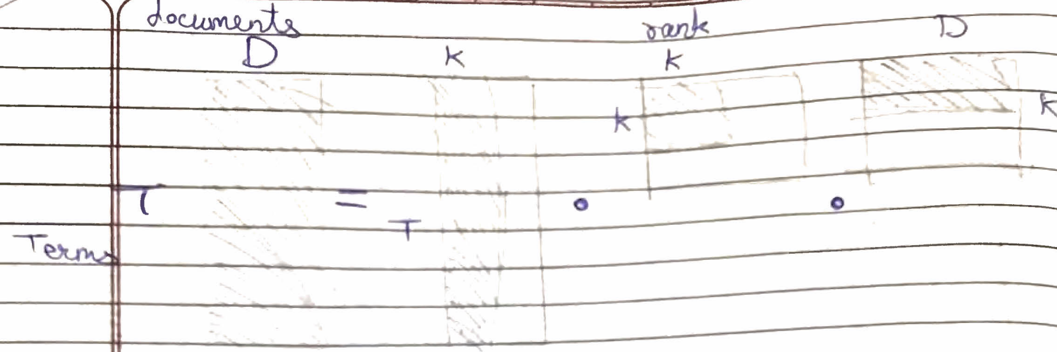
Not compatible.

∴ Change S $\longrightarrow$ $\begin{bmatrix} \sqrt{12} & 0 & 0 \\ 0 & \sqrt{10} & 0 \end{bmatrix}_{2\times 3}$

0 0 0

10/03/2023
Friday

PAGE NO.
DATE: Lec 12



$$A_k = U_k \cdot S_k \cdot V_k^T$$

Dynamic Programming

Reuse values from previous calculations

Ref # 24/02/2023

Parameter estimation for HMM

1 Viterbi Algorithm. (Ref # 15/03/2023)

2 Forward Algorithm (Ref # 15/03/2023)

3 Baum Welch Algorithm. (Ref # 15/03/2023)

PAGE NO.
DATE :

Expectation Maximization Algorithm. (EM Algorithm)

Note

KMeans is hard-clustering algo.

Hard clustering - One point belongs to only one cluster.

Soft clustering - Overlapping clusters.

fuzzy clustering | probabilistic

→ EM Algorithm is soft clustering algorithm.

15/03/2023
Wednesday

Ref # 17/02/2023

PAGE NO.
DATE: Lec 13

λ Hidden Markov Models.

π 1 Initial probability vector
q 2 # states
A 3 state transition prob. matrix.
B 5 Emmission probability matrix
$\frac{V}{O}$ 4 Observation from each state

Also in Observable markov models.

— Notation: $q_1 q_2 \cdots q_n$ / $Q_1 Q_2 \cdots Q_n$ } states

— q_{ij} = transition probability from q_i to q_j

— $O = o_1 o_2 o_3 \cdots o_n$ } observation

— $B = \begin{bmatrix} b_i(V_j) & \cdots \\ & \end{bmatrix}$

— probability that observation V_j is emmitted by state q_i