Natural language Processing

○ NLP problem use case.

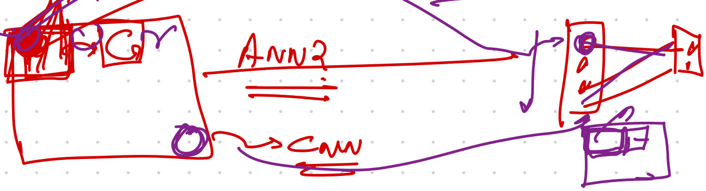
○ How do we structure words / sentence

↓

Feature engineering

○ Sequence models → RNN, LSTM, GRU, Transformer

Attention



vectors

words → Embeddings → it's a way to structure our language

words / sentences / documents → Embedding Vectors → Sequence model

Large language models

RLHF → ChatGPT

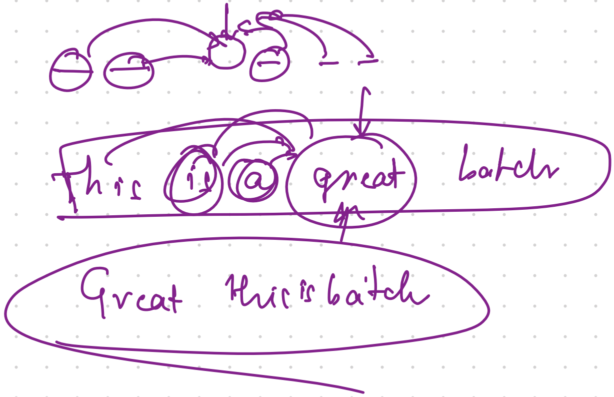
Img → Text

multimodal

Question → Encoder → [ ] → Decoder → Answers

Prompt

Natural language - is a Sequence

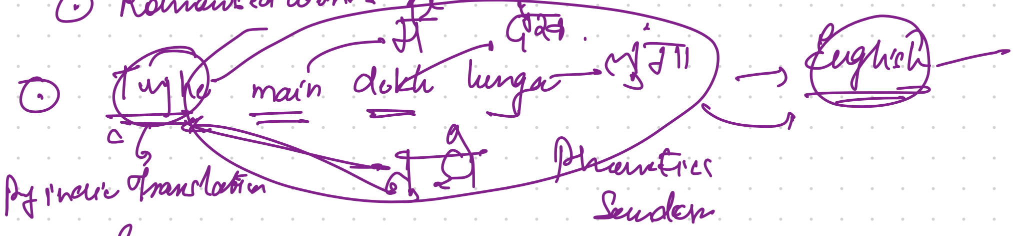


Natural language Processing

Understanding information from text

- Unstructured →
- Content / Sentiments / slangs / Sarcasm

- Indiga . . . .
- Romanised words



- Structured

Use Cases:

Translation

gmail auto type → auto fill

Search engines

Sentiment analysis

Gmail | Spam → Spam
→ HAm

Chatbots

Text classification, Sum

NER, information

sementitial

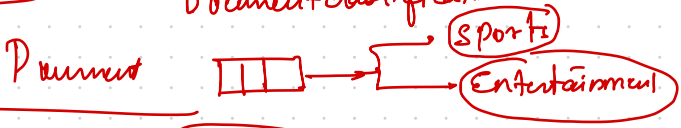
$\boxed{Y} = \boxed{f}(x)$ Language model,

$\boxed{\text{Language model}}$

Sentences are very unclear ← Socialmedia
← Human Conversations

Preprocessing techniques:

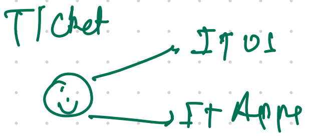
Document classification



Special characters @ http....

Removal of unwanted characters or patterns of characters from our source sentence.

Ticket classifier



Sentiment classification

Regex → a process (logic) of defining certain patterns using certain rules.

" I am very hungry. I will order ~~s~~ pizzas and give ~~it~~ to my friend!"

Regex → Extract some defined patterns from sentences

[w]
[s]

Name: Subhodeep
Age: 30
Education: master

~~Stan~~ words [Name]
words [Age]

Stopwords an, a, the
because and Then

Translation , Chatbot → Tomo Tem 25'C
→ I am en------

Whats the temp tomorrow

मैं घर जारहा हूँ → Tom ~~I am~~ going home

Stemming / Lemmatization

Embedding
words → vectors → model

Embedding

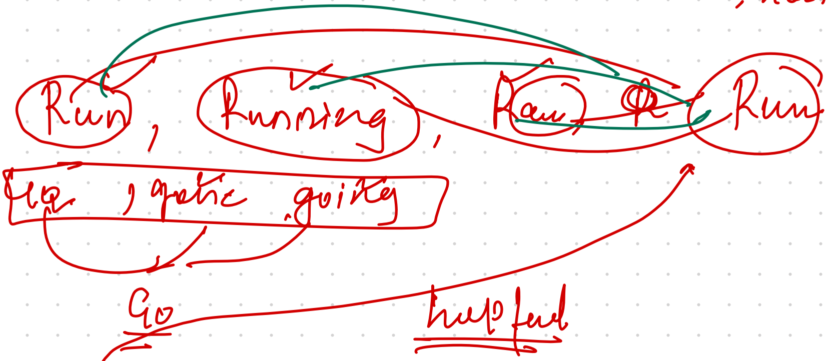
Frequency based
Count of the words
in the sentence

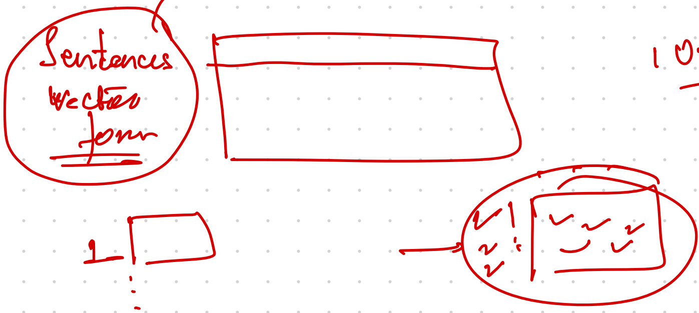
Content?

Prediction based.
vectors generated
through NN.

I am going home., home is where heart is

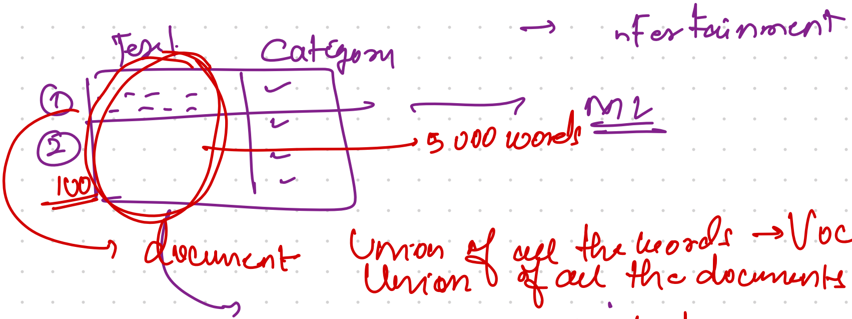
I : 1 , am : 1, going : 1 home 2 , is : 2
where 1., heart : 1





10:33 → 10:40pm

How to represent sentences /documents in a vectorised form using frequency embeddings.



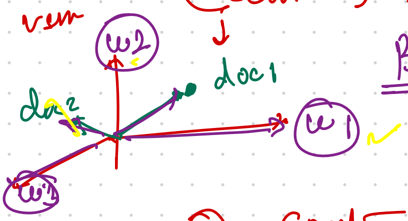
Union of all the words → Vocabulary.
Union of all the documents → Corpus
one word → Token

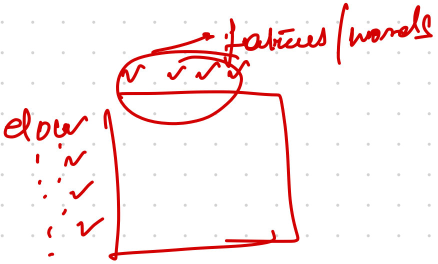
Bag of words model

Document classification

(Document) → Outcome

BOW

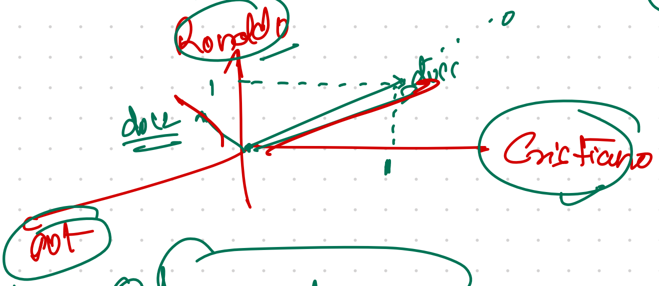




① Cristiano Ronaldo got transferred. Cristiano is great
→ Sports

② Puca hrpa Performs in Cannes.

	Cristiano	Ronaldo	got	transferred	Dua	Lipa	Performs	in	Concert
1	1	1	1	1	0	0 1	0	0	0
2	0	0	0	0	1	1	1	1	1

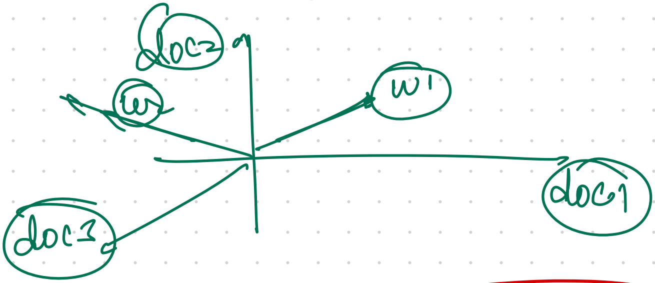


Problem: ① too many dimensions

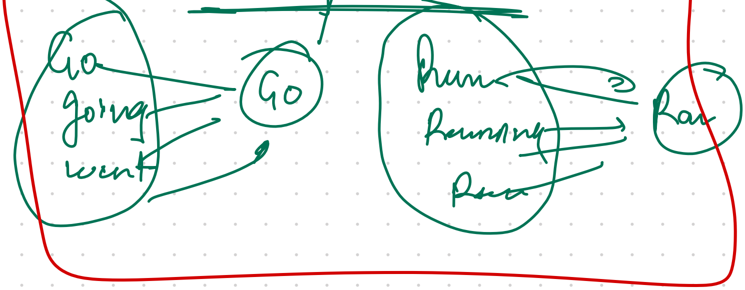
② does not maintain context

↓ order not maintained

By definition given all words are independent



Stemming / Lemmatization



Stemming & Lemmatization
Enforcement

Porter Stemmer

Lemmatization

Verb

Running → Run
Ran

Speeding ✓
Speeds ✓
Caresses ✓
Cry → Cri

going → go
gone
go

verb
I am go to visit my home and will
I am going to visit my home and will make → make

Sure my ac is running
sure my ac is run

✓ Dependency Par
✓
~ my own car awesome
~ POS

"I | am | great | at | public | speaking"

Stemming & Lemmatization
(word)

word tokenizer
Sentence → words

Sentence tokeniser

documents → Sentence

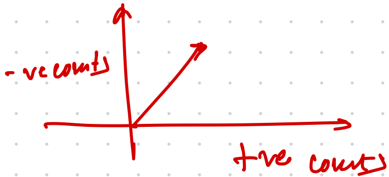
.split(' ')

" I | am | going | to | my | home" .split(' ')

['I', 'am', 'going', 'to', 'my', 'home']

~~to~~ nltk → word.tokeniser (' ')

tweet → words → (count of the word in the tweets, count of the word in negative tweets)



tweet → (count of the +ve words, count of -ve words, 1)

+ve 50 10

60 100

+ve -ve

tweet: The Game

+ve

tweet (1, 110, 110)

freqs

{(game,0): 20
(game,1): 40
(awesome,0): 50
(awesome,1): 90 }

tweet: game is awesome

game awesome

pos = freqs[(game,1)] = 40 + freqs[(awesome,1)] ,90

neg = freqs[(game,0)] = 20 + freqs[(awesome,0)] 50

pos = 130

neg = 70

tweet = (1, 130, 70)