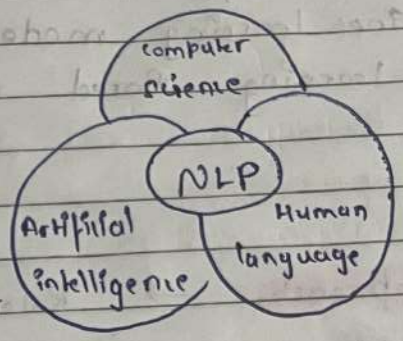
PAGE NO.:
DATE: / /

Natural language processing.



- Real world Application.
  - Contextual Advertisement
  - Email client - spam filtering
  - social media - removing adult content

↳ common NLP tasks

(1) Text / Document classification
(2) Sentiment Analysis
(3) Information Retrieval
(4) Part of speech tracking.
(5) language Detection and Machine Translation
(6) Conversational Agents.
(7) Knowledge graph & QA System.

(8) Text Summarization
(9) Topic modelling
(10) Text generation.
(11) spell checking and Grammar correction
(12) Text Parsing
(13) Speech to Text.

{ Rules are made By Human itself }

PAGE NO.:
DATE: / /

- Approaches to NLP:

→ Heuristic Approaches.
→ Machine learning models
→ Deep learning Based models.

- ML approach. → Rules are made by model itself after seeing the data.

→ ML flow

Algorithms used.

- Naive Bayes
- logistic regression
- SVM
- LDA
- Hidden Markov model.

- Deep learning model approach. :-

Advantage → Retain Sequential Information.

Architecture used.

- RNN
- LSTM
- GRU / CNN

(Gated Recurrent Unit) → Text generation.

- Transformers
- Auto encoders.

PAGE NO.:
DATE: / /

- Challenges in NLP
  - It is difficult to make machine understand all these

(1) Ambiguity
  - I saw the boy on the beach with my binoculars
  - I have never tasted a cake quite like that one before

(2) contextual words.
  - I ran to the store because we ran out of milk.

(3) Colloquialisms and slang.
  - Piece of cake, - pulling your leg.

(4) Synonyms

(5) spelling error

(6) Creativity

(7) Irony, sarcasm

PAGE NO.:
DATE: / /

- NLP pipeline.

(1) Data Acquisition

(2) Text Preparation → Text cleanup.
→ Basic Preprocessing
→ Advance Preprocessing

(3) feature Engineering.

(4) Modelling → model Building
→ model evaluation

(5) Deployment.
→ Deployment
→ monitering
→ model updates.

Pornb.

→ this pipeline is not universal
→ Deeplearning pipelines are more difficult

* Data Acquisition :- Data,

Available.
→ table.

less data.
→ data Augment
→ synonym.
→ bigram flip
→ Back translate.
→ add noise

PAGE NO.:
DATE: / /

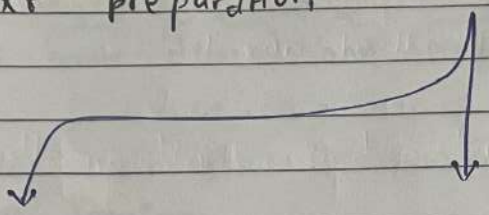
Other.

→ public dataset

→ Web scraping. → beautiful soup.

→ Api → rapid API → request → json.

* Text preparation



Cleaning.

- html tags cleaning.
- emoji (unicode normalization)
- spell checking.

Basic preprocessing

Basic

- tokenization
  - sentence
  - words.

optional

- stop word removal
- stemming
- Removing digits
- lower casing
- language detection.

* **Advance preprocessing.**

- Pos tagging → (Parts of speech tagging)
- Parsing. → understand syntax structure. (eg. chatbot)
- Co-reference Resolution eg. (Caplin directed, composed music of movies.) Both same. person

his

PAGE NO.:
DATE: / /

* Feature Engineering :

→ Task is to convert words into numbers.
techniques can be.

① Bag of words
② Tfidf
③ OHE (one Hot encoding)
④ word 2 vec

} depends on Application

* modelling.

ml models (less data)

DL models (more data needed)

* evaluation

→ Intrinsic

extrinsic (Business Based)

→ perplexity.

* Deployment

example :-

→ Input

chaplin wrote, directed and composed the music for most of his film

→ Tokenization with lemmatization
(stemming)

chaplin write direct and compose the music for most of the film.

chaplin wrote, directed and composed the music for most of his film

→ POS Tagging.

NNP VBD VBD CC VBN DT NN IN JJS IN PRP$

chaplin wrote, directed and composed the music for most of his fi

→ Coreference Resolution

mention ..... coref ..... mentio

chaplin wrote, directed and composed the music for most of his

Text Representation :-

{Hello, how are you?}

=> Topic is to convert "text meaning into numbers"

* Common words.

↳ (1) corpus → Combination of all the words in the Dataset. {words can repeat}

↳ (2) Vocabulary → All the unique words of corpus is vocabulary.

↳ (3) Document → Each "review" in Dataset is Document.

↳ (4) word. → any word in Dataset

PAGE NO.:
DATE: / /

(1) One hot Encoding :- Not used These days

(sparse array)
(no fixed size)
(no capturing of symentic meaning

D_1 - people watch campus x
D_2 - campus x watch campus x
D_3 - people write comment
D_4 - campus x write comment

Corpus

↳ people watch campus x campus x watch campus x people write comment campus x write comment

Vocabulary. $(V = 5)$

↳ people watch campus x write comment

now.

every sentence or document D_1, D_2, D_3, D_4 will be converted "V" dimensional array.

	people	watch	campus x	write	comment
people →	1	0	0	0	0
watch →	0	1	0	0	0
campus x →	0	0	1	0	0
write →	0	0	0	1	0
comment →	0	0	0	0	1

$D_1 = [[1,0,0,0,0], [0,1,0,0,0], [0,0,1,0,0]]$

$D_2 = [[0,0,1,0,0], [0,1,0,0,0], [0,0,1,0,0]]$

PAGE NO. :
DATE : / /

Bag of words :

	text	output
D_1 =	people watch campusx	1
D_2 =	campusx watch campusx	1
D_3 =	people write comment	0
D_4 =	campusx write comment	0

$V = 5$

	People	watch	campusx	write	comment
D_1	1	1	1	0	0
D_2	0	1	2	0	0
D_3	1	0	0	1	1
D_4	0	0	1	1	1

How many time the word is coming in document

Better than "OHE"

Advantages.

1. Sparcity
2. OOV
3. Do not consider ordering.

Disadvantages.

N- Grams.

D_1	people	watch campusX	1
D_2	campusX	watch campusX	1
D_3	people	write comment	0
D_4	campusX	write comment	0

Bi- Gram → Vocabulary of two words.
tri- Gram → Vocabulary of 3 words.

Volabulary of Bigram. ⇒ { ① people watch, ② watch campusX, ③ campusX watch, ④ people write, ⑤ write comment, ⑥ campusX write }

	①	②	③	④	⑤	⑥
D_1	1	1	0	0	0	0
D_2	0	1	1	0	0	0
D_3	0	0	0	1	1	0
D_4	0	0	0	0	1	1

Bi- Gram is better than Bag of words.

eg.
D_1 This movie is very good
D_2 This movie is not good.

Vocabulary ⇒ This movie is very good not

① BOW → D_1 1 1 1 1 1 0
→ D_2 1 1 1 0 1 0

These two vectors are not far apart in 6D space.

60 space.

PAGE NO.:
DATE: / /

Bigram.

	①	②	③	④	⑤	⑥
Volabulary =>	This movie	movie is	is very	very good	is not	not good
D_1	1	1	1	1	0	0
D_2	1	1	0	0	1	1

↑ ↑ ↑ ↑

vectors are far apart than BOW

→ able to capture more semantic of the sentence.

→ Its advantage.

→ slows down the algo

→ dimension increases

→ out of vocab

Tf-Idf :-

Concept is when a word is rare in the whole corpus But is coming more times in a Document then it will allot more importance to that word to that Document

Tf → Term frequency

Idf → Inverse dot frequency

PAGE NO.:
DATE: / /

TF (Term frequency) = (Number of ourrence of term "t" in doument "d") / (Total number of term in document d)

IdF (inverse dor frequency) (t) = $\log_e$ (Total number of document in the corpus) / (number of documents with term t in them)

D_1	people watch campux	1
D_2	campuex watch campuex	1
D_3	people write comment	0
D_4	campuex write comment	0

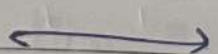
	IDF
people.	log (4/2)
watch	log (4/2)
campuex	log (4/3)
write	log (4/2)
comment	log (4/2)

if the "term" is coming in every Doument then (IDF = 0) so we add 1 in it so that they should not be ignored entirely.

TF (people, D_1) = 1/3

TF (watch, D_1) = 1/3

value of vector of every word in Doument = (TF x IdF)



	people	watch	campus	write	comment
D_1	$\frac{1}{3} \times 0.3$	$\frac{1}{3} \times 0.3$	$\frac{1}{3} \times 0.125$	0	0
D_2	0	$\frac{1}{3} \times 0.3$	$\frac{2}{3} \times 0.125$	0	0
D_3	$\frac{1}{3} \times 1.3$	0	0.	$\frac{1}{3} \times 0.3$	$\frac{1}{3} \times 0.3$
D_4	0	0	$\frac{1}{3} \times 0.125$	$\frac{1}{3} \times 0.3$	$\frac{1}{3} \times 0.3$

* Advantages

→ good in information retrival

* disAdvantages

→ sparsity → Semantic meaning (cannot take out)
→ OOV
→ dimension

* word embeddings :-

→ words are represented in the form of vectors that encodes the meaning of the word such that the words that are closer in the vector space are expected to be similar."

PAGE NO. :
DATE : / /

* Word 2 Vec :- (Deeplearning Based embedding Technique)

↳ (can capture symentic meaning very properly)

→ eg. can capture the closeness of the word "joy" and "Happy" very ~~proper~~ accurately.

↳ **Low dimensional** → (300)

→ makes computation easy.

↳ **Dense vector**

→ very less non-zero values, ~~w~~ which reduces overfitting.

→ Demo of word 2 vec

- Intiution :- word 2 vec

→ It creates features to them and apply probabilistic values.

features ↓	King	Queen	Man	woman	monkey
gender	1	0	1	0	1
wealth	1	1	0.3	0.3	0
Power	1	0.7	0.2	0.2	0
Speak.	1	1	1	1	0
weight	0.8	0.4	0.6	0.5	0.3

features of King = [1 1 1 1 0.8]

PAGE NO. :
DATE : / /

let's do

king	− man	+ women	=
1	− 1	+ 0	= 0
1	− 0.3	+ 0.3	= 1
1	− 0.2	+ 0.2	= 1
0.8	− 0.1	+ 0.5	= 0.7
1	− 1	+ 1	= 1

(1)

→ Note → The model donot create feature that the example shown above

→ It create features like f_1, f_2, f_3 ---

→ (2) Who word2vec decide what the feature will be ??

=) The underlying assumption of word2vec is that two words sharing similar contexts also share a similar meaning and consequently a similar vector representation from the model.

[8.0 1 1 1] =

Word2 Vec

(CBOW) (Skip gram)

PAGE NO.:
DATE: / /

(1) CBOW :- (Continuous Bag of words)

→ It creates a dummy problem to create context of the words.

eg. :- (watch) campusx for data science

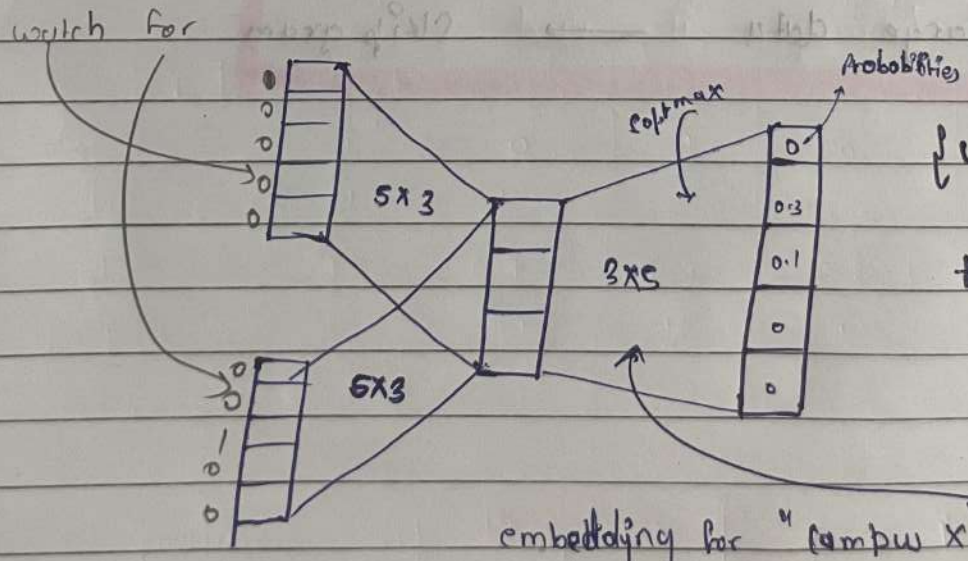
Target

X	Y
① watch, for	campusx
campusx, data	For
For, science	data

Vector for each word.

watch → [1 0 0 0 0]
campusx → [0 1 0 0 0]
for → [0 0 1 0 0]
data → [0 0 0 1 0]
science → [0 0 0 0 1]

↳ now creating a neural network from scratch.



Values will be compared with output vector then Backpropagation will happen and weights and biases will be rearranged

embedding for "campus x" will be weights of ← hill layer.

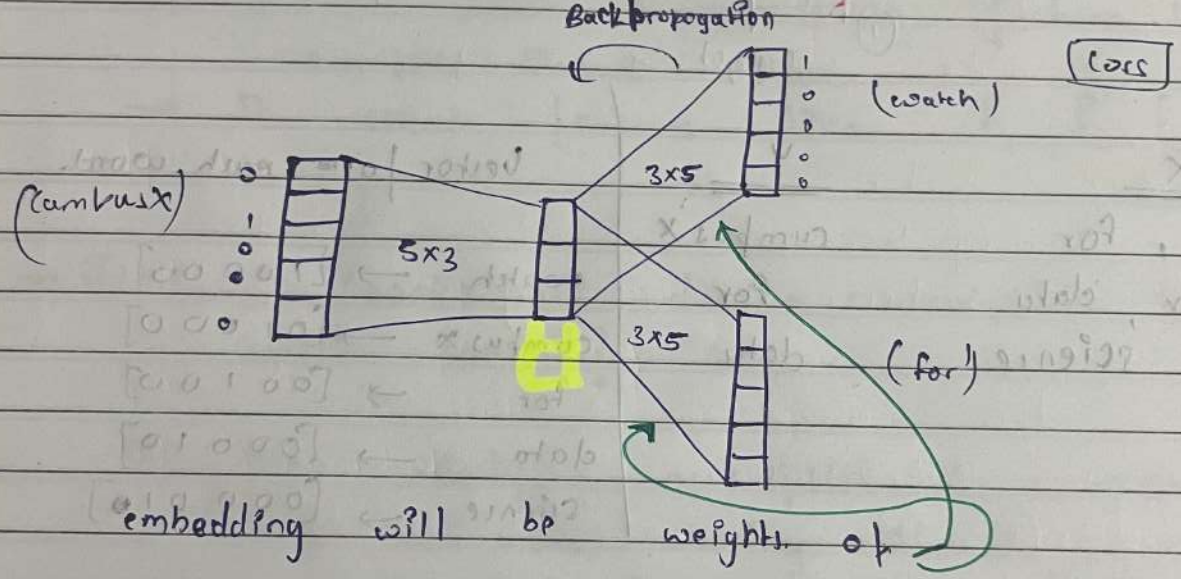
PAGE NO.:
DATE: / /

* Skip - Gram :-

window = 3

watch campusx for data science

x	y
campusx for data	watch, for campusx, data for, science



embedding will be weights of

Note. :-

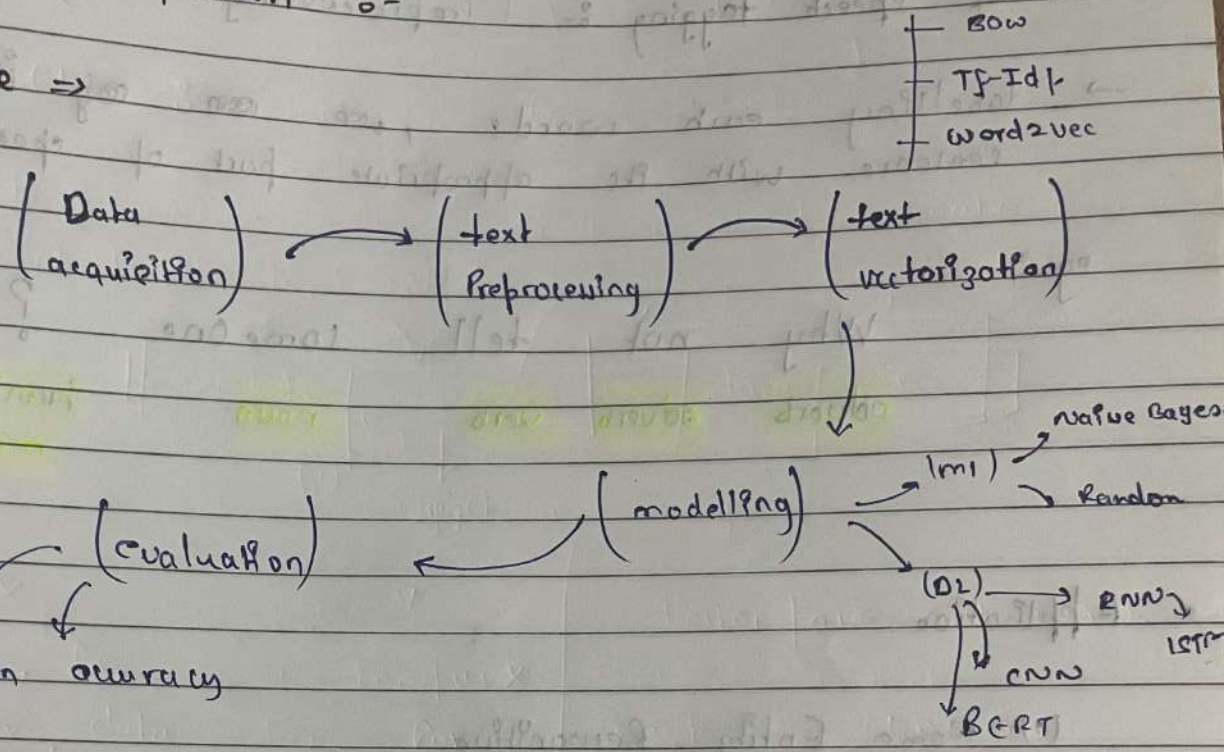
small data ⟶ CBOW

large data ⟶ Skip gram

PAGE NO. :
DATE : / /

Text classification :-

Pipline →



PAGE NO.:
DATE: / /

* Parts of speech tagging :- (Preprocessing task)

→ labelling each words, one tag in a sentence with its appropriate part of speech.

eg.:-

Why	not	tell	someone	?
adverb	adverb	verb	noun	punctuation mark.

- Application

(1) Name Entity Recognition
(2) Question Answering system
(3) word sense disambiguation.
(4) chatbot

→ POS works on "Hidden Markov model"

→ Spacy library is used