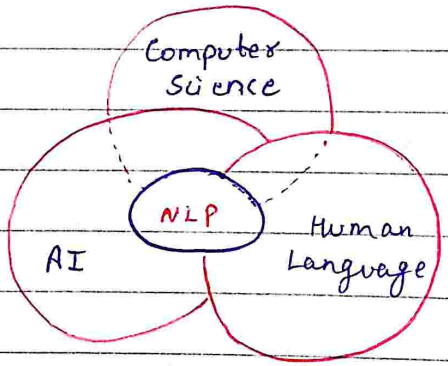
NLP

4/4/2024

INTRODUCTION

Natural Language processing is a subfield of linguistics, computer science, and artificial intelligence concerned with the interactions between computers and human language, in particular how to program computers to process and analyze large amounts of natural language data.



Real World Applications

- Contextual Advertisements
- Emails clients - spam filtering, smart reply
- Social Media - removing adult content, opinion mining
- Search Engines
- Chatbots

Common NLP Tasks

- Text/ Document Classification
- Sentiment Analysis
- Information Retrieval
- Parts of speech Tagging
- Language Detection & Machine Translation
- Conversational Agents
- Knowledge Graph and QA systems
- Text Summarization
- Topic Modelling
- Text Generation
- Spell checking & Grammar Correction
- Text Parsing
- Speech to Text

Approaches to NLP

→ Heuristic Approaches
→ ML Based Methods
→ DL Based Methods

★ Heuristic Approaches
- जुगाड़ लगाना
Ex → Regular Expressions
→ Wordnet

challenges in NLP

- Ambiguity
- Contextual words
- Colloquialisms & slang
- Synonyms
- Irony, Sarcasm & tonal diff
- Spelling Errors
- Creativity
- Diversity

NLP Pipeline

6/4/24

↳ WHAT IS NLP Pipeline

→ NLP is a set of steps followed to build an end to end NLP software.
→ NLP software consists of the following steps :

- Data Acquisition
- Text Preparation
  - Text cleanup
  - Basic Preprocessing
  - Advance Preprocessing
- Feature Engineering
- Modelling
  - Model Building
  - Evaluation
- Deployment
  - Deployment
  - Monitoring
  - Model Update

↳ Points to Remember

→ It is not universal
→ Deep learning pipelines are slightly different
→ Pipeline is non-linear

Data Acquisition

Data Augmentation
- Synonym
- Bigram Flip
- Back translate
- Add Noise

TEXT Preparation

→ Clean Tag / HTML Tag
→ Clean Emoji
→ Spelling Checker

Basic Preprocessing

Basic
↓
Tokenization
↓ ↓
Syntax Word
Sentence

Optional
- Stop word Removal
- Stemming
- Remving. digits, gender
- Lowercasing
- Lang. detection

Advance Preprocessing
- POS tagging
- Parsing
- Coreference resolution

Feature Engineering

- Feature selection
- OHE, BoW, Word2Vec ...

Modelling

Modelling
Evaluations
intrinsic extrinsic

Deployment

Deploy Monitoring Update

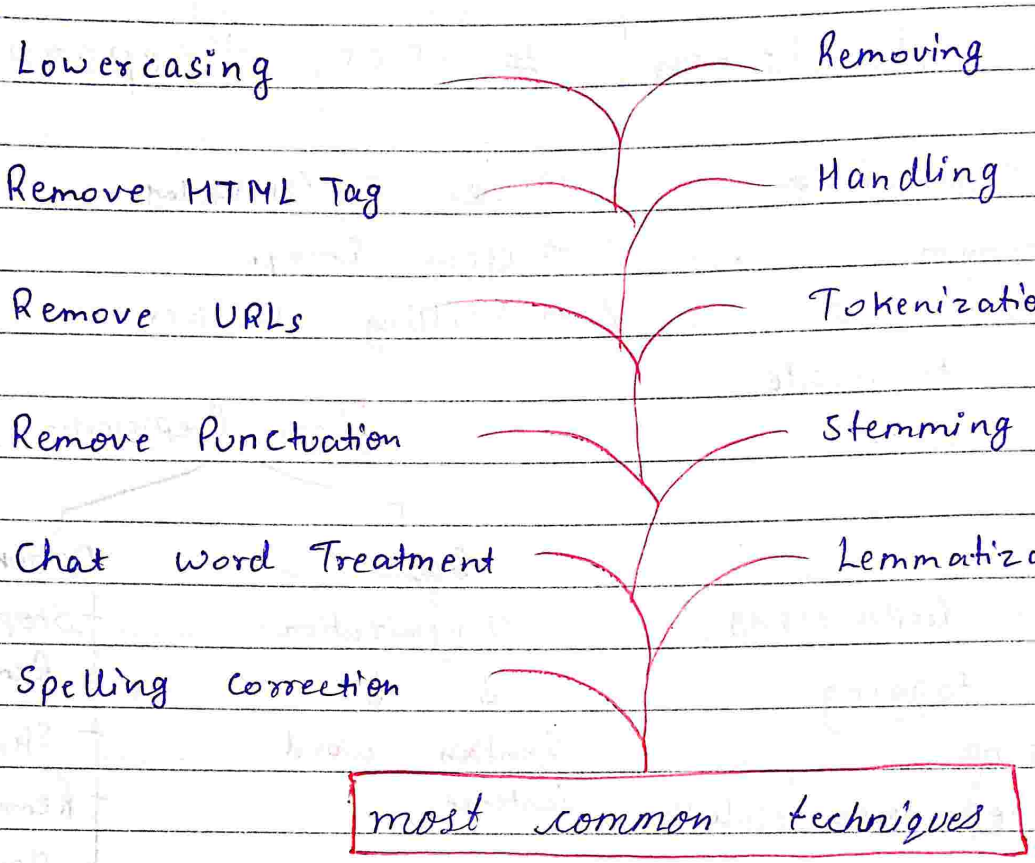
TEXT

PREPROCESSING

rgen101.com

IIMT

6/4/24



Punctuation : '!"#$%&\'()*+,-./:;<=>?@[\\]^_`{|}~'

Spelling correction : 'textblob'

Removing stop words : If you will apply POS then don't apply stop words 'nltk.corpus' + some code

Handling emoji → emoji.demorize

NLTK, Spacy for → Tokenization

Stemming

In grammar, inflection is the modification of a word to express different gramatical categories such as tense, case, voice, aspect, person, number, gender and mood.

Ex walk, walking, walked, walks

"Stemming is the process of reducing inflection in words to their root forms such as mapping a group of words to the same stem even if the stem itself is not a valid word in the language."

NLTK

Porter Stemmer → only for English

Snow ball Stemmer → for other languages

Problem in Stemming

⇒ Non english / language word पर भी Apply होता है।
जैसे Story → stori

Sol^{n} ⇒ Lemitization

6/4/24

Text Representation

IIMT

Introduction

Similar terms: Text vectorization, representation, feature extraction

OHE {One Hot Encoding}

Disadvantages
- → sparsity
- → No fined size
- → OOV {Out of Vocabulary}
- → No capturing of semantic

Common terms
- → Corpus (C): Concatenation of all words in dataset
- → Vocabulary (V): Unique words in Corpus
- → Document (D): string of each datapoint of each feature
- → Word (W): group of characters

BoW {Bag of Words}

	Data	Label
D_1	people watch campusx	1
D_2	campusx watch campusx	1
D_3	people write comment	0
D_4	campusx write comment	0

Adv
- Simple & intuitive

Disadv
- # Sparcity
- OOV
- Disordering of Words

- V-dimentional vector

	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

with Binary
without Binary

Why log in IDF? bcz if 1 LKH total docs & only a word present only in one document then without log IDF = 100000 हो TF की Dominate कर देगा {Mean TF का कोई महत्व नहीं रह जाता}

N-grams

"Bag of N-grams"

NOTE» हमारे Prev Data पर हम quad-gram नहीं बना सकते

bi-, tri-, n-, ⇒grams

As per previous data
here we are using bi-grams

	people watch	watch campusx	campusx watch	people write	people write comment	campusx write
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

Benefits	Disadvantages
→ Able to capture semantic of the sentence	→ Slows down the Algo
→ easy implement	→ ↑se dim
	→ OOV

TF-IDF

Term Frequency
Inverse Dot Frequency

In Sklearn → TF × (IDF + 1)

$0 < TF < 1$

TF × IDF

Why? If not, if a word present in all documents then $\log_e(\frac{N}{N})$ will 0 final → 0

Advantage
↓
Used in Information Retrieval Systems
↓
Search Engines

Disadvantages
Same as Prev

$$TF(t,d) = \left(\frac{\text{No. of occurences of term } t \text{ in document } d}{\text{Total no. of terms in document } d}\right)$$

$$IDF(t) = \log_e \frac{(\text{Total no. of documents in the corpus})}{(\text{No. of documents with term } t \text{ in them})}$$

Word 2 Vec

8/4/24

- A technique of text vectorization which able to understand semantic meaning it actually a Deep Neural Network.

Types

- CBOW
  - Continuous Bag of Words
  - ↳ for small data
- skip-gram
  - ↳ for large data

CBOW

fake problem
- ↳ solve → vector
  - ↳ by product

dummy problem

Skip-Gram

→ reverse of CBOW neural networks

Watch campusx for data science

POS TAGGING

spacy

IIMT

11/4/2024

What?

In simple words, we can say that POS tagging is a task of labelling each word in a sentence with its appropriate parts of speech.

In traditional grammar, a part of speech or part-of-speech is a category of words that have similar grammatical properties.

Example ⇒ Why not tell someone ?
Adverb Adverb verb noun Punctuation mark, Sentence closer

Applications

1. Named Entity Recognition
2. Question Answering System
3. Word sentence disambiguation
4. Chatbots

common example → will will Google Campusx

How POS Tagging works?

S- Nitish(N) loves(V) campusx(N) - E

S- can(M) Nitish(N) google(V) campusx(N) -E

S- will(M) Ankita(N) google(V) campusx(N) - E

S- Ankita(N) loves(V) will(N) - E

S- will(M) loves(V) google(N) - E

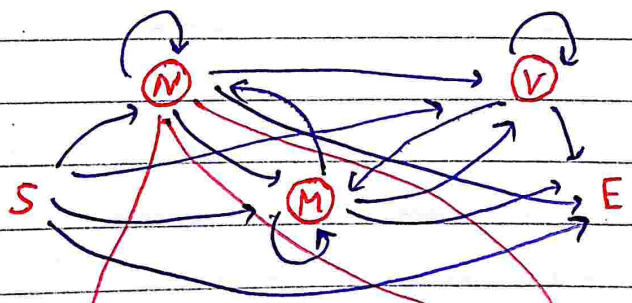
Emission Prob

	N	M	V
Nitish	2/10	0	0
loves	0	0	3/5
campusx	3/10	0	0
Google	1/10	0	2/5
will	2/10	1/2	0
Ankita	2/10	0	0
can	0	1/2	0

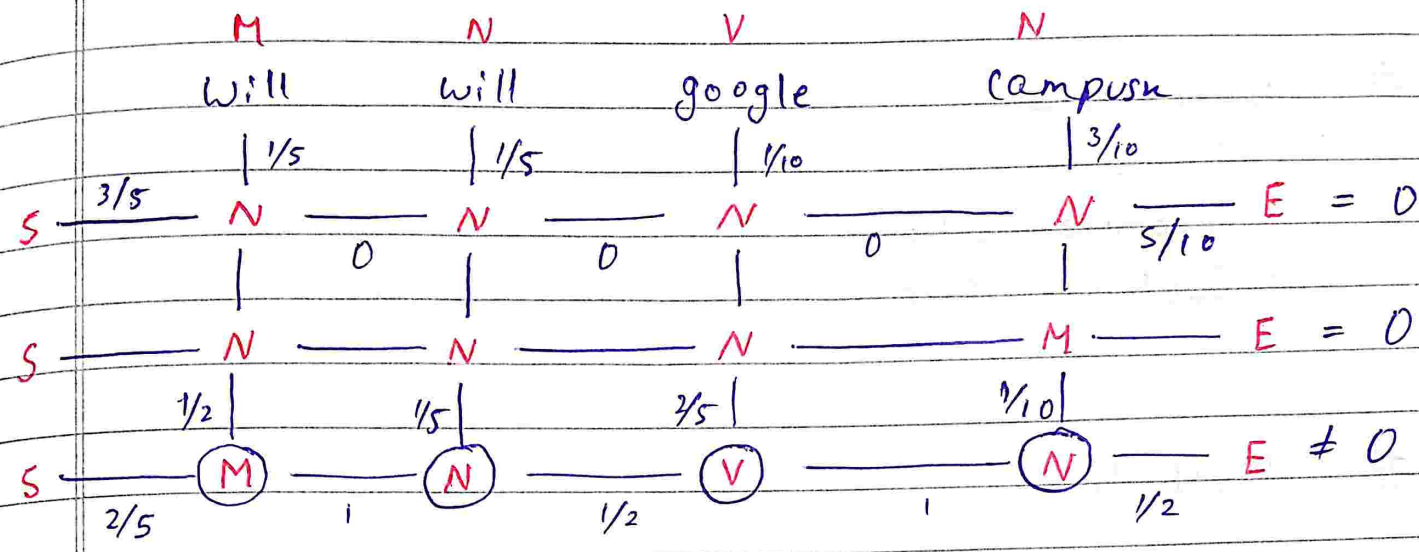
Transition Prob

	N	M	V	E
S	3/5	2/5	0	0
N	0	0	5/10	5/10
M	2/2	0	0	0
V	5/5	0	0	0

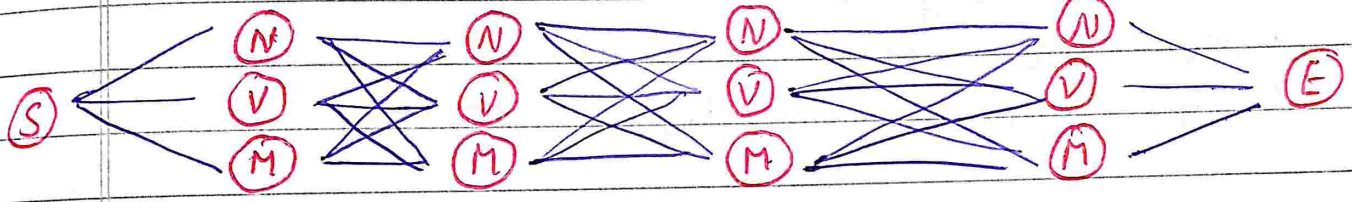
Hidden Markov Model



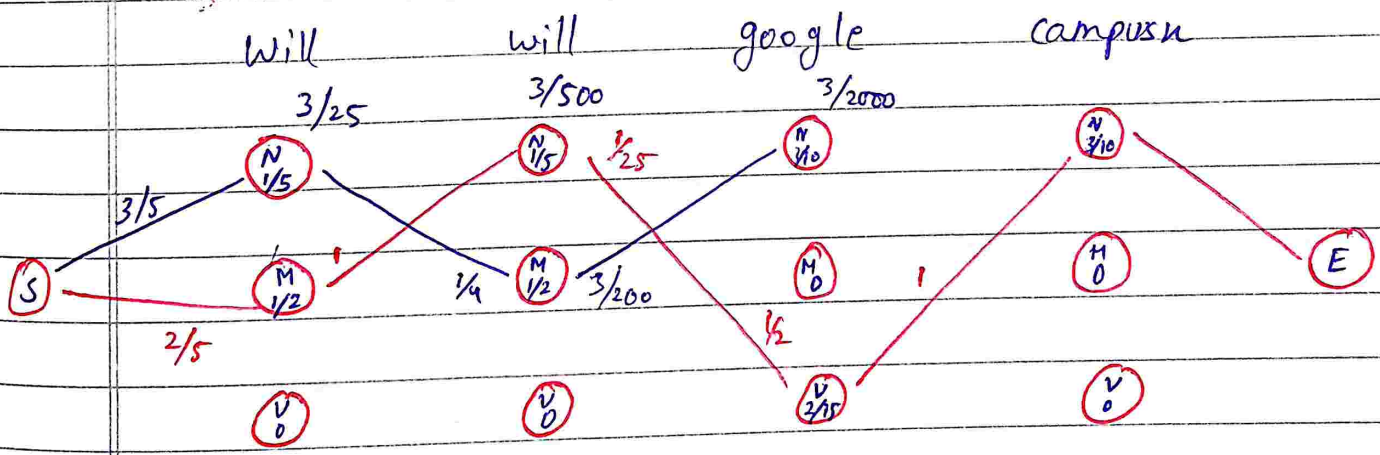
Nitish loves campusx will Ankita can



Will will google campusn $3^4 = 81$



Viterbi Algorithm



→ All probabilities calculated by multiplication at the end.

→ Next Topics : Topic Modelling, Named Entity Recognition

NER

Named Entity Recognition

9/4/24

Applications

- Search
- Recommendations
- Customer Care

Methods

1) Lexicon Based Methods → Basically Heuristic Methods, X-training
2) Rule Based Methods → Regex, Grammar based patterns
3) Machine Learning based Method
   - Multiclass classification
   - Conditional Random Field (CRF)

4) Dictionary/Lookup-Based Methods
   → Match words against a predefined dictionary or gazetteer
   → Good Precision, Low Recall (Misses unseen entities)