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

Preprocessing :-

Preprocessing consists of (a) tokenization (b) normalization (c) substitution.

→ Case folding → white space, newline, tabs
→ Stemming → removing contraction
→ Lemmatization → remove scripts, form variables → Tokenization
→ remove misspellings → Punctuations

Statistical Properties of Words :-

Process & harness info from large corpus

Ideal Properties of a language corpus

→ Collection of a written text in a digital form
→ Useful to verify a hypothesis about a language
→ To determine how the usage of a particular sound word or syntactic construction varies in different content
→ Contains most of the words of a language
→ changes as function of time - regular increase of corpus size with addition of new text samples.
→ corpus is huge - several billions of words
→ Even distribution of text from all domains of language use.
→ Represents all areas of coverage of texts of a language.
→ Access of language data in an easy and simplified manner.

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

Typical NLP tasks :-

Information Extraction	find documents based on keywords
Language Generation	Description based on a photo
Information Extraction	Identify & extract personal name, date etc
Text Clustering	Automatic grouping of documents
Text Classification	Assigning predefined categorization to documents
Machine Translation	Translate any language text to another
Grammer checkers	check the grammar for any language

⇒ A corpus is a collection of machine readable text collected according certain criteria
⇒ Representative collection of text
⇒ Used for statistical analysis and hypothesis testing
⇒ Used for validating linguistic clues within a specific language

Statistical Properties of Words

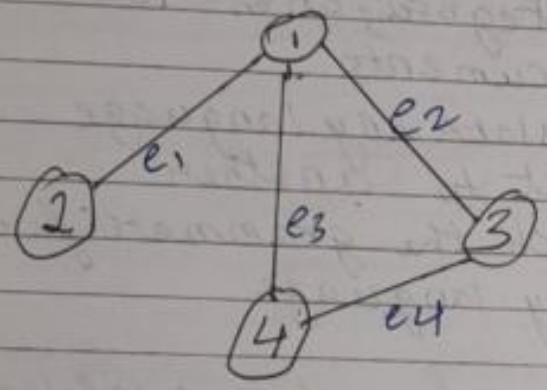
Date ..../..../.....

Incidence Matrix

Let G be a graph with n vertices $(v_1, v_2 \ldots v_n)$ and m edges $(e_1, e_2 \ldots e_m)$. Then incidence matrix of size $n \times m$ is defined as

$$x_{ij} = \begin{cases} 1 & \text{if there is an edge connecting } i \text{ \& } j \\ 0 & \text{otherwise} \end{cases}$$

It is also called vertex - edge incidence matrix and is denoted by $X(G)$



The incident matrix for this:-

	e_1	e_2	e_3	e_4
1	1	1	1	0
2	1	0	0	0
3	0	1	0	1
4	0	0	1	1

$$x_{ij} = \begin{cases} 1 & \text{if edge } i \text{ connect to vertice } j \\ 0 & \text{otherwise} \end{cases}$$

Term-Document Binary Incidence Matrix

	Antony & Cleopatra	Julius Caesar	The tempest
antony	1	1	0
brutus	1	1	0
caesar	1	1	0
calpurnia	0	1	0
cleopatra	1	0	0

This incidence matrix does not represent any information related word order & frequency

Spiral

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

IR Using Binary Incidence Matrix

	Antony & Cleo	JuliusCaesar	The Tempest	Hamlet	Othello
Antony	1	1	0	0	0
brutus →	1	1	0	1	0
caesar	1	1	0	1	1
calipurnia	0	1	0	0	0
cleopatra	1	0	0	0	0

Query:- Brutus AND Caesar AND NOT Calipurnia

Brutus & Caesar & not Calipurnia

110100 & 110111 & 101111

= 100100

The answer is found in Antony & cleopatra and Hamlet which are 1 in the answer

Words & Terms

→ word is our atomic unit for the purpose of NLP

→ Numerical representation of the word for computation purposes

→ Vocabulary of size $N = 1 \ldots 0$ is defined as $V = w_1, w_2, w_3, \ldots w_n$ is vocabulary containing unique words of a language

→ Some words found in V appear in documents ($D = D_1, D_2, D_3 \ldots, D_m$) once or several times or may not appear at all.

Spiral

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

Term Frequency

For the given document, term frequency is defined as the number of occurrences of a term, t_i, in a document d_i belonging to a corpus $(d_1, d_2, d_3 \ldots d_m)$.
This is denoted by $tf_{t,d}$

Multiple weighting factors TF

Boolean - 0, 1 Present / Absent
Raw count = $tf_{i,d}$ frequency count
Adjusted to document length: $\frac{tf_{i,d}}{M}$

log weighting - $\begin{cases} tf_{t,d} = 1 + \log tf_{i,d} & \text{if } tf_{i,d} > 0 \\ 0 & \text{otherwise} \end{cases}$

Disadvantage of raw frequency

→ All terms are given equal importance
→ The common term "The" has no relevance to the document, but gets high relevancy
→ May not be suitable for classification when common words appear in documents

Bag of words

The collection $[tf_1, tf_2, tf_5 \ldots tf_n]$ is known as bag of words
→ The ordering of the terms is no important
→ Two documents with a similar BOW are similar in content
→ It refers to the quantitative representation of the document.

Spiral

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

Type Token Ratio

The lexical variety of the text is defined the Type Token Ratio (TTR). It can be used to measure the vocabulary variation or lexical density of the written text and speech. The type is the unique vocabulary in the text which is devoid of any repititions

$$TTR = \frac{V}{T_n}$$

$\frac{\text{unique words}}{\text{total words}}$

where V is the vocabulary and T_n is the number of tokens in the speech or written text

It is not reasonable to compare two unequal sized documents. A standardized TTR is used for fair comparison where the token size is restricted to the first 15000 tokens

application of TTR:-

→ Monitor the vocabulary usage
→ Monitor child vocabulary development
→ Estimate the vocabulary variation in the text.

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

Inverse Document Frequency

In order to attenuate the effect of frequently occuring terms, it is important to scale it down and at the same time it is necessary to increase the weight of terms that occur rarely. Inverse document frequency (IDF)

$$IDF_t = \log\left(\frac{N}{Df_t}\right)$$ total no. of doc / document containing terms

N = total no. of documents in a collection

Df_t = Count of documents containing the term t

→ Rare documents gets a significantly higher value.

→ Commonly occuring terms are attenuated

→ It is a measure of informativeness

→ Reduce the tf weight of a term by a factor that grows with its collection frequency

→ If a term appears in all the documents then IDF is zero. This implies that term is not very important

TF-IDF

Composition of TF-IDF produces a composite scaling for each term in the document.

$$tf\text{-}idf_{t,d} = tf_{td} \times idf_{t,d}$$

→ The value is high t occurs many times within a few documents

→ The value is very low when a term appears in all documents.

Spiral

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

IDF of a term t = $\log_{10}\left(\frac{\text{total no. of documents}}{\text{count of doc with term t}}\right)$

Ex:- Consider a corpus with 100000 document. The word moon occurs in some documents (say, 100) with the following frequency

$TF_{d_1} = \frac{20}{427}$ $TF_{d_2} = \frac{30}{250}$ ..... $TF_{d_{1000}} = \frac{20}{1000}$

Total no. of words in corpus = 100000

$\therefore IDF_{d_1} = \log_{10}\left(\frac{100000}{100}\right)$

$TF_{d_1} * IDF = 0.141$

Document Ranking using TF - IDF

Document Name	tf	tf-idf	Rank
d_1	0.047	0.14	3
d_2	0.012	0.36	1
d_3	0.08	0.24	2
d_4	0.04	0.12	4
d_{1000}	0.02	0.06	5

Spiral

ZIPF'S LAW

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

Zipf's law states that for a given corpus the frequency of any word is inversely proportional to its rank in the term frequency table

$$f(r) \propto \frac{1}{r^{\alpha}} \Rightarrow f(r) \cdot r = k$$

where $\alpha \approx 1$, r is the frequency rank of a word and $f(r)$ is the frequency in the corpus. The most frequent word will have the value 1, the word ranked second in the frequency will have $\frac{1}{2^{\alpha}}$, the word ranked third in the frequency will have $\frac{1}{3^{\alpha}}$ etc

⇒ The emperical law models the frequency distribution of words in languages. This distribution is observed across several languages with a language corpus. It may not be good enough to fit the frequency linearly, but enough to approximately model word frequencies.

Mandelbrot Approximation

$$f(r) \propto \frac{1}{(r+\beta)^{\alpha}}$$ where $\alpha \approx 1$ $\beta \approx 2.7$

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

Heap's law

This is used to determine the number of unique terms M in a corpus given the total number of tokens

$$M \propto T^b$$
$$= kT^b$$

where $30 \leq k \leq 100$ & $b \approx 0.49$

According to this emperical law, the dictionary or the vocabulary size increases linearly with the total number of tokens/words in the corpus. It emphasizes the importance of the compression of dictionary