proschool
An IMS Initiative

Text Mining

Chapter

Overview

- Text mining is the process of extracting high quality information from text.
- It is a generalized term used to describe a wide range of technologies for analyzing and processing semi-structured and unstructured data.
- Information is derived from observing and finding out patterns and trends in the text.
- Typically text mining involves converting text into numbers to apply machine learning and find out results.

Applications of Text Mining

- **Document clustering** - grouping text/paragraphs/documents based on clustering methods.

- **Document classification** - categorizing text/paragraphs/documents based on data mining methods using labeled dataset.

- **Natural Language Processing** - sentiment analysis, topic modelling.

Examples of Applications

- Sentiment analysis of people during elections
- Chatbots answering commonly asked questions
- Screening job applicants based on key words present in Resume
- Predict customer churn based on reviews for a product/company

Text Data Preprocessing

- Before text can be used effectively, it needs to be cleaned into a standard format. For that we need to follow some specific steps to get "clean" text.

- Let's take an example.

If it looks like a dog, runs like a dog, barks like a dog, then it probably is a dog.

Text Data Preprocessing

- Tokenization
- Stop Words
- Stemming
- Lemmatization
- Rules for text preprocessing

Tokenization

- Process of splitting text into separate words/entities.

- Example: If it looks like a dog, runs like a dog, barks like a dog, then it probably is a dog.

The above sentence is tokenized as:

['If','it','looks','like','a','dog,','runs','like','a','dog,','barks','like','a','dog,','then','it','probably','is','a','dog.']

Stop Words

- Stop words are words which are very common and which do not produce any meaningful insight or information to the user.
- Examples include "who", "what", "the", "where", "a", "it", etc.
- From previous example
  [looks like dog, runs like dog, barks like dog, probably dog]

Stemming

- Stemming is the process of bringing the root word for a particular word.

- For example, playing, plays, played all come from the root word "play".

- If we stem the example sentence, we get

```
[If  :  If], [it  :  it], [looks  :  look], [like  :  like], [a  :  a], [dog  :  dog], [,  :  ,],
[runs  :  run], [like  :  like], [a  :  a], [dog  :  dog], [,  :  ,],
[barks  :  bark], [like  :  like], [a  :  a], [dog  :  dog], [,  :  ,],
[then  :  then], [it  :  it], [probably  :  probabl],
[is  :  is], [a  :  a], [dog  :  dog], [.  :  .]
```

Lemmatization

- Lemmatization, unlike Stemming, reduces the inflected words properly ensuring that the root word belongs to the language.

- In Lemmatization root word is called Lemma.

- A lemma (plural lemmas or lemmata) is the canonical form, dictionary form, or citation form of a set of words.

- For example, runs, running, ran are all forms of the word run, therefore run is the lemma of all these words.

- From our example, in stemming, ***probably*** became ***probabl***, but when lemmatizing, it became an actual word ***probably***. This is the key difference.

More examples of Lemmatization

- am, are, is → be
- car, cars, car's, cars' → car
- “the boy's cars are different colors “

  becomes

  “the boy car be differ color”

Rules for text preprocessing

- Remove punctuation (, . “ ‘ ‘)
- Lemmatization/stemming
- Lower case (ABcd to abcd)
- Remove numbers, symbols (1234, @&)
- Remove stopwords (to, a, the)
- Strip extra whitespace
- Remove common words (eg. Book)
- Remove rare words (eg. confluence)
- Spelling correction
- Expand abbreviations

Word Cloud

proschool
An ims Initiative

GOVERNMENT
FACTOR
MANAGE
ONLINE
DAY
WORLDWIDE
SYSTEMS
TECHNOLOGIES
MASSIVE
DEFINITION
USING
RESULTS
SOCIAL
YEARS
PROCESS
FRAMEWORK
BETWEEN
STATISTICS
DETERMINE
WELL
SET
QUERIES
HIGH
BUSINESS
EFFECTIVE
VISUALIZATION
DEPARTMENT
USERS
DRIVE
SAN
NAS
MAKING
INTEGRATION
DATABASES
SIZE
ANALYTICS
EXABYTES
AMOUNT
REAL
REPLICATION
VOLUME
PROCESSING
SIMULATIONS
HIGHER
RATE
STRUCTURE
WORK
HANDLE
SOFTWARE
ALGORITHMS
COMPANIES
PETABYTES
APPLICATIONS
INFRASTRUCTURE
SOURCES
MANAGEMENT
ANALYSIS
TERMS
HAND
LIMITS
CHALLENGES
LARGE
BIG
CAPACITY
ONE
INFORMATION
PRIVATE
RESEARCH
YEAR
COMPLEX
SUCCESSFUL
CAPABILITIES
ANNOUNCED
STORAGE
PRIVACY
INTERNATIONAL
IMPLEMENTATION
BECOMES
LEARNING
LESS
HUGE
NEEDED
USES
PAST
SECOND
APPROACHES
EVERY
WORLD
INTERNET
COMPARED
DATA
OUT
VARIOUS
MILLION
SETS
DEVELOPMENT
NATIONAL
INTELLIGENCE
BASED
TIMES
SHARED
INCLUDE
NETWORKS
TECHNOLOGY
MANY
SEARCH
HYPOTHESIS
SCIENCE
PROCESSES
ARCHITECTURE
APPROACH
FLOW
END
FUTURE
CRITIQUES
BILLION
NEW
PARADIGM
FUNDING
MOST
NEARLY
MAPREDUCE
INITIATIVE
ECONOMIC
DISTRIBUTED
PER
RECORDS
COST
INSIGHT
TOOLS
CENTER
PARALLEL
USE
PROJECT
RELEVANT
STATE
CRITIQUE
TERABYTES
MARKET
VARIETY
ANNUAL
USED
PEOPLE
TRAFFIC
CURRENTLY
RELATED
TIME
SENSOR
WOULD
NOW

What is word cloud?

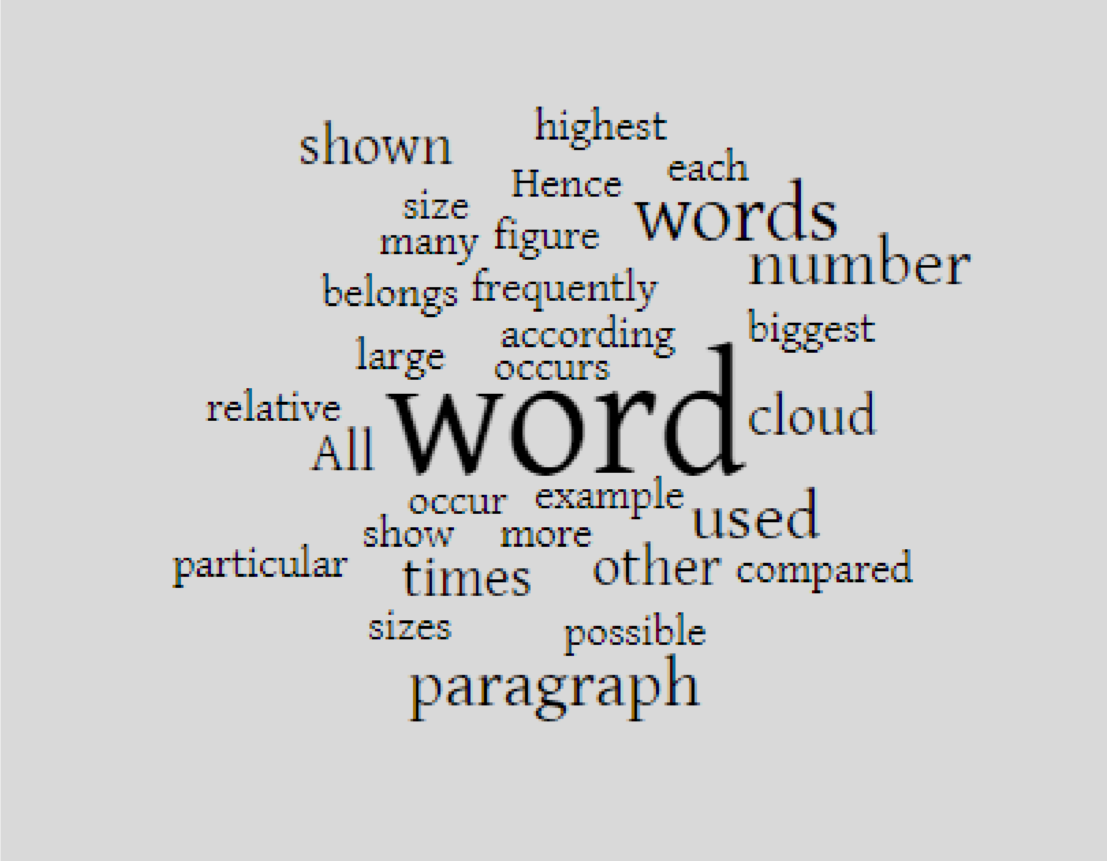
- This is a form of visualization that is used to show/depict how often each word occurs in the given text.
- Thus, the bigger the word in the visualization, the more number of times it occurs.

How is a word cloud created?

- First the text goes through all the preprocessing steps.
- Then count the frequency of each word in the sentence.
- Then each word is plotted, with the size of each word proportional to the frequency of occurrence.

Word Cloud Example

- This is an example paragraph for a word cloud. A word cloud is a figure used to show the words which occur more frequently compared to other words. If the number of words of a particular word is a large number, then the word is shown with the highest size possible. All sizes of all words are relative to each other according to the number of times the word occurs in the paragraph the word belongs to. So in this paragraph, we have used the word “word” for many times. Hence, the word “word” is shown as the biggest word used.



Sentiment Analysis

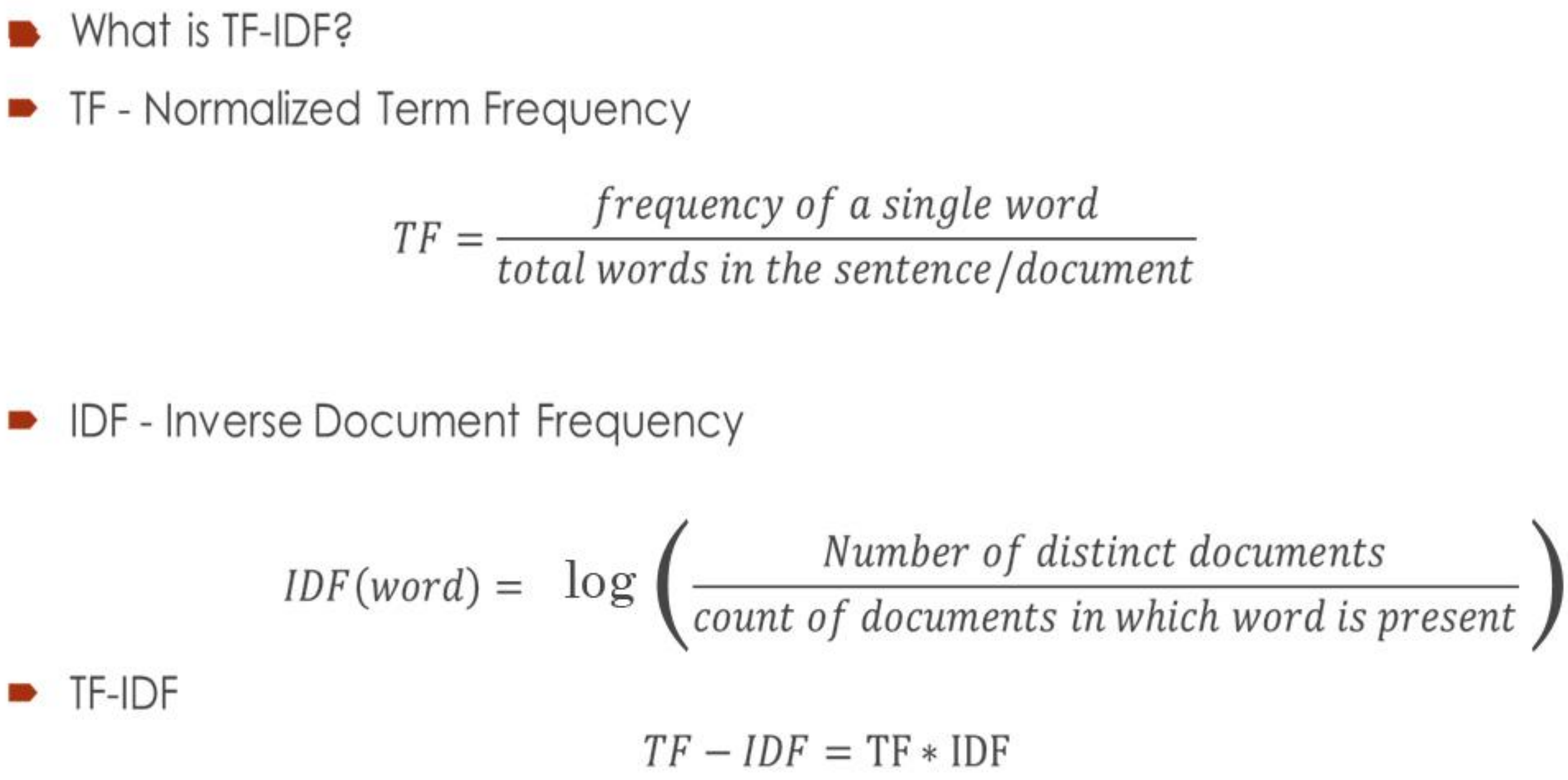
- It is the process of finding out the sentiment of a particular sentence/paragraph/text.

- Examples
  - I loved this restaurant's fried rice (positive sentiment)
  - This mayor has done a really good job in his tenure (positive sentiment)
  - The product I have received is red in colour (neutral sentiment)
  - I had a bad experience during this cab ride (negative sentiment)

Sentiment Analysis – Data Sources

- Twitter tweets
- Facebooks comments
- Online reviews
- Sms/emails/messages
- Novels, books
- Any other text based sources like books/newspapers/magazines/etc

TF-IDF



- What is TF-IDF?
- TF - Normalized Term Frequency

$$TF = \frac{frequency\ of\ a\ single\ word}{total\ words\ in\ the\ sentence/document}$$

- IDF - Inverse Document Frequency

$$IDF(word) = \log\left(\frac{Number\ of\ distinct\ documents}{count\ of\ documents\ in\ which\ word\ is\ present}\right)$$

- TF-IDF

$$TF - IDF = \mathrm{TF} * \mathrm{IDF}$$

Example of TF-IDF

- Document 1 - Ram studies about computers in the lab
- Document 2 – Ram studies at Delhi
- Document 3 - Computer scientists work on data with computers

$$TF(word) = frequency\ of\ word\ in\ the\ document$$

$$TF(Ram) = 1$$

Thus, TF for all the documents is:

	Ram	study	about	computer	Delhi	scientist	work	data
Doc1	1	1	1	2	0	0	0	0
Doc2	1	1	0	0	1	0	0	0
Doc3	0	0	0	3	0	1	1	1

Example of TF-IDF (2/4)

- Document 1 - Ram studies about computers in the lab
- Document 2 – Ram studies at Delhi
- Document 3 - Computer scientists work on data with computers

$$TF\ (normalized)(eg.Ram) = \frac{Number\ of\ occurences\ of\ word\ (eg\ Ram)}{Total\ number\ of\ words\ in\ the\ document\ (doc\ 1)} = \frac{1}{8} = 0.125$$

TF (Normalized)

	Ram	study	about	computer	Delhi	scientist	work	data
Doc1	0.125	0.125	0.125	0.25	0	0	0	0
Doc2	0.25	0.25	0	0	0.25	0	0	0
Doc3	0	0	0	0.375	0	0.125	0.125	0.125

Example of TF-IDF (3/4)

- Document 1 - Ram studies about computers in the lab
- Document 2 – Ram studies at Delhi
- Document 3 - Computer scientists work on data with computers

$$TF - IDF = TF(term) * IDF(term)$$

$$TF - IDF\ (Ram) = 0.125\ * 0.405 = 0.0506$$

Thus, TF-IDF for the entire table is the following.

	Ram	study	about	Computer	Delhi	scientist	work	data
No of documents in which word (T) occurs (N_T)	2	2	1	3	1	1	1	1
IDF = LOG(N/N_T)	0.405	0.405	1.098	0	1.098	1.098	1.098	1.098

Example of TF-IDF (4/4)

- Document 1 - Ram studies about computers in the computer lab
- Document 2 – Ram studies at Delhi
- Document 3 - Computer scientists work on computer data with computers

$$TF - IDF = TF(term) * IDF(term)$$

$$TF - IDF\ (Ram) = 0.125\ * 0.405 = 0.0506$$

Thus, TF-IDF for the entire table is the following.

	Ram	Study	about	computer	Delhi	scientist	work	data
Doc1	0.05	0.05	0.137	0	0	0	0	0
Doc2	0.10	0.10	0	0	0.274	0	0	0
Doc3	0	0	0	0	0	0.137	0.137	0.137

Thank You.

proschool
An IMS Initiative