NLP : Natural Language Processing

SECTION ONE

1. What is NLP?
2. Where is NLP used?
3. Challenges in understanding
natural language text
4. NLP Workflow Description

What is NLP ?

Natural language processing (NLP) is a branch of artificial intelligence that helps computers understand, interpret and manipulate human language. NLP draws from many disciplines, including computer science and computational linguistics, in its pursuit to fill the gap between human communication and computer understanding.

● Subfield of computer science and artificial intelligence

● Allows humans to bypass programming languages to speak to computers and instead use normal human speech

● Applications: text classification, machine translation, sentiment analysis

● Our devices nowadays: Apple's Siri, Amazon's Alexa, and Gmail's spam filter

Relationship with DS/AI/ML

- Machine learning can help NLP powered systems adjust actions according to the historical context and patterns it picks up in a conversation.

- NLP technology is human-like in the sense that more conversation can lead to better comprehension

Where Is NLP Used?

***1**.Machine Translation*
***2**.Text Classification*

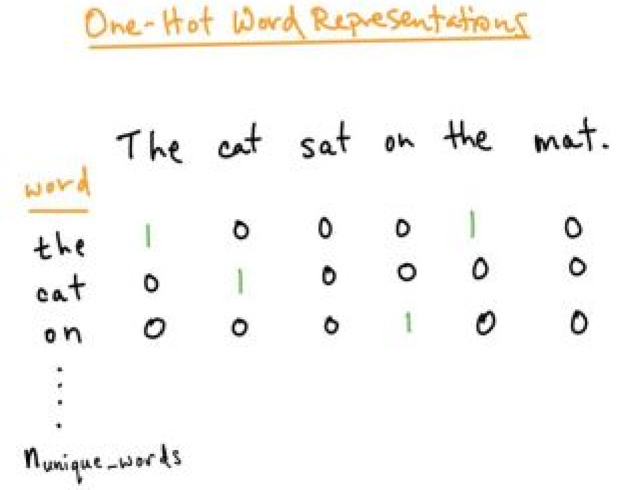
3.*Sentiment Analysis*
4.*NLP in Our Everyday Lives*

- *Email assistant*
- *Ask Siri*
- *Answering questions*
- *5 Amazing Applications:*
  - *Livox app*
  - *SignAll*
  - *Google Translate*
  - *Aircraft maintenance*
  - *Predictive police work*

SECTION TWO

1.Feature Engineering Overview
2. TF-IDF
3. Bag-of-Words
4.Implement with Google Colab(Sentiment Analysis)

1.Feature Engineering Overview



One-Hot Word Representations

word	The	cat	sat	on	the	mat.
the	1	0	0	0	1	0
cat	0	1	0	0	0	0
on	0	0	0	1	0	0
⋮						
n_{unique_words}						

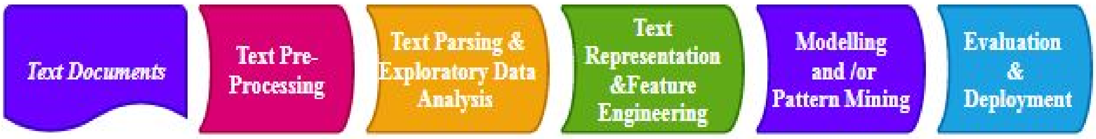
ML algorithms cannot work on the raw text directly

- Algorithms can only process numeric representation of an actual text
- FE techniques used to convert text into a matrix (or vector)
- Popular methods of feature extraction are : Bag-of-Words, TF-IDF

Techniques to Understand Text

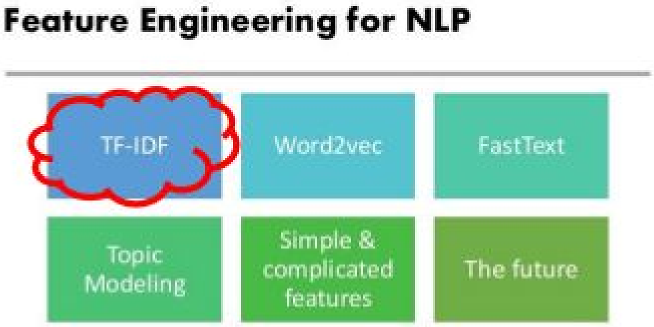
- Shallow Parsing or Chunking
- Named Entity Recognition (NER)
- N-Grams

NLP WORKFLOW


Text Documents
Text Pre-Processing
Text Parsing & Exploratory Data Analysis
Text Representation &Feature Engineering
Modelling and /or Pattern Mining
Evaluation & Deployment

1.Text Documents
2.Text Preprocessing
3.Text Parsing & Exploratory Data Analysis
4.Text Representation & Feature Engineering
5.Modelling and /or Pattern Mining
6.Evaluation Deployment

2.TF-IDF



In the TF-IDF Word bag approach, each word has the same weight. The idea behind the TF-IDF approach is that fewer words in all documents and more in individual documents contribute more to the classification.

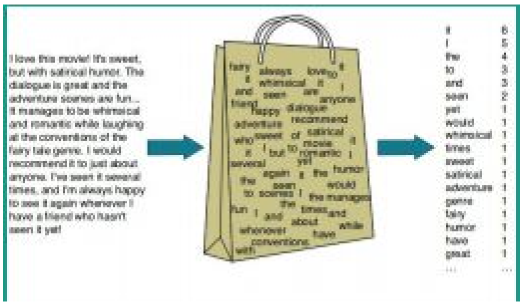
TF-IDF is a combination of two terms. Term frequency and Reverse Document frequency.

Goal of TF-IDF

- Text documents → vector models, based on word occurrence (without considering the exact ordering)
- Given: dataset of N text documents, TF and IDF defined as...
  - TF: count of a term “t” in a document “d”
  - IDF: logarithm of ratio of total documents (d) in the entire corpus and number of documents (d) containing the term “t”

- Together: relative importance of a term in a corpus

3.Bag-of-Words



I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet!

Why a “bag”?

- Any information about the order or structure of words is discarded

- Only concerned with whether known words occur in the document, not where
- General intuition: similar documents have similar content

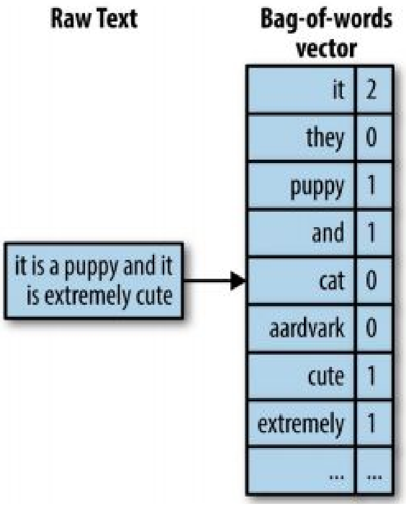
Bag-of-Words Model

- Transform tokens into a set of features

- Used in document classification, each word used as a feature for training the classifier

- **Ex:** review-based sentiment analysis

- Each text document = numeric vector

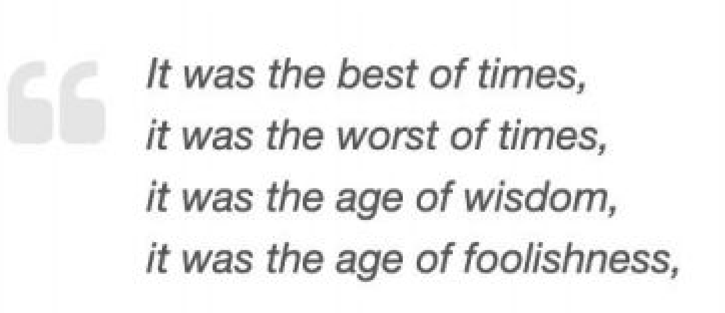


- Each dimension = specific word (from the corpus)

- Value = frequency in document, occurrence (1/0), or weighted value

Bag-of-Words Model: Simple Example

“A Tale of Two Cities”



> *It was the best of times,*
> *it was the worst of times,*
> *it was the age of wisdom,*
> *it was the age of foolishness,*

Charles Dickens

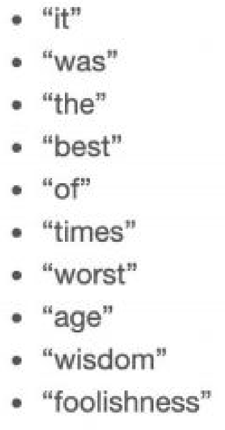
Step 1: Collect Data

- Data: first few lines of text from book
- Each line treated as a separate “document”
- Four lines = entire corpus of documents

Step 2: Design the Vocabulary

- Construct list of all words in the mode’s vocabulary
- Only the unique words (note: we ignore case and punctuation)

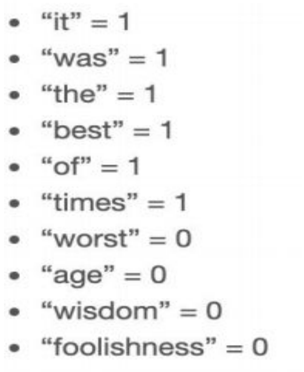
- Our list: vocab of 10 words, from corpus of 24 words



- “it”
- “was”
- “the”
- “best”
- “of”
- “times”
- “worst”
- “age”
- “wisdom”
- “foolishness”

Step 3: Create Document Vectors

- Score the words in each document
- Document of free text → vector
- Vector = input or output of model
- Vocab of 10 words → fixed-length vector
- Use 0/1 binary scoring



- “it” = 1
- “was” = 1
- “the” = 1
- “best” = 1
- “of” = 1
- “times” = 1
- “worst” = 0
- “age” = 0
- “wisdom” = 0
- “foolishness” = 0

Managing Vocabulary

- Vocab size Vector representation of documents
- Issue of sparse vector/representation
    - Memory, computational resources
- Decrease vocab size → text cleansing!
    - Ignore case
    - Ignore punctuation
    - Ignore stop words
    - Fix misspelled words
    - Stemming

Sources

1.https://www.kdnuggets.com/2019/10/introduction-natural-language-processing.html

2.https://www.slideshare.net/BillLiu31/feature-engineering-for-nlp

3.https://towardsdatascience.com/a-practitioners-guide-to-natural-language-processing-part-i-processing-understanding-text-9f4abfd13e72

4. https://www.slideshare.net/BillLiu31/feature-engineering-for-nlp

5.https://machinelearningmastery.com/gentle-introduction-bag-words-model/

6.https://www.sas.com/en_us/insights/analytics/what-is-natural-language-processing-nlp.html

7.https://stackabuse.com/python-for-nlp-sentiment-analysis-with-scikit-learn/