Natural Language Processing

CDAC - workshop

Simran K

Natural Language Processing (NLP) is a branch of AI that helps computers to understand, interpret and manipulate human language.

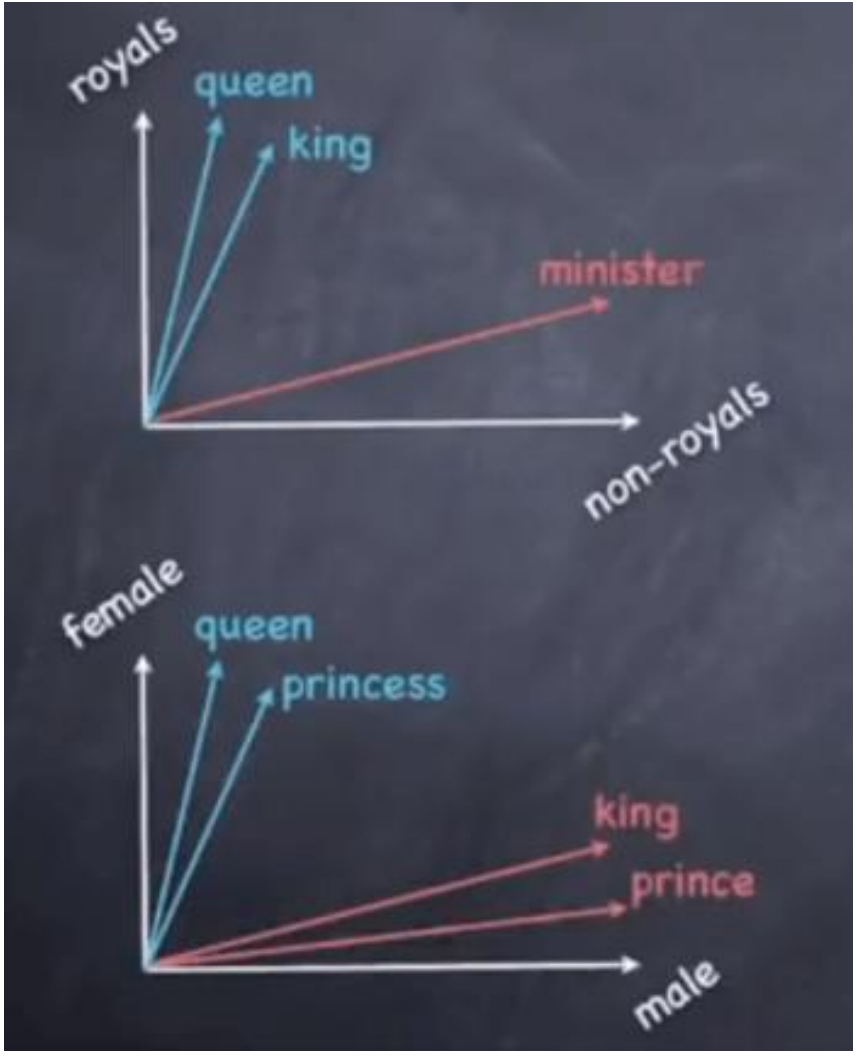
NLP is a way of computers to analyze, understand and derive meaning from a human languages such as English, Spanish, Hindi, etc.

How NLP works?

Man is to woman as king is to _____?

Meaning (king) – meaning (man) + meaning (woman)=?

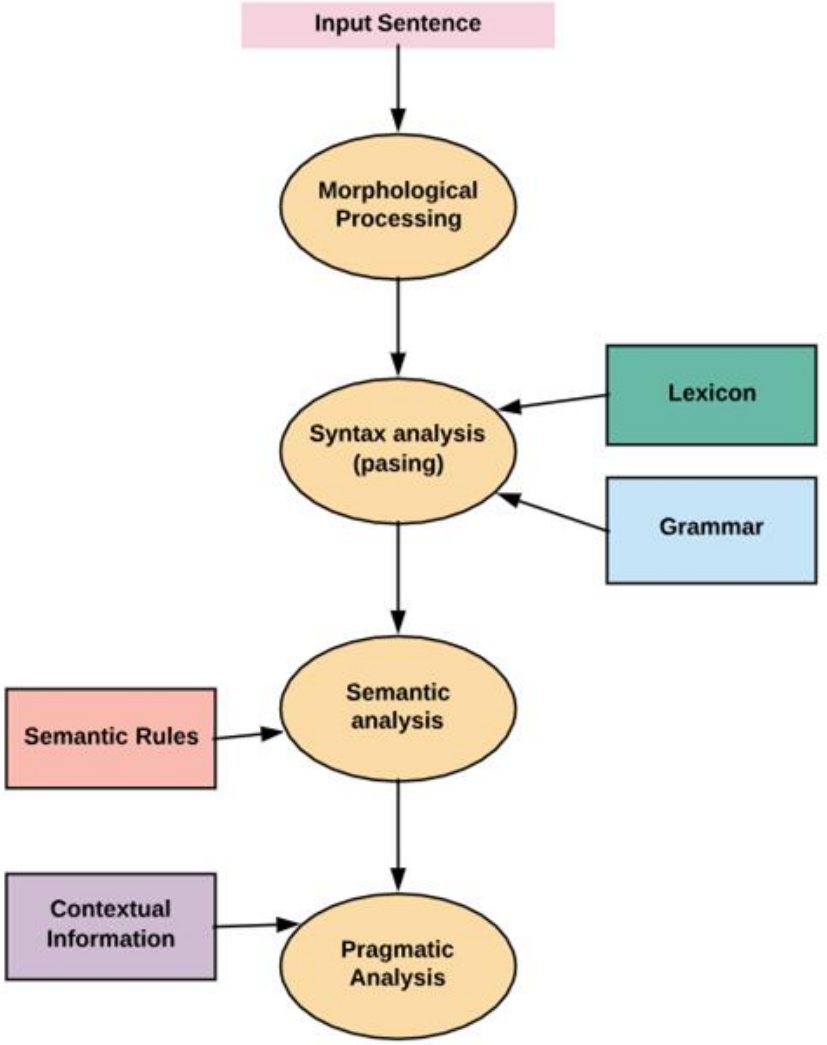
The answer is- queen



Meaning (king) – meaning (man) + meaning (woman)=?

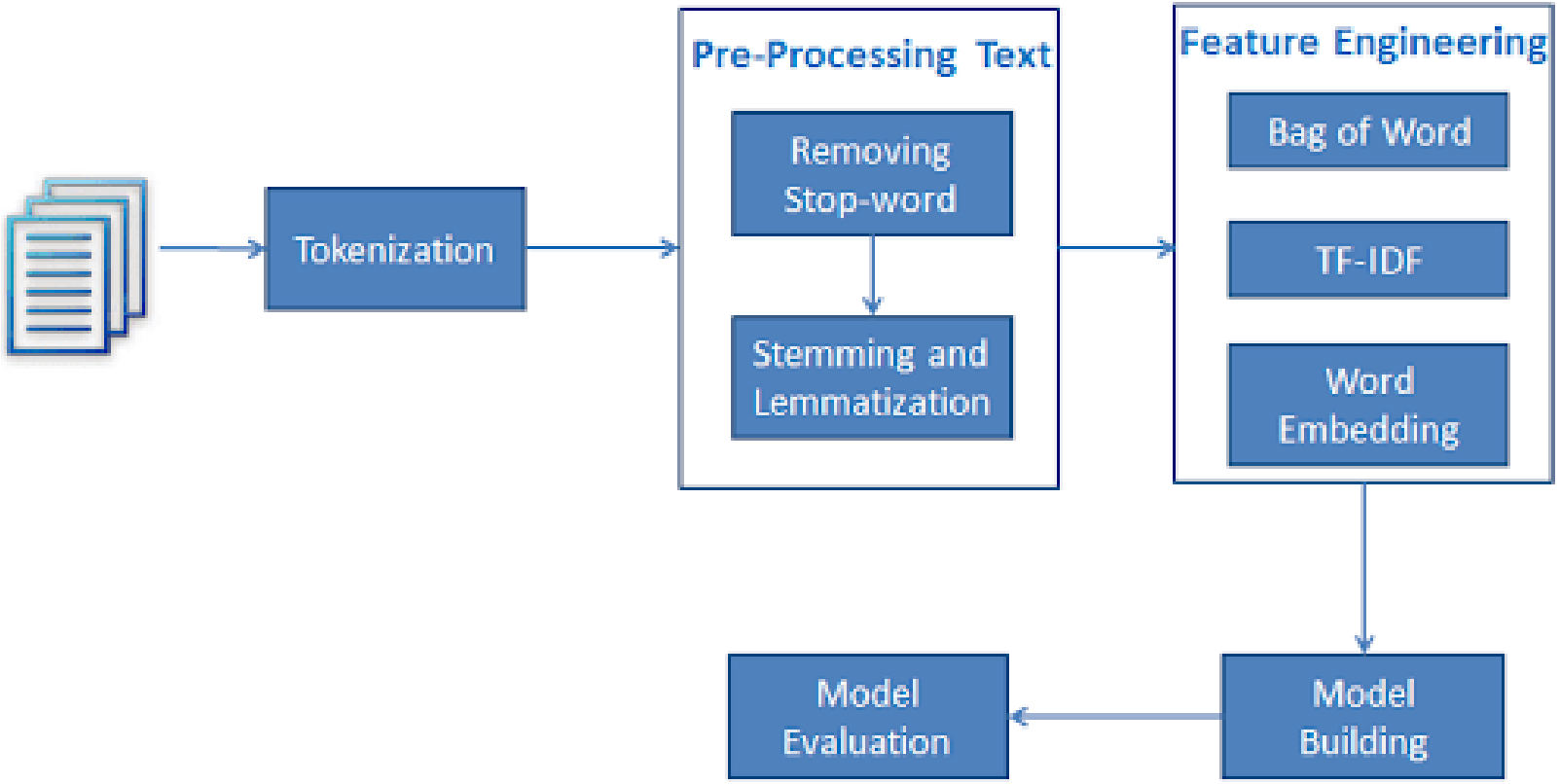
Vector (king) – vector (man) + vector (woman) = vector(?)

Main Component of **NLP**



Applications of NLP

- Information Retrieval and Web Search
- Grammar Correction
- Text Summarization
- Machine Translation
- Intelligent Chatbot


Tokenization
Pre-Processing Text
Removing Stop-word
Stemming and Lemmatization
Feature Engineering
Bag of Word
TF-IDF
Word Embedding
Model Building
Model Evaluation

NLTK

Text preprocessing

- Tokenization
- PoS (Part-of-Speech) Tagging
- Chunking
- Stemming (general operation)
- Lemmatization (intelligent operation)
- Stopwords

Tokenization

- The process by which big quantity of text is divided into smaller parts called **tokens**
- Example : “ CDAC workshop is on 5th and 6th july.”

  Output: [‘CDAC’ , ’workshop’ , ‘is’ , ’on’ , ’5th’ , ‘and’ , ’6th’ , ’july’ , ’.’]

PoS Tagging

- Parts of speech Tagging is responsible for reading the text in a language and assigning some specific token (Parts of Speech) to each word.
- Example : “CDAC workshop is on 5th and 6th july.”

  Output: [('CDAC', 'NNP'), ('workshop', 'NN'), ('is', 'VBZ'), ('on', 'IN'), ('5th', 'CD'), ('and', 'CC'), ('6th', 'CD'), ('july.', 'NN')]

Chunking

- selecting the subsets of tokens
- Example : “learn machine learning and NLP. ”

  Output: S learn/JJ (S machine/NN learning/NN) and/CC NLP/NNP ./.

Stemming

- Stemming is a kind of normalization for words.
- Example : eat, eats, eating ,eaten

  Output: eat, eat, eat, eat

Lemmatization

- Intelligent operation
- Base dictionary word
- Example: “Studies”

  Output: Study

Stop words

- Noise in the text
- Words like is, the, a, an, this etc.

Feature Engineering

Text representation

It aims to numerically represent the unstructured text documents to make them mathematically computable.

- Identical text must have the same representation and distance of zero (maximum similarity).
- When we have multiple texts, t1, t2, and t3, we want to have the ability to say that t1 is more similar to t2 than t3.
- Similarity/Distance should express the semantic comparison between texts, and text length should have little effect.

Text representations

- One-hot encoding
- Bag-of-words (BoW)
- Term document matrix (TDM)
- Term-frequency inverse document frequency (TFIDF)
- Word embedding (Skip gram, CBOW)
- Doc2vec
- Keras embedding

One - hot encoding

- Performs “binarization” of the category

Example:

[[1],[2], [3]]

```
[[ 1.  0.  0.],
 [ 0.  1.  0.],
 [ 0.  0.  1.]]
```

- Sparse matrix
- Cannot represent any new words like 4

Bag-of-words (BOW)

- It converts text into the matrix of occurrence of words within a document.
- Simplest way - does counting.

Example: This is the first document. This document is the second document. And this is the third one. Is this the first document?

```
[[0 1 1 1 0 0 1 0 1]
 [0 2 0 1 0 1 1 0 1]
 [1 0 0 1 1 0 1 1 1]
 [0 1 1 1 0 0 1 0 1]]
```

- No information about the word or order of the word
- Cannot represent any new words

Term document matrix (TDM)

- Matrix of document and words by counting the occurrence of words in the given document.

Example:

This is the first document. This document is the second document. And this is the third one. Is this the first document?

```
[[0 1 1 1 0 0 1 0 1]
 [0 2 0 1 0 1 1 0 1]
 [1 0 0 1 1 0 1 1 1]
 [0 1 1 1 0 0 1 0 1]]
```

- **N-grams**: Unigram, bi-gram, tri-gram.
- Sequence of occurences of words.

Term-frequency inverse document frequency (TF-IDF)

- Most frequently used words are given most importance.
- IDF - measures the amount of information a given word provides across the document.

$$\text{idf(W)} = \log \frac{\#(\text{documents})}{\#(\text{documents containing word W})}$$

Example: This is a sample. This is another sample.

[[0. 0.57735027 0.57735027 0.57735027]
[**0.63009934** 0.44832087 0.44832087 0.44832087]]

Singular value decomposition (SVD)

- To convert sparse matrix into dense representation.
- Reduced dimensional representation.

$$\begin{bmatrix} & X & \end{bmatrix}_{m \times n} =$$

$$\begin{bmatrix} \uparrow & \uparrow & \cdots & \uparrow \\ u_1 & u_2 & \cdots & u_r \\ \downarrow & \downarrow & \cdots & \downarrow \end{bmatrix}_{m \times r} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \end{bmatrix}_{r \times r} \begin{bmatrix} \leftarrow & v_1 & \rightarrow \\ \leftarrow & v_2 & \rightarrow \\ \vdots & \vdots & \vdots \\ \leftarrow & v_r & \rightarrow \end{bmatrix}_{r \times n}$$

$$= \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \cdots + \sigma_r u_r v_r^T$$

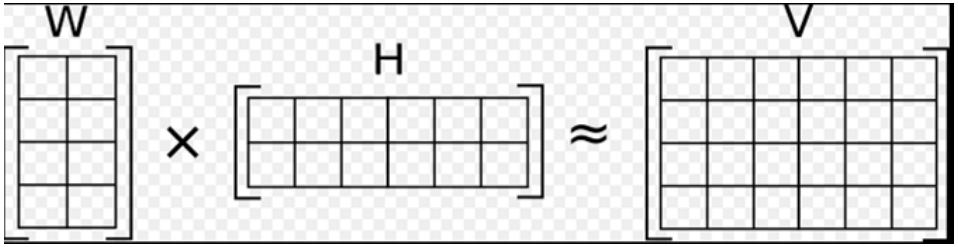
Non-negative matrix factorization (NMF)

Useful when there are many attributes and the attributes are ambiguous or have weak predictability.

By combining attributes, NMF can produce meaningful patterns, topics, or themes.

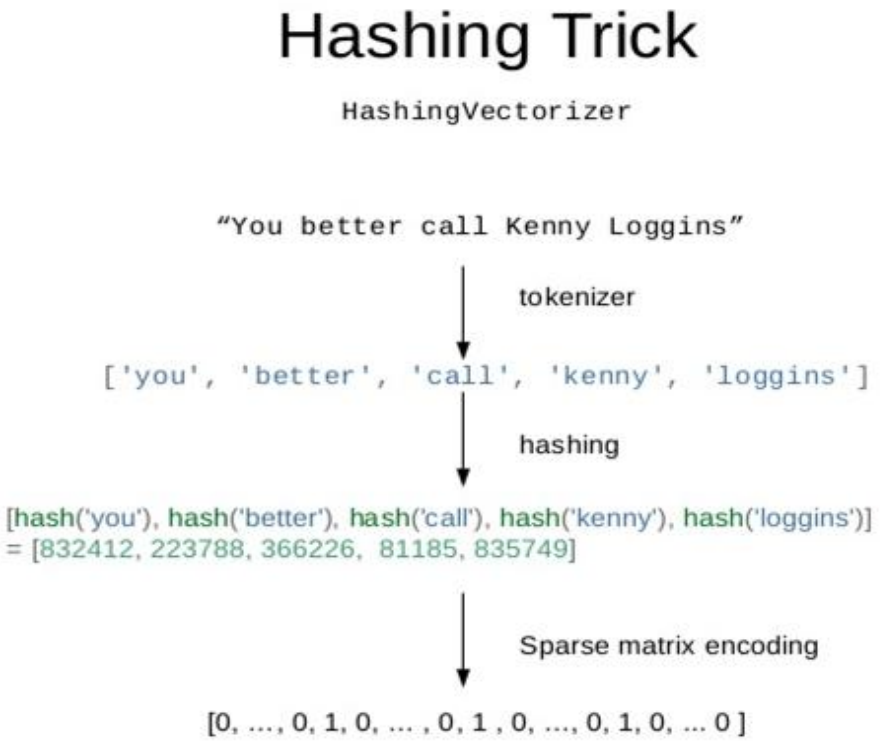
Each feature is a linear combination of the original attribute set

The coefficients of these linear combinations are non-negative.



Hashingvectorizer

- Very low memory scalable to large datasets
- No need to store a vocabulary dictionary in memory
- Faster
- Distinct tokens can be mapped to the same feature index.



Word2vec

- Convert words into dense vector representations.
- Created using two algorithms
  - Continuous Bag-of-Words model (**CBOW**)
  - **Skip-Gram** model

Continuous bag of words

- Given a sliding window and surrounding words. It tries to predict the center word.

Example : slide 2 _ _ [] _ _

Skip-Gram model

- One word to predict all surrounding words “context”.
- Slower than CBOW
- Considered more accurate

Example: Suppose features are Royal, Female, Male.

King = [0.9, 0.1, 0.9], Queen = [0.9, 0.9, 0.1]

Men = [0.1, 0.1, 0.9], Women = [0.1, 0.9, 0.1]

King - Man = [0.9, 0.1, 0.9] - [0.1, 0.1, 0.9] = [0.8, 0.0, 0.0]

[0.8, 0.0, 0.0] + women = [0.9, 0.9, 0.1] (Queen)

FastText

- Each word is represented as a bag of character n-grams in addition to the word itself.
- Preserves the meaning of shorter words .
- N-grams can be controlled.

Example: Matter

n = 3,

Character 3-gram is:

<ma, mat, att, tte, ter, er>

word <mat> is part of the vocabulary

Doc2vec

- Suppose the documents are D1, D2, D3
- D1: 3 sentences D2: 5 sentences D3: 10 sentences.
- So zero padding is used at the end of D1 and D2 to make the sizes same.
- Better option is Doc2vec
- It converts documents into vectors
- Decreases the number of zeros to be padded.

Keras embedding

- Keras offers an Embedding layer that can be used for neural networks on text data.
- Each word is represented by a unique integer
- The Embedding layer
  - Initialized with random weights and will learn an embedding for all of the words in the training dataset.
  - Defined as the first hidden layer of a network
- Flexible layer
  - Used alone to learn a word embedding - saved - Another model
  - Used as part of a deep learning model where the embedding is learned along with the model itself.
  - Used to load a pre-trained word embedding model (transfer learning).

Thank you