Natural Language Processing

NLP stands for Natural Language Processing. It is a branch of artificial intelligence (AI) that focuses on enabling computers to understand, interpret, and generate human language in a way that is both meaningful and useful. NLP involves the interaction between computers and humans through natural language, aiming to facilitate tasks such as language translation, sentiment analysis, text summarization, speech recognition, and more. NLP techniques allow computers to analyse and derive insights from large amounts of natural language data, opening up a wide range of applications across various industries, including customer service, healthcare, finance, and education.

Text Pre-processing:

Tokenization

Tokenization is the process of breaking down a text into smaller units called tokens. These tokens can be words, phrases, symbols, or any other meaningful units. Tokenization is a fundamental step in natural language processing (NLP) for various tasks such as text analysis, sentiment analysis, and machine translation.

Example: “Data science is a vast field.”

Tokens: [“Data”, “science”, “is”, “a”, “vast”, “field”, “.”]

Stop word removal

Stop word removal is a text pre-processing technique used in natural language processing (NLP) to filter out common words that do not carry significant meaning or contribute much to the understanding of the text. These words, known as stop words, include frequently occurring terms such as “the”, “is”, “and”, “in”, “of”, etc.

Example: “Data science is a vast field.”

Output: “Data science vast field.”

Text processing:

Text processing in NLP involves understanding and analysing human language using computers. When you feed a computer text, it breaks it down into smaller parts to make useful meaning out of it.

methods:

Word Frequency

It refers to how often words occur within a given piece of text or a corpus (a collection of texts). In simple terms, it's like counting how many times each word appears in a document, paragraph, or any chunk of text. By analysing word frequency, you can gain insights into the importance, relevance, or characteristics of different words within the text.

Example:

He is a good boy. She is a good girl. (stopwords are removed)

he-1; good-2; boy-1; she-1; girl-1

Collocation

These are pairs or groups of words that often co-occur together in a text more frequently than would be expected by chance. The most frequent kinds of collocations in text are

- Bigrams: Frequent two-word combinations
- Trigrams: Frequent three-word combinations
- Quadgrams: Frequent four-word combinations

Example:

Data science involves data handling and analyses. ML, DL and AI combines to form data science.

Bigram – data science

Concordance

This is a tool that shows every occurrence of a given word along with its surrounding context.

Example:

This is a good book. I love reading books. It's very much helpful in keeping peace of mind.

Concordance of word 'book' - This is a good book. I love reading books.

TF-IDF (Term Frequency – Inverse Document Frequency)

This is to determine the importance of a word in a document relative to a collection of documents. TF measures how frequently a word appears in a document. IDF measures the significance of a word across a collection of documents.

TF = (number of repetition of words in a sentence) / (number of words in a sentence)

IDF = log((number of sentence) / (number of sentences containing words))

TF-IDF = IF*IDF

Example:

He is a good boy. She is a girl. Both boy and girl are good. (Let's assume pronoun and stopwords are removed)

Term Frequency (TF):

For sen1:

good = 1/2 = 0.5

boy = 1/2 = 0.5

For sen2:

good = 1/2 = 0.5

girl = 1/2 = 0.5

For sen3:

boy = 1/3 = 0.33

girl = 1/3 = 0.33

good = 1/3 = 0.33

Inverse Document Frequency (IDF):

good = log(3/3) = 0

boy = log(3/2) = 0.18

girl = log(3/2) = 0.18

TF-IDF:

For sen1:

good = 0.5 * 0 = 0

boy = 0.5 * 0.18 = 0.09

For sen2:

good = 0.5 * 0 = 0

girl = 0.5 * 0.18 = 0.09

For sen3:

good = 0.33 * 0 = 0

boy = 0.33 * 0.18 = 0.06

girl = 0.33 * 0.18 = 0.06

In this example, “boy” and “girl” have slightly higher TF-IDF scores than “good” because they appear in fewer documents(sentences), indicating they may be more significant in distinguishing between the documents.

Text Summarization

Text summarization is the process of condensing a piece of text while retaining its key information and main points. This helps in removing complexity in sentence and find its meaning in easier way. There are two types.

Extractive Summarization

Using an extractive approach, we summarize our text on the basis of simple and traditional algorithms. For example, when we want to summarize our text on the basis of the frequency method, we store all the important words and frequency of all those words in the dictionary. On the basis of high frequency words, we store the sentences containing that word in our final summary. This means the words which are in our summary confirm that they are part of the given text.

Abstractive Summarization.

An abstractive approach is more advanced. On the basis of time requirements, we exchange some sentences for smaller sentences with the same semantic approaches of our text data.

Text classification

Text classification is the process of categorizing text documents into predefined categories or classes based on their content. Text classification includes several subdivisions, including topic modelling and sentiment analysis. Various applications are spam detection, sentiment analysis, topic labelling, and content organization.

Topic modelling

It is a technique that automatically identifies and groups similar words and phrases in a text.

Sentiment analysis

It is a technique that determines the sentiment or emotion expressed (positive, negative or neutral) in a piece of text.

Keyword extraction

This mainly focuses on extracting the most important words or phrases from a text to represent its main topics or themes.

Example:

Data science is a vast field which involves programming, mathematics, data visualization, machine learning, deep learning and artificial intelligence.

Keyword extraction using tokenization (converting sentence into separate words) and removing stopwords.

['data', 'science', 'vast', 'field', 'involves', 'programming', 'mathematics', 'data', 'visualization', 'machine', 'learning', 'deep', 'learning', 'artificial', 'intelligence']

Lemmatization and stemming

Lemmatization is the process of getting root/base word in a dictionary form (whether root/base words should be present in dictionary)

Example: “Scientific studies show that the Earth is warming”

Lemmatized – “Scientific study show that the Earth is warm”

Stemming is a is the process of getting root/base word which involves stems or removes last few characters from a word. It often leading to incorrect meanings and spelling

Example: “Scientific studies show that the Earth is warming”

Stemmed - “Scientif studi show that the Earth is warm”

Deep Learning

Deep learning is a subset of machine learning that focuses on learning data representations through the use of neural networks with multiple layers. It is inspired by the structure and function of the human brain, where each layer of neurons processes different aspects of the input data. Deep learning algorithms can automatically learn to extract features from raw data, without the need for manual feature engineering.

1. **Neural Networks**: Neural networks are the foundation of deep learning. They consist of interconnected layers of artificial neurons (nodes) that process input data and generate output. Each connection between neurons has an associated weight, which determines the strength of the connection. Neural networks can have different architectures, including feedforward networks, convolutional neural networks (CNNs), recurrent neural networks (RNNs), and more.
2. **Activation Functions**: Activation functions introduce non-linearity into neural networks, allowing them to learn complex patterns in data. Common activation functions include sigmoid, tanh, ReLU (Rectified Linear Unit), and softmax.
3. **Loss Functions**: Loss functions measure the difference between the predicted output of a neural network and the true target output. They are used to quantify the model's performance during training and guide the optimization process. Common loss functions include mean squared error (MSE), cross-entropy loss, and hinge loss.
4. **Optimization Algorithms**: Optimization algorithms are used to update the weights of a neural network during training in order to minimize the loss function. Popular optimization algorithms include stochastic gradient descent (SGD), Adam, RMSprop, and AdaGrad.
5. **Backpropagation**: Backpropagation is a key algorithm used to train neural networks. It involves computing the gradient of the loss function with respect to the weights of the network, and then updating the weights in the opposite direction of the gradient to minimize the loss.
6. **Convolutional Neural Networks (CNNs)**: CNNs are specialized neural networks designed for processing grid-like data, such as images. They consist of convolutional layers that apply filters to input data, followed by pooling layers that down-sample the feature maps. CNNs are widely used for tasks like image classification, object detection, and image segmentation.
7. **Recurrent Neural Networks (RNNs)**: RNNs are neural networks designed for processing sequential data, such as text or time-series data. They have connections between neurons that form directed cycles, allowing them to capture temporal dependencies in the data. RNNs are commonly used for tasks like language modelling, machine translation, and speech recognition.

8. **Long Short-Term Memory (LSTM) Networks**: LSTM networks are a type of RNN that are designed to overcome the vanishing gradient problem, which occurs when training RNNs on long sequences of data. LSTMs use specialized units called memory cells, which can store information over long periods of time and selectively update or forget information.
9. **Autoencoders**: Autoencoders are neural networks used for unsupervised learning and dimensionality reduction. They consist of an encoder network that maps input data to a lower-dimensional representation (encoding), and a decoder network that reconstructs the original input data from the encoded representation. Autoencoders are used for tasks like data compression, denoising, and feature learning.
10. **Generative Adversarial Networks (GANs)**: GANs are a type of generative model consisting of two neural networks: a generator and a discriminator. The generator generates synthetic data samples, while the discriminator tries to distinguish between real and fake samples. GANs are used for tasks like image generation, style transfer, and data augmentation.