Page Rank Algorithm-NLP

Preeti AI · Follow
2 min read · Aug 3

First, you'll need to install the necessary libraries:

```
pip install nltk
pip install numpy

import nltk
import numpy as np
from nltk.tokenize import sent_tokenize, word_tokenize
from nltk.corpus import stopwords
from sklearn.metrics.pairwise import cosine_similarity
from sklearn.feature_extraction.text import CountVectorizer

nltk.download('punkt')
nltk.download('stopwords')

def text_summarization(text, num_sentences=2):
# Tokenize the text into sentences
sentences = sent_tokenize(text)
```

```
# Tokenize the text into words
word_tokens = [word_tokenize(sentence.lower()) for sentence in sentences]

# Remove stop words
stop_words = set(stopwords.words('english'))
filtered_tokens = [[word for word in words if word not in stop_words] for words in word_tokens]

# Create sentence vectors using CountVectorizer
vectorizer = CountVectorizer().fit_transform([' '.join(words) for words in filtered_tokens])
sentence_vectors = vectorizer.toarray()

# Calculate similarity matrix using cosine similarity
similarity_matrix = cosine_similarity(sentence_vectors)

# Convert similarity matrix to graph and apply PageRank algorithm (TextRank)
damping_factor = 0.85
scores = np.ones(len(sentences))
for _ in range(100):
scores = (1 — damping_factor) + damping_factor * np.dot(similarity_matrix, scores)

# Sort the sentences based on scores and get the top num_sentences
ranked_sentences = [sentences[idx] for idx in np.argsort(scores)[-num_sentences:]]

return ' '.join(ranked_sentences)
```

```
# Example text for summarization
text = """
Natural language processing (NLP) is a field of artificial intelligence that focuses on the interaction between humans and computers using natural language. It involves the processing and analysis of large amounts of natural language data to extract meaningful insights and patterns. NLP has various applications, such as machine translation, sentiment analysis, speech recognition, and text summarization.

Text summarization is the process of generating a concise and coherent summary of a longer piece of text. There are different approaches to text summarization, including extractive and abstractive methods. Extractive summarization involves selecting and combining existing sentences from the text, while abstractive summarization involves generating new sentences to form the summary.

In this example, we will use an extractive text summarization technique based on the TextRank algorithm. TextRank is an unsupervised algorithm that applies the PageRank algorithm to sentences in a text document. It calculates the importance of each sentence based on its similarity to other sentences in the document.

Let's implement text summarization using NLP and the TextRank algorithm.
"""

# Perform text summarization
summary = text_summarization(text, num_sentences=2)
print(summary)
```

Pagerank Algorithm | NLP | AI | Cosine Similarity | Text Summarization

More from the list: "NLP"

Curated by Himanshu Birla

 Jon Gi... in Towards Data ...

Characteristics of Word Embeddings

· 11 min read · Sep 4, 2021

 Jon Gi... in Towards Data ...

The Word2vec Hyperparameters

· 6 min read · Sep 3, 2021

 Jon Gi... in

The Word2ve

· 15 min rea

View list

Written by Preeti AI

Follow

14 Followers

Talks about #AI/ML/DL/Computer vision/Data Science/NLP. Connect via:: priyasi.ai7@gmail.com

More from Preeti AI

Preeti AI

Sentiment extraction using bilstm elmo crf

import tensorflow as tf import tensorflow_hub as hub import numpy as np from keras.layer...

1 min read · Sep 2

1

Preeti AI

LLM Project for beginners

This is a basic example on how to use OpenAI's GPT-3 (which is one of their...

1 min read · Jul 28

Preeti AI

VAE-Anomaly Detection

import numpy as np import pandas as pd import tensorflow as tf from tensorflow impo...

2 min read · Sep 2

Preeti AI

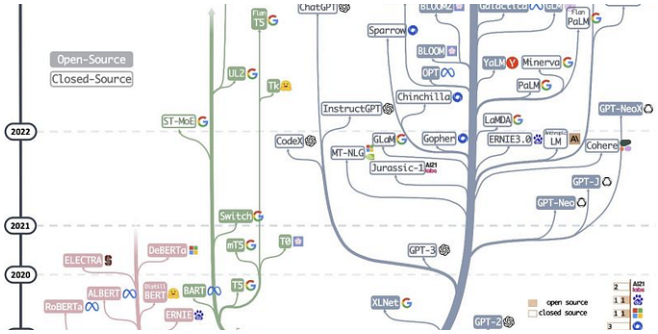
Medical Chatbot using GPT-3

We'll build a simple medical chatbot using the GPT-3 language model from OpenAI. Please...

2 min read · Jul 28

See all from Preeti AI

Recommended from Medium



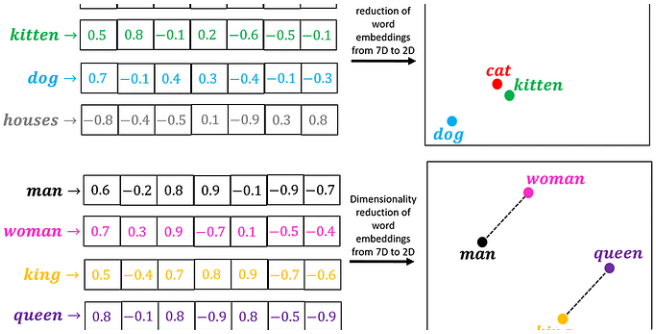
Haifeng Li

A Tutorial on LLM

Generative artificial intelligence (GenAI), especially ChatGPT, captures everyone's...

15 min read · Sep 14

372



Rahul Nair

Unleashing the Power of Word Embeddings: A Comprehensive...

Topics covered: 1) Introduction 2) Understanding embedding layers 3)...

5 min read · Jul 13

2

Lists

The New Chatbots: ChatGPT, Bard, and Beyond

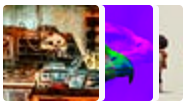
13 stories · 133 saves

Natural Language Processing

669 stories · 283 saves

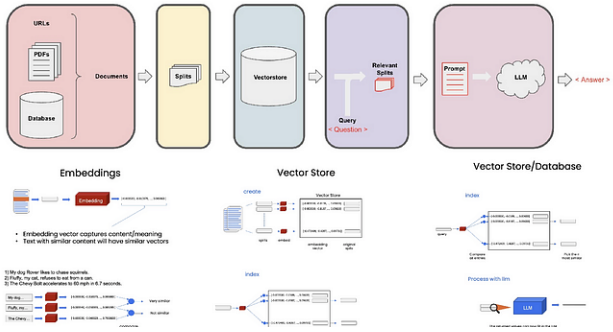
Generative AI Recommended Reading

52 stories · 274 saves



What is ChatGPT?

9 stories · 182 saves



TeeTracker

Chat with your PDF (Streamlit Demo)

Conversation with specific files

4 min read · Sep 15

56

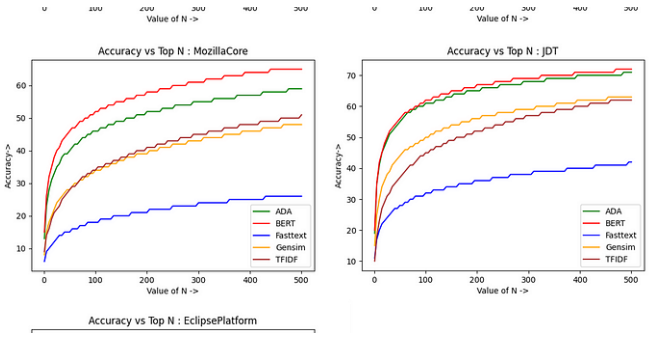
Lukas Niederhäuser

Exploratory Text Analysis of Swiss and German companies from...

The objective of this article is to gather and analyse text from Wikipedia pages that...

8 min read · Jul 5

39



Avinash Patil

Embeddings: BERT better than ChatGPT4?

In this study, we compared the effectiveness of semantic textual similarity methods for...

Seffa B

Named Entity Recognition with Transformers: Extracting Metadata

4 min read · Sep 19

3 min read · Jun 12

 3  1 

 7  

See more recommendations