KOCAELİ ÜNİVERSİTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

YAZILIM LAB.II – 1.PROJE:

Web İndeksleme Uygulaması

MÜRVET NUR ŞEN- ÜMMÜHAN TEPEBAŞ
190201097 – 180201088

ÖZET:

Bizden istenen; verilen bir URL'deki web sayfa içeriğine göre diğer birden fazla web sayfasını benzerlik bakımından indeksleyip sıralayan web tabanlı bir uygulama geliştirmektir.

Projemizi geliştirirken Python-Flask Web Framework' ü ve Javascript dilini ve Visual Studio Code geliştirme ortamını kullandık. Projemiz İngilizce dilinde web siteleri kullanılarak geliştirilmiştir.

GENEL BİLGİLER :

Proje dokümanında belirtilen "Aşama 1: Sayfada geçen kelimelerin frekanslarını hesaplama" adımı için BeautifulSoup modulü kullanılmıştır. Http requests ataması için Jquery- Axios kullanılmıştır.

Web scraping işlemlerinde Python – Natural Language Toolkit kullanılmıştır. NLTK kütüphanesi için detaylı bilgiler şöyledir:

NLTK, doğal dil araç takımı anlamına gelir. Natural Language Toolkit; insan dili verileriyle çalışmak için Python programlama dili ile geliştirilmiş ve geliştirilmekte olan 50'nin üzerinde derleme(corpus) ve sözcük kaynağı(lexical resources) ile oluşturulmuş açık kaynaklı bir kütüphanedir.

NLTK, bir veri setinin ön işlemesini(preprocessing), yani veriyi makinenin anlayacağı hale getireceğimiz zaman bizi gereksiz kelimelerle uğraştırmaktan da kurtarıyor. Biz de bu özelliği sayfalardaki (İngilizce dilindeki) bağlaç ve gereksiz kelimeleri çıkarmak için kullandık .

http://127.0.0.1:5000/frekans Sayfası Aşama 1 için verilen sitedeki tüm kelimeleri frekansları ile birlikte başarılı olarak döndürmektedir.

"Aşama 2: Anahtar kelime çıkarma ve Aşama 3: Benzerlik skorlaması" adımlarında girilen her iki Url için 15 anahtar kelimeyi frekansları ile birlikte döndürmektedir. Ve Aşama 3 için benzerlik oranını her iki Url'de dönen aynı anahtar kelimelerin frekanslarının, toplam anahtar kelime frekansına yüzdesi ile hesapladık.

http://127.0.0.1:5000/scoring Sayfasında Bootstrap progress bar kullanarak yüzdeyi ve anahtar kelimeleri başarılı bir şekilde ekrana bastırdık.

"Aşama 4: Site indeksleme ve sıralama" http://127.0.0.1:5000/indexing Sayfasında

Verilen Url içindeki Url'lerde gezerek kelimeleri başarılı bir şekilde çıkartır.

“Aşama 5: Semantik Analiz” adımında NLTK “from nltk.corpus import wordnet” kütüphanesini kullanarak girilen Url deki anahtar kelimelerin benzer, yakın anlamlı kelimeleri sayfaya döndürüyor. Anahtar kelimeleri kullanırken NLTK - “from nltk.stem import PorterStemmer” kütüphanesini kullanarak kelimeyi kök halinde işleme alır. Bu kelimelerin anlamsal olarak alakalı, benzer olma oranını inisiyatif olarak 0.65% den büyük olan kelimeleri kullanmak üzere sayfaya yazdırır.

http://127.0.0.1:5000/semantik Sayfasında girilen ilk Url’deki anahtar kelimeleri ikinci Url’deki bunlarla yakın anlamlı kelimelerle birlikte başarılı olarak yazdırmaktadır.

KULLANILAN KÜTÜPHANELER:

- scrapper_controller.py
  - from flask import Flask, render_template, jsonify
  - from flask import request
  - from utils import scrape_url, scrape_url2, recursive_helper, semantik

- utils.py
  - import io
  - import re
  - import requests
  - import operator
  - from nltk.corpus import wordnet
  - from nltk.corpus import stopwords
  - from snowballstemmer import stemmer
  - from nltk.stem import PorterStemmer
  - from bs4 import BeautifulSoup,Comment
  - from nltk.tokenize import word_tokenize

KULLANILAN FONKSİYONLAR:

- def scrape_url(url):
  def sozlukolustur(tumkelimeler):

def sembolleritemizle(tumkelimeler):

Fonksiyona alınan web sitesi URl'indeki kelimeleri alıp boşluklarına göre ayırıp gereksiz sembollerden ayırmak için kullanılan bir fonksiyondur.

- def scrape_url2(url):

Yukarıda belirttiğim "scrape_url" fonksiyonun işlemlerine ek olarak Aşama 2 ve 3 için anahtar kelimeleri return eder.

- def create_nested_list(limit):

Verilen web sitesi kümesini 3 derinlikli olarak alır

- def get_all_links(soup, current_url):

Verilen sitenin içindeki "href" etiketli linkleri alır

- def recursive_scrapper(input_url_list):
- def scrape_url3(url):

Girilen Urlde aşağıdaki fonksiyonlarla işlem yapıyor.

- def sozlukolustur(tumkelimeler):
- def sembolleritemizle(tumkelimeler):
- def get_all_links(soup):

- def semantik(first_url,second_url):

Fonksiyonu ile sayfadan 2 farklı Url alır. “scrape_url2” fonksiyonu çağırarak her Url'e ait ayri semantik_list return eder

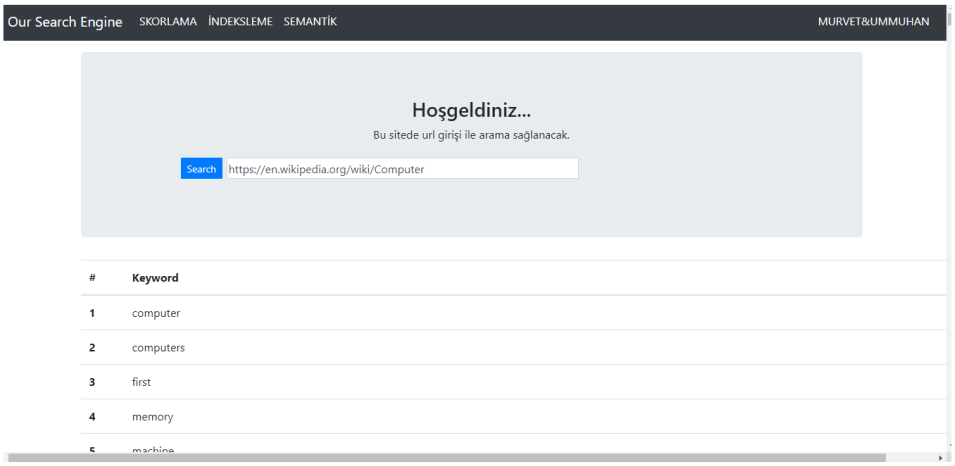
- def find_similarity(first_list, second_list):

Semantik_listlerdeki benzer, yakın anlamlı kelimeleri return eder.

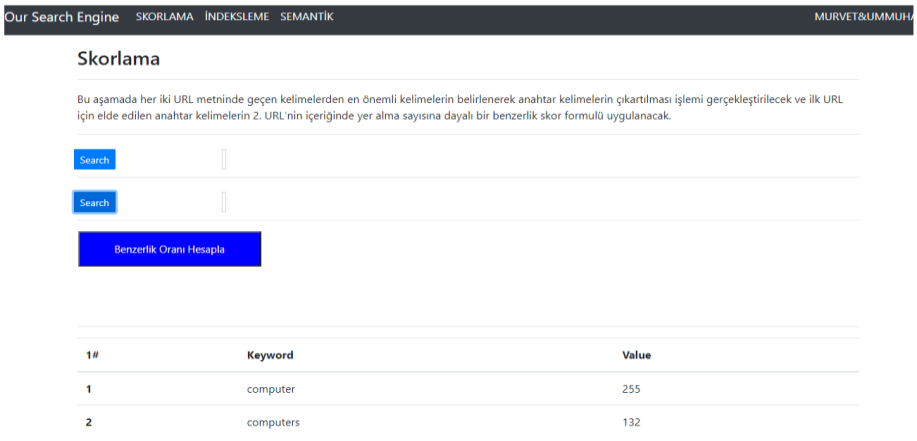
Önemli bilgi: İşlem yoğunluğunun fazla zaman alması sebebiyle verilen her iki URL'deki tüm kelimelerin semantik karşılığını bulmak yerine kısa süreli sonuç

için ilk 15 anahtar kelimeyi kütüphane içerisinde işledik.

PROJE GENEL GÖRÜNÜM



Page: http://127.0.0.1:5000/frekans

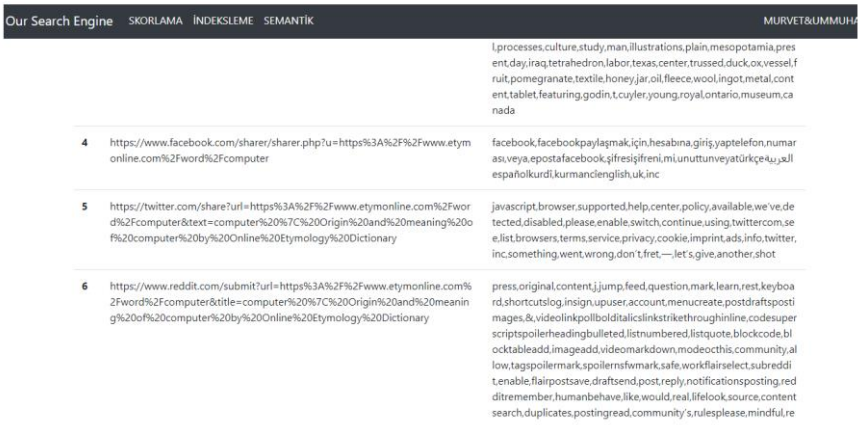


6	program	50
7	devices	49

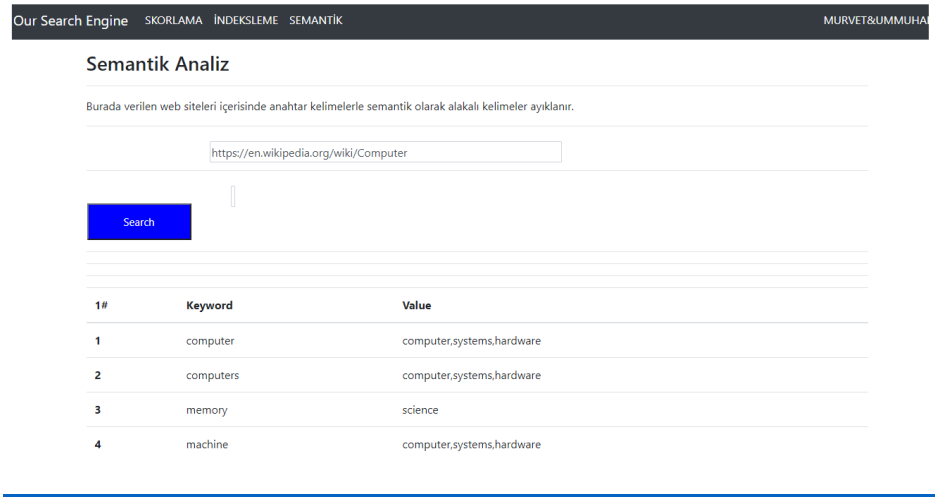
2#	Keyword	Value
1	computer	138
2	engineering	95
3	systems	35
4	software	29
5	engineers	28
6	design	25
7	degree	21

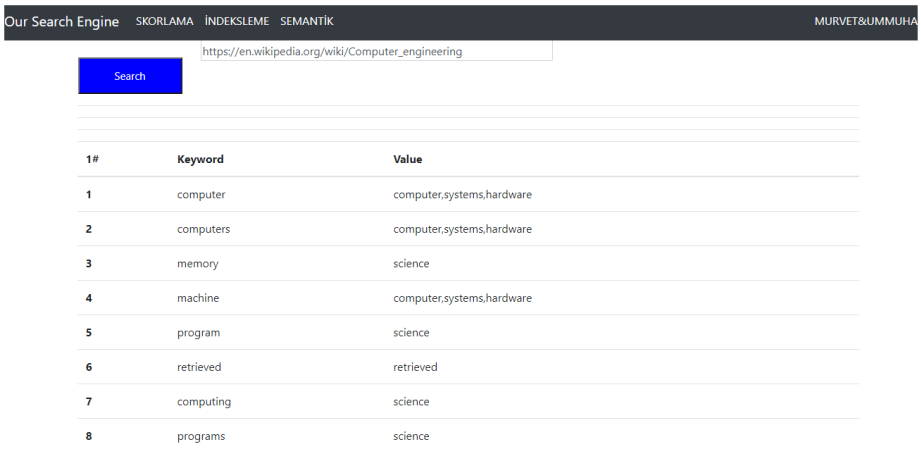
ORAN: 37.861271676300575%

Page: http://127.0.0.1:5000/scoring



Page: http://127.0.0.1:5000/indexing





Page: http://127.0.0.1:5000/semantik

KAYNAKÇA:

- Kaynakça 1:
  - https://towardsdatascience.com/synonyms-and-antonyms-in-python-a865a5e14ce8# =
- Kaynakça 2:
  - https://www.veribilimiokulu.com/natural-language-toolkitnltk/
- Kaynakça 3:
  - https://www.nltk.org/data.html
- Kaynakça 4:
  - https://www.youtube.com/watch?v=II9mlqPmgII&t=1s
- Kaynakça 5:
- Udemy - Mustafa Murat Coşkun (40+ Saat) Python | Sıfırdan İleri Seviye Programlama (2020)