WEB İNDEXLEME UYGULAMASI

Halim Ahat AKTURAN
Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü
Kocaeli Üniversitesi
170201049
170201049@kocaeli.edu.tr

Emre ARIK
Mühendislik Fakültesi,Bilgisayar Mühendisliği Bölümü
Kocaeli Üniversitesi
190201075
190201075@kocaeli.edu.tr

Özet – Projemizin temel amacı kullanıcı tarafından girilen linklerin kelime frekansı, anahtar kelime frekansı, bir siteyle karşılaştırılması, girilen url kümesiyle karşılaştırılması ve semantic analizinin yapılmasıdır.

Anahtar Kelimeler—python programlama dili, flask, site kazıma

I. PROBLEM TANIMI

Bu projenin temel amacı olarak kullanıcı tarafından girilen web siteleri için kelime frekansı, anahtar kelime frekansı, başka bir siteyle karşılaştırılması, alt site indexlerinin alınarak bir url kümesiyle karşılaştırılması ve semantik analizinin yapılması hedeflenmiştir. Bir web sitesi arayüzüyle kullanıcıya işlem yapabileceği 5 adet buton sunulur. Kullanıcı bu butonlardaki işlemlere tıklayarak gerekli işlemlerini gerçekleştirir.

A. Kelime Frekansı

Projenin bu kısmında kullanıcıdan bir url girilmesi istenilir. URL girildikten sonra URL analiz edilerek en çok tekrarlanan kelimeler kullanıcıya sunulur.

B. Anahtar Kelime Frekansı

Bu kısımda kullanıcıdan URL girilmesi istenilir. URL analiz edilir ve en çok tekrar edilen kelimeler bulunur. Ancak bu kısımda hazırladığımız yaklaşık 1000 kelimelik bir CSV dosyası vardır. Bu CSV dosyasının içeriğinde İngilizce temel bağlaçlar, edatlar, yardımcı fiiller gibi anahtar kelimelere etki edecek kelimeler çıkarılmıştır.

En çok tekrar edilen kelimelerden bu CSV dosyasının içeriği çıkartılır ve geriye kalan en çok tekrar eden 10 kelime anahtar kelimeler olarak seçilir.

C. İki Sayfa Arasındaki Benzerlik Sıralaması

Bu aşamada kullanıcıdan 2 adet URL istenilir. Bu URL'ler girildikten sonra siteler analiz edilir. İki sitenin de anahtar kelimeleri alınır ve listelenir. Ek olarak burada ortak kelimeler isimli bir sekme bulunmaktadır. Ortak kelimelerden kasıt iki site içerisindeki anahtar kelimeleri karşılaştırır ve ortak kelimeler listesine ekler. Bu kelimeler butonla ek olarak açılır bir menü olarak arayüzde gösterilmiştir.

İkinci olarak benzerlik skoru hesaplanır. Bu skor bir progress bar içerisine basılarak görsel bir arayüz gösterilir.

Benzerlik skoru hesaplanırken aşağıdaki işlem temel olarak esas alınmıştır.

$$\frac{\frac{Ortak\ Kelime\ Sayısı}{Birinci\ Linkin\ Tüm\ Kelime\ Sayısı} + \frac{Ortak\ Kelime\ Sayısı}{İkinci\ Linkin\ Tüm\ Kelime\ Sayısı}}{2} * 100$$

Örnek :
Ortak Kelime Sayısı:171 , Birinci Link Kelime Sayısı: 576 , İkinci Link Kelime Sayısı : 648

$$\frac{\frac{171}{576} + \frac{171}{648}}{2} * 100 = \%28.04$$

D. Site İndexleme ve Sıralama

Bu aşamada kullanıcıdan bir ana URL ve URL kümesi istenilir. Yine üçüncü aşamadaki gibi benzerlik hesabı yapılır.

Ancak bu aşamada farklı olarak URL kümesinin alt URL'leri de hesaba katılır. Örneğin URL kümesindeki bir linkten 3 tane alt link üretilir. Bu 3 alt linkten de 3 alt link üretilir. Yani toplam 12 tane link olur. Bu kısımda site alt alta 3 derinlikte dallanır. Sistemde çok fazla request gönderdiğimiz için biz alt link sayısını 3 olarak sınırlandırdık.

Ek olarak bu aşamada Alt Linkler ve Derinlikleri isimli bir açılır tablo bulunmaktadır. Bu tabloda kırmızı renkli derinlik 2 kısmında url kümesindeki bir linkten türetilmiş 3 link gösterilmektedir. Derinlik 3 kısmında ise derinlik 2'deki linklerden üretilmiş 3 link gösterilmektedir. Bu tablo renklendirilmiş olarak kullanıcıya sunulmuştur..

Bir sonraki kısımda 3.aşamadaki formüle göre tüm alt linklerin benzerlik oranı hesaplanır ve büyükten küçüğe listelenir. Ek olarak bu aşamada tabloda ana link kelime sayısı, site kelime sayısı, ortak kelime sayısı, karşılaştırma hesabının matematiksel formülü ve karşılaştırma sonucu kullanıcıya bir web arayüzünde gösterilmiştir.

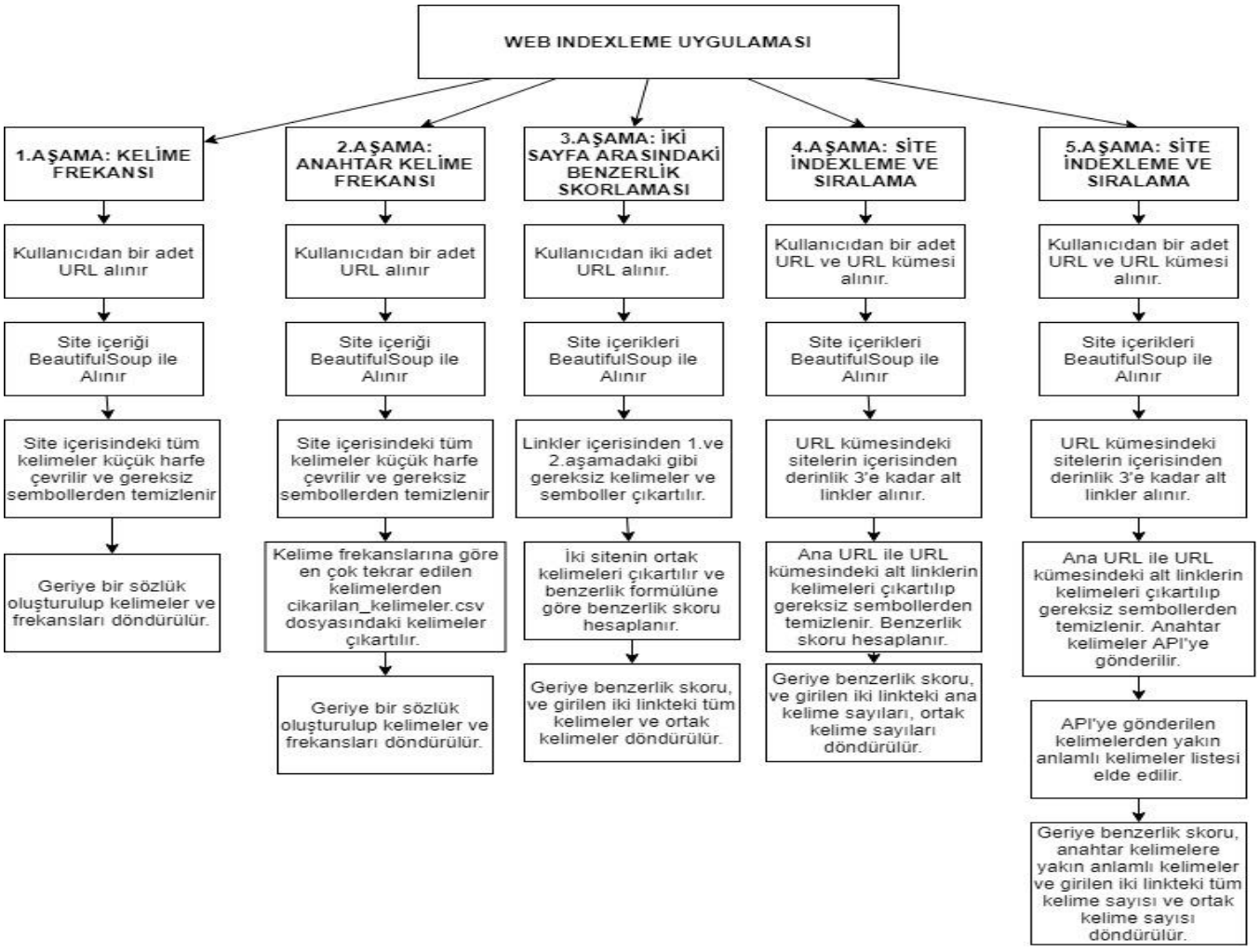
E. Semantik Analiz

Bu aşamada yine kullanıcıdan bir URL ve URL kümesi alınmaktadır. Alınan URL kümesinin alt linkleri yine 4.aşamadaki gibi açılır bir butonla kullanıcıya sunulmuştur.

Bu aşamada yine alt linkler benzerlik skorlarına göre en yüksekten en alta doğru sıralanmıştır. Bu aşamanın 4.aşamadan farkı ise sıralanan sitelerdeki anahtar kelimeler alınmıştır ve İngilizce yakın anlamlı kelimelerin listelenmesi için bir API'ye gönderilmiştir. Bu api https://api.datamuse.com linkinde bulunan günde 100 bin isteğe izin veren ücretsiz bir apidir. Kelime API'ye gönderilir ve benzerlik skorlarına göre en yüksek kelime seçilip listeye eklenilir ve web arayüzünde kullanıcıya gösterilir. Ancak burada dezavantajımız API'ye yaklaşık olarak her siteden 10 anahtar kelimeyi tek tek gönderdiğimiz için API'nın bize yanıt verip kelimelerin listeye eklenmesi uzun sürmektedir. Bu sebeple bu aşama biraz yavaş çalışmaktadır.

Progress bar'ın altında semantik analiz isimli buton bulunmaktadır. Bu butona göre anahtar kelimeye yakın anlamlı kelimeler listesi listelenmektedir.

II. AKIŞ ŞEMASI



III. PROJENIN GENEL YAPISI VE KULLANILAN KÜTÜPHANELER

A. Site Arayüzü ve Kullanılan Kütüphaneler

Proje oluşturulurken temel olarak Python programlama dili kullanılmıştır. Python programlama dilinden elde ettiğimiz verileri websitesine yansıtmak için ise Flask kütüphanesi kullanılmıştır. Ek olarak web site arayüzünde HTML, CSS, Bootstrap ve JavaScriptten faydalanılmıştır.

B. Kelime Frekansı

Kelime frekansı aşamasında kullanıcıdan URL web arayüzünden alınıp Flask kütüphanesi aracılığıyla islem1 sayfasına iletilmiştir. İslem1 sayfasında Post edilen URL alınıp Asama1 python dosyasına gönderilmiştir. Bu dosyada link alınıp BeautifulSoup ile parçalanıp get_text fonksiyonuyla site içerisindeki tüm String'ler çekilmiştir. Çekilen bu stringler en çok tekrar edenden en az tekrar edene kadar sıralanıp websitesine bastırılmıştır.

C. Anahtar Kelime Bulma

Bu aşamada işlemler Kelime Frekansı aşamasıyla benzerdir. Tek farkı ise GenelFonksiyonlar dosyasında bulunan python standart kütüphanelerinden csv okuma kütüphanesi entegre edilip csvOku fonksiyonuyla CSV okunup bu kelimelerin çıkartılması için CSV içeriği lambda fonksiyonuna gönderilmiştir. Tüm kelimelerden CSV içeriği çıkarılmış ve geriye kalan kelimeler döndürülüp ilk 10 kelime tabloda gri bir arkaplan rengiyle işaretlenmiştir.

D. İki Sayfa Arasındaki Benzerlik Skoru

Bu aşamada kullanıcıdan 2 adet URL web sayfasından alınır ve Flask kütüphanesi aracılığıyla islem3 sayfasına gönderilir. İslem3 sayfasında ise bu iki url Asama3 python dosyasına gönderilir. Bu python dosyasında iki sitenin tüm kelimeleri alınıp içerisinden gereksiz kelimeler çıkarılır. Bu kelimeler web arayüzünde kullanıcıya sunulur. Ek olarak tüm kelimeler karşılaştırılıp ortak kelimeler lambda fonksiyonuyla alınır. Ortak kelimeler de web arayüzünde kullanıcıya sunulur. Benzerlik skorunun hesaplanması için 1.sayfadaki tüm kelimeler, 2.sayfadaki tüm kelimeler,ortak kelime sayısı benzerlik skoru hesaplama fonksiyonuna gönderilir ve elde edilen çıktılar flask web arayüzünde kullanıcıya gösterilir.

E. Site Indexleme ve Sıralama

Burada kullanıcıdan bir URL ve URL kümesi alınır ve Flask kütüphanesi aracılığıyla islem4 sayfasına gönderilir. İslem4 sayfasından ise Asama4 python dosyasına gönderilir. Burada ilk olarak ana URL'deki kelime sayısı gereksiz kelimelerden ve sembollerden çıkarılıp sözlük oluşturulur. İkincı kısımda textarea'dan gelen URL kümesi alınır ve split edilerek URL'ler parçalanır. Parçalanan URL içeriği GenelFonksiyonlardaki getLinks fonksiyonuna gönderilir. Bu fonksiyon rekürsif olarak çalışarak 3 derinliğe inerek alt linkleri bulmaktadır. Geriye URL kümesinin alt linklerini içeren bir liste gönderilir.

Bu aşamada getLinks içerisinde BeautifulSoup kütüphanesinden farklı olarak request_html kütüphanesi içerisinde bulunan absolute links fonksiyonu kullanılmıştır. Bunun sebebi işlemin daha hızlı olmasını sağlamaktır.

Bu liste yine for döngüsüne girilerek kelime sayıları, ortak kelime sayıları ve benzerlik skorları hesaplanır. Hesaplanan bu içerikler x listesine atanır ve veriler isimli listeye append edilir. Geriye veriler isimli liste benzerlik skorlarına göre büyükten küçüğe sıralanarak döndürülür. X listesinin içeriğini JSON yapısına benzemektedir. Projenin daha hızlı çalışması için bu yapıya benzetilmiştir.

F. Semantik Analiz

Bu aşama aslında 4.aşamanın çok benzeridir. Burada ek olarak X listesinin içeriğinde URL kümesinin alt dallarındaki sitelerin anahtar kelimeleri de alınır. Anahtar kelimeler https://api.datamuse.com/words?rel_syn= sayfasına =’den sonra yazılacak şekilde gönderilir. API geriye yakın anlamlı kelimelerden skoru en yüksek olanı anahtar kelimeyle eşleyerek bir liste şeklinde gönderilir. API'den gelen liste x listesiyle birleştirilerek Flask web sayfasına gönderilir.

Web sayfası içeriğinde ana link kelime sayısı, alt link kelime sayısı, ortak kelime sayısı, karşılaştırma hesabı ve semantik analiz isimli bir buton bulunmaktadır. Semantik analiz butonuna basıldığında açılır menü olarak analiz sunulmaktadır. Ancak bu kısımda API'ye tek tek istek yaptığımız için bu aşama uzun sürmektedir. Bu sebeple bu sayfa örnek olarak aşağıdaki şekilde gözükmektedir.

Site	Ana Link Kelime Sayısı	Site Kelime Sayısı	Ortak Kelime Sayısı	Karşılaştırma Hesabı	Karşılaştırma Sonucu
https://docs.python.org/3/library/stdtypes.html	212	2638	93	$\frac{\frac{93}{212} + \frac{93}{2638}}{2} = \%23.7$	Semantik Analiz

Anahtar Kelime	Semantik Analiz
return	be restored
object	physical object
string	word string
sequence	chronological succession
set	exercise set
x	tenner

IV. SONUÇLAR

Bu projede sonuç olarak günümüz dünyasında kullanılan mini bir arama moturunun tersi yani kullanıcıdan url alınıp içerik filtrelenmesi yapılmıştır. Projede Python, Flask, BeautifulSoup, RequestHTML, CSV okuma gibi farklı farklı kütüphaneler kullanılmıştır. Seçilen linkler aşama aşama alt dallarına ayrılarak belirlenen formüle göre benzerlik skorlamaları hesaplanmıştır. Anahtar kelimelerin yakın anlamlarının bulunması için bir API'ye gönderilip API'den gelen veriler kullanıcıya web sitesi aracılığıyla sunulmuştur.

V. KAYNAKÇA

[1] Taşçı, Volkan(2019), Python Eğitim Kitabı (DikeyEksen Yayıncılık)

[2] http://tutorialspoint.com/python3/

[3] https://medium.com/@awesome_nyn/python-ile-flask-microframework-kullanarak-nas%C4%B1l-web-projesi-olu%C5%9Fturulur-fbca456e7c71

[4] https://api.datamuse.com//