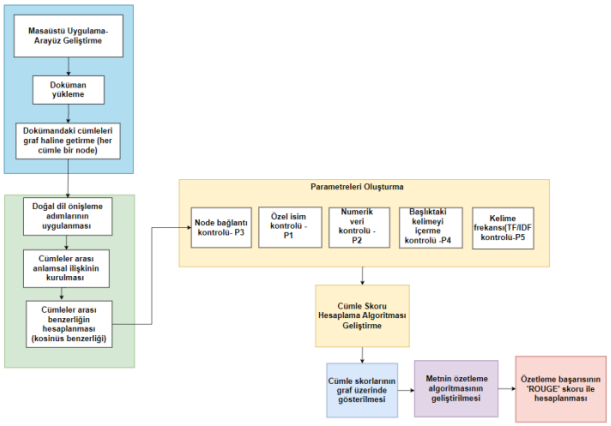
KOCAELI ÜNIVERSITESI
BILGISAYAR MÜHENDISLIĞI
BÖLÜMÜ

ÖZET

Bu raporda projeyi yapmak için kullandığımız yöntemler ve yapım aşamaları,sonucunda aldığımız çıktılar,projeyi anlatan açıklamalar, akış şeması ve yararlanılan kaynaklar bulunmaktadır.Bu projede Girilen dosyadaki cümlelerin benzerliklerini hesaplayarak graf oluşturmak ve cümle skorlarını kullanarak özetleme algoritması yapmak amaçlanmıştır.

GIRIŞ

Bu projede ilk olarak doküman yükleme işlemi gerçekleştirilecektir. Ardından yüklenen dokümandaki cümleleri graf yapısı haline getirmek ve bu graf yapısını görselleştirmek beklenmektedir. Bu grafta her bir cümle bir düğümü temsil edecektir. Cümleler arasındaki anlamsal ilişki kurulmalı, cümleler skorlanmalıdır. Belirli parametreleri kullanarak cümle skorunun hesaplama algoritmasını ve cümle skorlarına göre metin özeti çıkarma algoritmaları geliştirmek istenmektedir. Özet metni arayüzde sunmanız beklenmektedir. Sonuç olarak size verilen bir metnin özetini bu yöntem ile çıkarmanız ve gerçek özet ile benzerliğini "ROUGE" skorlaması ile ölçmemiz istenmektedir.



Şekil 1. *Proje akış diyagramı*

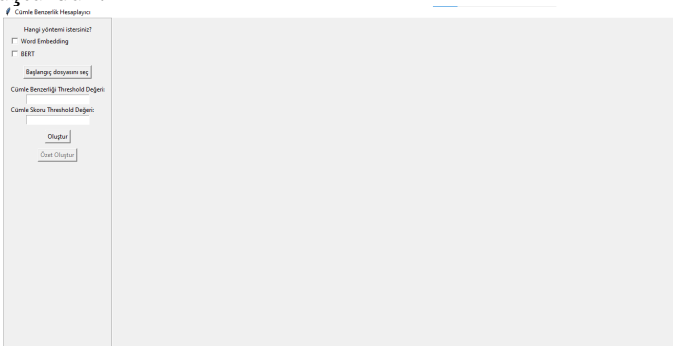
Projede temel amaç; cümleleri graf yapısına çevirip Cümle Seçerek Özetleme (Extractive Summarization) gerçekleştirmektir. Graf yapısına çevirerek cümlelerin metindeki anlamsal ilişkilerini görselleştirmek ve bu ilişkileri kullanarak önemli cümleleri belirlemek amaçlanmaktadır.

A. Masaüstü Arayüzü Geliştirilmesi ve Graf Yapısının Oluşturulması

Masaüstü arayüzü geliştirmemiz beklenmektedir. Arayüz aşağıdaki isterleri içermelidir:

- Kullanıcının doküman yükleyebileceği bir alan,
- Dokümanın graf halinde görüntüleneceği bir alan,
- Cümle benzerliği için threshold seçilebilecek bir araç,
- Cümle skorunun belirlenmesi için threshold seçilebilecek bir araç.
- Cümle benzerliği algoritmasına alternatif oluşturursanız bunun arayüzden seçilebilmesini sağlayan bir araç

Projemizi bu isterleri karşılayabilecek şekilde oluşturduk.



Dokümandaki cümleleri graf yapısına dönüştürmek için hazır bazı veritabanları, kütüphaneler veya API kullanabilirdik.Bunlardan NetworkX kullandık. NetworkX: Python programlama dili için açık kaynaklı bir graf kütüphanesidir. Düğümler ve kenarlar gibi grafik elemanlarını temsil etmek için birden fazla graf sınıfı sağlar.

B. Cümleler Arası Anlamsal İlişkinin Kurulması

Cümlelere NLTK kütüphanesi kullanılarak aşağıdaki ön işleme adımları uygulanmıştır:

- Tokenization: Bir metnin küçük parçalara ayrılmasıdır.
- Stemming: Kelimelerin kökünün bulunması işlemidir.
- Stop-word Elimination: Bir metindeki gereksiz sözcükleri çıkarma işlemidir.Stop word'ler, genellikle yaygın olarak kullanılan, ancak metnin anlamını belirlemekte önemli bir rol oynamayan kelime ve ifadelerdir.
- Punctuation: Cümledeki noktalama işaretlerinin kaldırılmasıdır

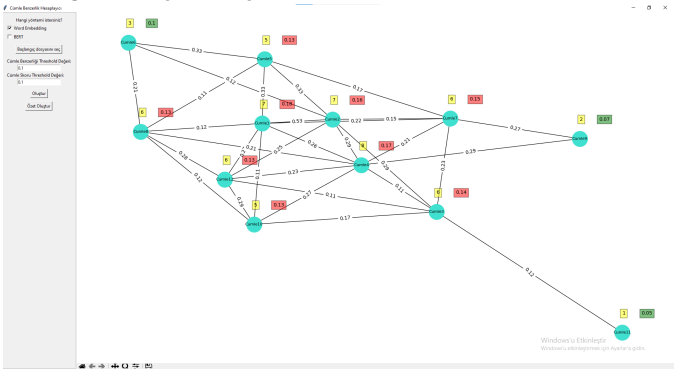
NOT: Cümle skoru hesaplama adımında yapılması gereken özel isim içerme ve nümerik veri içerme adımları bu ön işlemlerden önce gerçekleştirilmelidir.
İki cümle arasındaki anlamsal ilişkiyi kurmak için aşağıdaki yöntemlerden ikisi de kullanılmıştır:

- Word Embedding: Kelime düzeyindeki anlamsal ilişkileri yakalamak için kullanılan bir makine öğrenimi tekniğidir. Cümleleri temsil etmek için word embedding kullanıldığında, her kelime;

vektörleri ile temsil edilir ve cümle vektörü, içerdikleri kelime vektörlerinin toplamıdır. Bu şekilde, cümlelerin anlamsal ilişkileri vektör uzayında ölçülebilir hale gelir.

- BERT: Özellikle doğal dil işleme (NLP) alanında kullanılan bir derin öğrenme modelidir. BERT, bir cümleyi tamamen anlamak ve cümleyi oluşturan kelimelerin birbirleriyle olan ilişkilerini anlamak için kullanılabilir. BERT, önceden eğitilmiş bir modeldir ve büyük bir metin korpusunda önceden eğitilir. Bu sayede, dildeki örüntüleri ve anlamsal ilişkileri öğrenir ve genelleştirir.

Benzerliği ölçmek için "kosinüs benzerliği" yöntemini uyguladık. Kosinüs benzerliği, iki vektör arasındaki benzerliği ölçmek için kullanıldığı gibi, iki cümle arasındaki benzerliği de ölçmek için kullanılır.



C. Cümle Skoru Hesaplama Algoritmasının Geliştirilmesi

Cümle Skoru Hesaplama sırasında aşağıdaki parametreler oluşturuldu.

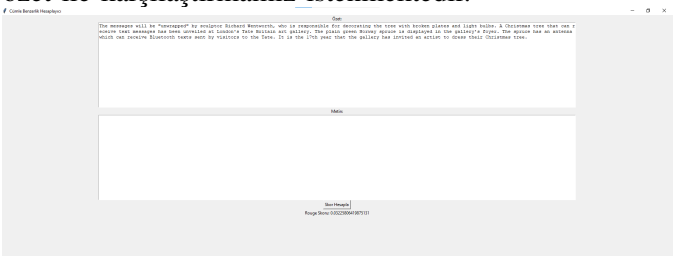
- Cümle özel isim kontrolü (P1) Cümledeki özel isim sayısı / Cümlenin uzunluğu
- Cümlede numerik veri olup olmadığının kontrolü (P2) Cümledeki numerik veri sayısı / Cümlenin uzunluğu
- Cümle benzerliği threshold'unu geçen node'ların bulunması (P3) Tresholdu geçen nodeların bağlantı sayısı / Toplam bağlantı sayısı
- Cümlede başlıktaki kelimelerin olup olmadığının kontrolü (P4) Cümledeki başlıkta geçen kelime sayısı / Cümlenin uzunluğu
- Her kelimenin TF-IDF değerinin hesaplanması (P5). Buna göre dokümandaki toplam kelime sayısının yüzde 10'u 'tema kelimeler' olarak belirlenmelidir. Cümlenin içinde geçen tema kelime sayısı / Cümlenin uzunluğu

D. Skorlara Göre Metin Özetleme Algoritmasının Geliştirilmesi

Önemli cümleler üzerinden gidilerek özet çıkarılacaktır. Özet çıkarmada kullanılan bazı yöntemler şunlardır; Cümle seçerek özetleme: Burada amaç metin içerisindeki önemli cümleleri puanlandırma yöntemleri kullanarak, istatistiksel metotlar ve sezgisel yaklaşımlar ile cümle seçmektir. Yorumlayarak özetleme : Bu tip özetlemedeki amaç metin içerisindeki cümlelerin kısaltılmasıdır. Projede cümle seçerek özetleme yapılmalıdır, yani "var olan cümle yapısı bozulmadan cümleler seçilerek çıkarılıp özet elde edilecektir". Oluşan node skorlarına göre node seçip bunlar ile özet oluşturacak bir metin özetleme algoritması geliştirmeniz beklenmektedir. Algoritmanızda metin özetlenirken hangi cümlelerin hangi sıra ile seçileceğini, cümle skorlarını kullanarak belirledik. Oluşturulan özet arayüzde gösterdik

E. Özetleme Başarısının ROUGE Skoru ile Hesaplanması

Algoritma sonucu oluşan Özet ile metnin gerçek özeti arasındaki benzerliği ROUGE skoru ile hesapladık."ROUGE" skoru, iki metnin benzerliğini ölçmek için kullanılır. Bu benzerlik genellikle referans metinde bulunan kelimelerin özetlenmiş metinde de bulunup bulunmadığına dayanır. Size verilen bir dokümanı özetlemeniz ve yine size verilecek gerçek özet ile karşılaştırmamız istenmektedir.



YONTEM

Tkinter kütüphanesini kullandık. Tkinter, grafik kullanıcı arabirimleri (GUI) oluşturmak için standart bir Python kitaplığıdır.

Bu projede, çeşitli işlevleri ve GUI uygulamasının ana penceresi olan tk.Tk sınıfından türetilen Uygulama adlı bir sınıfı tanımlar.Projede işlevler şunlardır:

- dosyacevir(dosyaYolu): Bir dosyadan satırları okur, cümleleri tokenize eder ve "cumleler.csv" adlı bir

CSV dosyasına kaydeder.

- Onİsleme(): Porter Stemmer kullanarak cümleleri tokenize ederek, stopwords ve noktalama işaretlerini kaldırarak ve kelimeleri köklendirerek önişler. Önceden işlenmiş cümleler daha sonra ”islenmiscumleler.csv” adlı bir CSV dosyasına kaydedilir.

- WordEmbeddingYontemi(): Her cümle için vektör gösterimleri elde etmek için gensim kütüphanesinden Word2Vec modelini kullanarak Word Gömme tekniğini uygular. İşlev, cümle vektörlerinin bir listesini döndürür.

- BertYontemi(): Transformers kitaplığından BERT (Transformers'tan Çift Yönlü Kodlayıcı Temsilleri) modelini uygulayarak her cümle için vektör temsilleri elde eder. İşlev, cümle vektörlerinin bir listesini döndürür.

- ÖzelİsimYazdırma(cumle): Cümledeki özel adların (başlık durumu olan sözcükler) sayısını sayar ve sayımı eksi 1 olarak verir.

- NumerikVeriSadirma(cumle): Bir cümledeki sayısal değerlerin sayısını sayar.

- BaslikSadirma(cumle,ilksatir): Belgenin hem cümlede hem de ilk satırda (başlık olduğu varsayılan) geçen sözcük sayısını sayar.

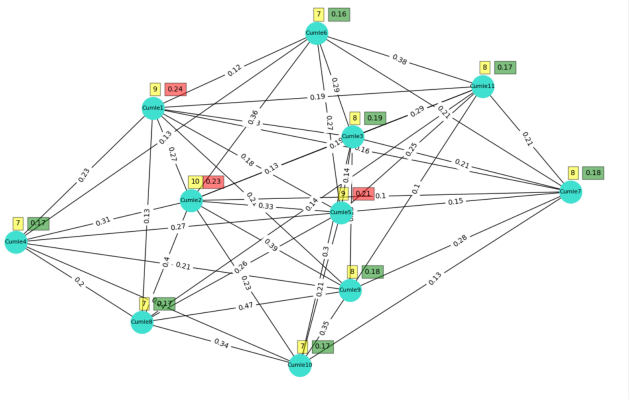
- TemaKelimeSadırma(cumle, dokuman): Cümle ve belge için TF-IDF (Term Frequency-Term Frequency-Term Document Frequency) matrisini hesaplar ve bir eşiğe göre cümledeki tema kelime sayısını verir.

Uygulama sınıfı: Ana uygulama penceresini temsil eder. Bir dosya seçme, bir hesaplama yöntemi (Word Embedding veya BERT) seçme, eşik değerleri ayarlama ve seçilen yönteme göre bir benzerlik grafiği oluşturma yöntemlerini içerir. Ayrıca cümle puanlarını hesaplamak ve grafiği görüntülemek için bir yöntem içerir.

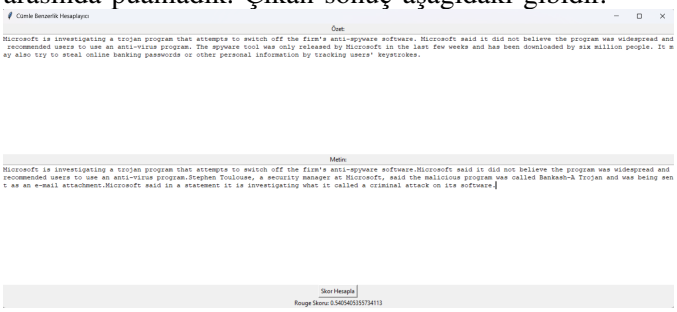
Genel olarak, projemiz Word Gömme ve BERT tekniklerini kullanarak cümle benzerliğini hesaplamak için GUI tabanlı bir arayüz sağlamayı ve sonuçları bir grafik kullanarak görselleştirmeyi amaçlıyor.

DENEYSEL SONUCLAR:

Projemizde aldığımız değerlerin açıklamaları aşağıdaki görseller ile gösterilmiştir:



Yukarıdaki sonucumuzda cümle benzerliği threshold değerini 0.1 ve cümle skoru threshold değerini 0.2 aldığımızda aldık. Bu alanda yeşil ve kırmızı renkli yerler skor değerini göstermektedir. Eğer skor değeri belirlemiş olduğumuz cümle skoru threshold değerinden büyükse kırmızı renkle gösterilmektedir. Küçükse yeşil renk ile gösterilmiştir.
Cümleler arasındaki çizgiler arasındaki değerler cümleler arasındaki benzerlik oranlarını göstermektedir. Ekranda gözüken sarı renkli alanlarda ise bağlantı sayıları gösterilmektedir.Projemizde yüklemiş olduğumuz dökümanın özeti ile kullanıcının özet karşılaştırmasını yaparak arasındaki benzerliği 0 ile 1 arasında puanladık. Çıkan sonuç aşağıdaki gibidir.

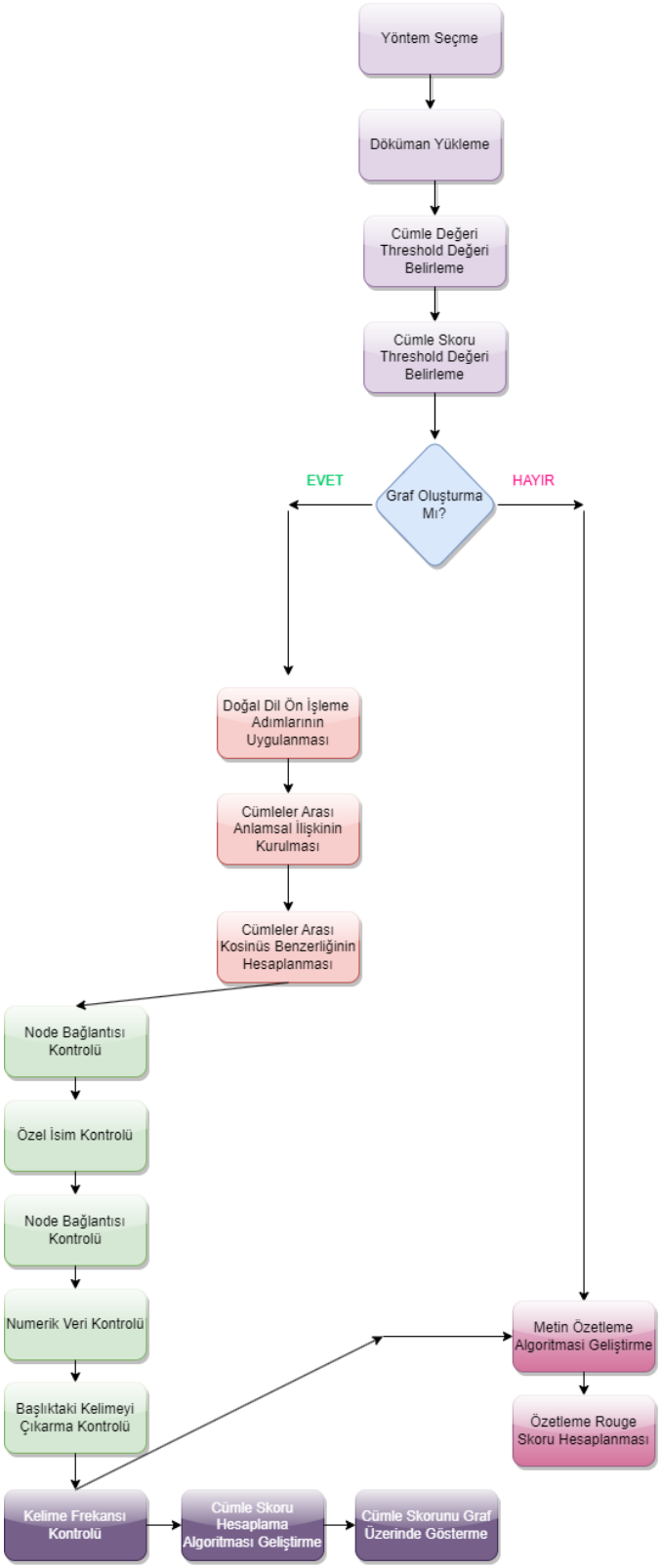


Burada aldığımız değer 0.54 gibi bir değer gelmiştir. Özetleme algoritmamızda özetimizde yüzde 50'lik bir başarı gözükmektedir. Fakat başka özetlerde denendiğinde bu oran yüzde 80'lere de ulaşmaktadır. Projemizde elde etmiş olduğumuz deneysel sonuçları bu şekilde ifade edebilmekteyiz.

SONUÇ

Kod 391 satırdan oluşmaktadır.Masaüstü uygulaması için Tkinter kütüphanesi kullanılmıştır.Grafik gösterimi için NetworkX kullanılmıştır.Alınan cümleler ve işlenmiş cümleler tablo şekline getirilmiştir.Bert ve Word Embedding ikisi de kullanılmıştır.

AKIŞ ŞEMASI



KAYNAKÇA

REFERENCES

[1] HARK, Cengiz, et al. Doğal dil İşleme yaklaşimlari ile yapisal olmayan dökümanlarin benzerliği. In: 2017 International Artificial Intelligence and Data Processing Symposium (IDAP). IEEE, 2017. p. 1-6.

[2] BABÜROĞLU, Barış; TEKEREK, Adem; TEKEREK, Mehmet. Türkçe için derin öğrenme tabanlı doğal dil işleme modeli geliştirilmesi. In: 13th International Computer and Instructional Technology Symposium. 2019. p. 671-679.

[3] KÜÇÜK, Doğan; ARICI, Nursal. Doğal Dil İşlemede Derin Öğrenme Uygulamalari Üzerine Bir Literatür Çalişmasi. Uluslararası Yönetim Bilişim Sistemleri ve Bilgisayar Bilimleri Dergisi, 2018, 2.2: 76-86.

[4] https://medium.com/datarunner/matplotlibkutuphanesi-1-99087692102b;

[5] https://medium.com/cits-tech/python-networkx-ile-graf-teorisi-931699540e73;

[6] https://www.datasciencearth.com/dogal-dil-isleme-1-4-python-dogal-dil-isleme-kutuphaneleri/;

[7] https://medium.com/data-science-tr/makine;

[8] https://medium.com/bili

[9] https://medium.com/@toprakucar/bert-modeli-ile-t

[10] https://www.hosting.com.tr/blog/bert/;

[11] https://www.veribilimiokulu.com/word2vec/;