

FilmAB

İFL Eleştirmenler

İFL Eleřtirmenler

Öneri sistemlerine farklı bir
pencereden bakan lise
öğrencilerinin Türkçe Doğal Dil
İřleme Çalışma Grubu



FilmAB
"Ara Bul"

01

Ekibimiz

Bu kısımda ekibimiz ile ilgili bilgi vereceğiz.

02

Ekip Üyelerinin Projeye Sunduğu Katkı

Ekibimizin görev dağılımı ve projeye sunduğu katkıyı bu kısımda konuşacağız.

03

Ele Aldığımız Problem

Üzerinde çalışmaya değer gördüğümüz problemimizi anlatacağız.

04

Problemın Çözümü

Mevcut alanyazında (literatürde) yapılanlar dışında önerdiğimiz çözümümüz hakkında bilgi vereceğiz.

05

Yöntemimiz

Problem çözümümüzde kullandığımız yöntemimizi açıklayacağız.

Teknik Çalışmalarımız

Proje kapsamında yaptığımız teknik çalışmaları genel hatları ile sunacağız.

Proje İş Akışımız

Proje sürecini nasıl yönettiğimizi anlatacağız.

Proje İle İlgili Yol Haritamız

Bulduğumuz çözüme nasıl ulaştığımız ve ileride yapacağımız iyileştirme ve geliştirme çalışmalarına değineceğiz.

Proje Ön Gösterim Videomuz

Projemizin nasıl çalıştığını gösterdiğimiz videomuzu izleyeceğiz.

Kapanış

Sunumumuzu özetleyip, soru cevap bölümünü yapıp teşekkür edeceğiz.

06

07

08

09

10

Ali Batu ADA

TAKIM KAPTANI

- 11. sınıf öğrencisi
- Teknik becerileri yüksek
- Alanyazında üretim yapmaya istekli
- ÇOK KONUŞKAN (!)

Fahriye Gül OLUR

TAKIM ÜYESİ

- 10. sınıf öğrencisi
- Bilgisayar bilimi alanında bireysel ve takım çalışmalarına yatkın
- Yeni nesil yazılım uzmanı ;)

Sertaç ATEŞ

DANIŞMAN

- Bilişim Teknolojileri Öğretmenimiz

Saadin OYUCU

AKIL HOCASI (Mentor)

- Alanyazın Uzmanı

EKİP ÜYELERİNİN PROJEYE SUNDUĞU KATKI

Ali Batu ADA

1. Ara yüzün tasarlanması ve geliştirilmesi (py)
2. Verilen film isimlerinden o filmlere benzer 10 adet filmin bulunmasından sorumlu modülün yazımı (ipynb)
3. Proje ön gösterim videosunun hazırlanması

Fahriye Gül OLUR

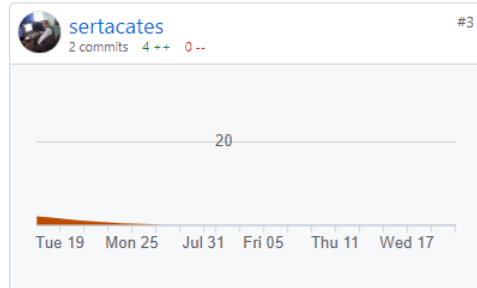
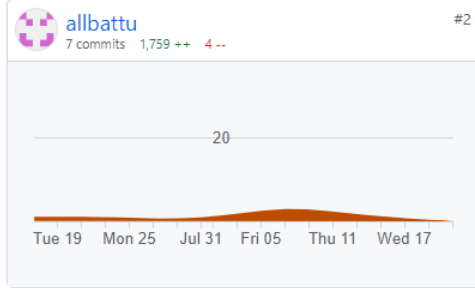
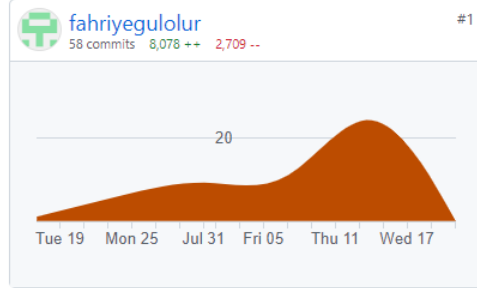
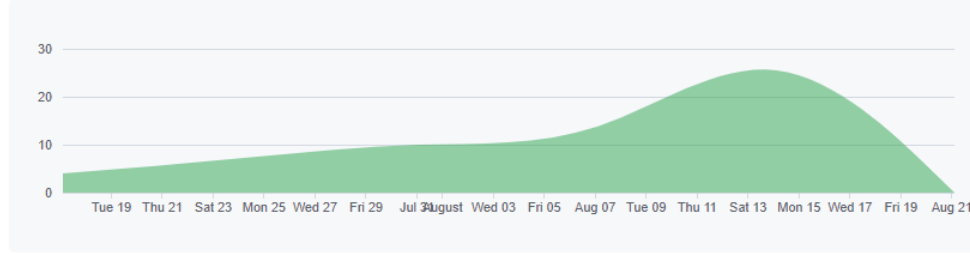
1. Veri kümesinin Türkçeleştirilmesi (csv)
2. Kullanıcının belirttiği senaryoya en uygun 10 adet filmin bulunmasından sorumlu modülün yazımı (ipynb)
3. Konu ile ilgili Türkçe Doğal Dil İşleme Alanyazının takip edilmesi ve proje dokümantasyonunda belirtilmesi (Code Markup)
4. Github hesabının yönetimi
 1. Kurallar bütünlüğü (Code of Conduct)
 2. Projeye katkı sunumu (Contributing)
 3. Proje detay raporu (Readme)
 4. Logo tasarımı

EKİP ÜYELERİNİN PROJEYE SUNDUĞU KATKI

Jul 17, 2022 – Aug 21, 2022

Contributions: Commits ▾

Contributions to main, excluding merge commits and bot accounts



EKİP ÜYELERİNİN PROJEYE SUNDUĞU KATKI

Sertaç ATEŞ (Danışman Öğretmen)

1. İletişim (Ali Batu'nun veya Ali Batu'nun babasının aranması)
2. Google Colab PRO hesabının yönetimi
3. Takıldığımız kısımlarda Danışmanlık kapsamında sorduğumuz sorulara cevap verilmesi veya alanyazına yönlendirme yapılması

Saadin OYUCU (Akıl Hocası - Mentor)

1. Alanyazında öneri sistemlerinin, özellikle film öneri sistemlerinin olduğu bilgisinin verilmesi ve proje öncesinde alanyazında derinlemesine çalışma yapmamızın önerilmesi
2. Proje takibi ve haftalık toplantılarda sorulan sorulara Akıl Hocalığı (Mentorluk) kapsamında cevap verilmesi
3. Google Colab PRO hesabı alınmasının önerilmesi

ELE ALDIĞIMIZ PROBLEM

Can sıkıntısının verimli bir şekilde giderilmesi

- Okulda hep çalış hep çalış nereye kadar?
- Biz de film çevirmesini biliyoruz, çevirelim mi?
- Hiç hobimiz olmasın mı?
- İstedığımız bir filmi izlemek için çevrilmiş tüm filmlerin tek tek özetlerini mi okuyalım?
- Her zaman başka arkadaşlarımızın deneyimlerine göre mi film izleyelim?
- Film isminden film içeriği anlaşılır mı!!!

Türkçe veri kümesi oluşturulması ve Film öneri sistemi hazırlanması

- Alanyazında Türkçe Film Özetleri içeren ve açık lisans (creative commons) ile paylaşılan bir **veri kümesi** yoktur!
- İçerik tabanlı öneri sistemlerinde (content based recommendation) kullanıcının yazdığı **senaryoya** veya **seçtiği filmlere** göre film özetlerini dikkate alarak öneri yapan Türkçe bir uygulama yoktur.

“Bir Trenin La Ciotat Garına Gelişi” adlı film, tarihin ilk sinema filmi olarak 1895 tarihinde Paris’te halkla buluşmuş ve bu buluşma bugünün sinema severleri için bir milat olarak kabul edilmiştir.

Türkçe Doğal Dil İşleme Yapılarak Türkçe Veri Kümesi Oluşturulması

- Huggingface Modelleri ile çeviri
- TXXAI ile çeviri
- Google Translate ile çeviri
- Üretilen veri kümesinin temizlenmesi
- **filmOzetleriVeriKumesi_TURKCE.csv**

Kelime Kökü ve Özü Bulma Çalışması

- Çeşitli kelime kökü / özü bulucu sistemlerin karşılaştırılması (stemming ve Lemmatization)
- Çeşitli kelime / cümle bölümü sistemlerin karşılaştırılması (tokenization)
- Veri ön işlemleri ve hazırlama yapılması

TF / IDF (Dizey) Matrisi Oluşturulması

Benzerliklerin vektörel olarak ölçülmesi
için kelimelerin dizeyde vektörel olarak
saklanması

Benzerlikler

Metinler arasındaki benzerliğin vektörel
olarak ölçülmesi

<https://towardsdatascience.com/17-types-of-similarity-and-dissimilarity-measures-used-in-data-science-3eb914d2681>

YÖNTEMİMİZ

Türkçe Veri Kümesi Oluşturulması

- Huggingface Modelleri, TXXAI ve Google Translate ile kısıtlı metinlerde ön çeviri yapıldı.
- Çeviri doğruluğu ve çeviri hızı dikkate alınarak çeviri yöntemi seçildi.
- Üretilen veri kümesi incelendi ve temizlenmesi gereken veriler temizlendi.
- Github üzerinden **filmOzetleriVeriKumesi_TURKCE.csv** dosyası **rar** olarak herkesin kullanımına sunuldu.

Veri Ön İşlemleri ve Veri Hazırlama

- snowballstemmer, PorterStemmer, TurkishStemmer, WordNetLemmatizer, LancasterStemmer, RegexpStemmer ve ZEMBEREK test kelimeleri ile incelendi. Zaman olarak **en kötü işlem süresine sahip** olan ZEMBEREK veri hazırlama için seçildi.

TF / IDF Veri Dizeyinin Oluşturulması

- TF (Terim Sıklığı), DF (Döküman Sıklığı) ve IDF (Ters Döküman Sıklığı) ile kelime vektörleri için dizey (matris) oluşturuldu.
- Yapılan kök / öz bulma işleminin vektör dizeyinin boyutunu nasıl etkilediği görüldü.

Benzerliklerden yararlanarak önerilerin yapılması

- Verilen senaryoya veya filmlerin özetlerindeki metinlerin vektörleri ile TF / IDF dizeyinin (matrisinin) kosinüs benzerliği ölçülüp en yakın 10 film özeti için 10 filmin ismi kullanıcıya öneri olarak döndürüldü.

Türkçe Veri Kümesi
Oluşturulması

- Huggingface Modelleri, TXTAI ve Google Translate ile kısıtlı metinlerde ön çeviri yapıldı.
- Çeviri doğruluğu ve çeviri hızı dikkate alınarak çeviri yöntemi seçildi.
- Üretilen veri kümesi incelendi ve temizlenmesi gereken veriler temizlendi.
- Github üzerinden **filmOzetleriVeriKumesi_TURKCE.csv** dosyası **rar** olarak herkesin kullanımına sunuldu.

Veri Ön İşlemleri ve
Veri Hazırlama

ÇEVİRİYİ txtai ile yaparsak çeviri için 24 gün beklememiz gerekiyor!

```
# # txtai ve ilgili tüm bağılılıklarını yüklememiz gerekiyor.
# %%capture
# !pip install git+https://github.com/neuml/txtai#egg=txtai[pipeline]
# # Çeviri Modelini Çağırıyoruz / Oluşturuyoruz
# %%capture
# from txtai.pipeline import Translation
# # Çeviri modelinin örnek kullanımı:
# cevirci = Translation()
# cevrilecekVeri = "The sky is blue, the stars are far"
# ceviri = cevirci(cevrilecekVeri, "tr")
# print(ceviri)
#
# # Google Colab'ta txtai çeviricisini kullanarak veri kümesi oluşturulması
# cevirci = Translation()
# satirSayisi, sutunSayisi = df.shape
# for satir in range(satirSayisi):
#     cevrilecekVeri = df['Olay Dizisi'][satir]
#     ceviri = cevirci(cevrilecekVeri, "tr")
#     df.at[satir, 'Olay Dizisi']=ceviri
```

TF / IDF Veri
Dizisinin
Oluşturulması

Benzerliklerden
yararlanarak
önerilerin yapılması

Limonata'dan Şekeri, Suyu ve Naneyi Çıkar ÖZÜ LİMON

LİMONA değil LİMO değil LEMAN hiç değil!

Türkçe Veri Kümesi
Oluşturulması

Veri Ön İşlemleri ve
Veri Hazırlama

TF / IDF Veri
Dizeyinin
Oluşturulması

Benzerliklerden
yararlanarak
önerilerin yapılması

- snowballstemmer, PorterStemmer, TurkishStemmer, WordNetLemmatizer, LancasterStemmer, RegexpStemmer ve ZEMBEREK test kelimeleri ile incelendi. Zaman olarak **en kötü işlem süresine sahip** olan ZEMBEREK veri hazırlama için seçildi.

STEMMING Yöntemi	STEMMING Aracı	Test kelimeleri: ['kalem', 'ilişkilendiremediklerimiz', 'gözlük', 'gözlem']
Kök Bulma YÖNTEM 1	snowballstemmer	Çıktılar: ['kale', 'ilişkilendiremedik', 'gözlük', 'gözle']
Kök Bulma YÖNTEM 2	PorterStemmer	Çıktılar: ['kalem', 'ilişkilendiremediklerimiz', 'gözlük', 'gözlem']
Kök Bulma YÖNTEM 3	TurkishStemmer	Çıktılar: ['kalem', 'ilişkilendiremedik', 'gözlük', 'gözle']
Kök Bulma YÖNTEM 4	WordNetLemmatizer	Çıktılar: ['kalem', 'ilişkilendiremediklerimiz', 'gözlük', 'gözlem']
Kök Bulma YÖNTEM 5	LancasterStemmer	Çıktılar: ['kalem', 'ilişkilendiremediklerim', 'gözlük', 'gözlem']
Kök Bulma YÖNTEM 6	RegexpStemmer	Çıktılar: ['kalem', 'ilişkilendiremediklerimiz', 'gözlük', 'gözlem']
Kök Bulma YÖNTEM 7	ZEMBEREK	Çıktılar: ['kale', 'ilişki', 'gözlük', 'gözlem']

Türkçe doğal dil işleme alanında daha hızlı sonuç ve daha nitelikli sonuç veren kök / öz bulucu araçlara İHTİYAÇ vardır!

YÖNTEMİMİZ

Türkçe Veri Kümesi
Oluşturulması

Veri Ön İşlemleri ve
Veri Hazırlama

TF / IDF Veri
Dizeyinin
Oluşturulması

Benzerliklerden
yararlanarak
önerilerin yapılması

Öz Bulucu	TF IDF Nesnesinin Dizeyinin (Matrisinin) Boyutu (aranilacak_tfidfDizeyi.shape)
Veri ön işlemleri olmadan	(34886, 225577)
nltk.stem	(34886, 215685)
TurkishStemmer	(34886, 113148)
Zemberek	(34886, 88404)

- TF (Terim Sıklığı), DF (Döküman Sıklığı) ve IDF (Ters Döküman Sıklığı) ile kelime vektörleri için dizey (matris) oluşturuldu.
- Yapılan kök / öz bulma işleminin vektör dizeyinin boyutunu nasıl etkilediği görüldü.

- Verilen senaryoya veya filmlerin özetlerindeki metinlerin vektörleri ile TF / IDF dizeyinin (matrisinin) **kosinüs benzerliği** ölçülüp en yakın 10 film özeti için 10 filmin ismi kullanıcıya öneri olarak döndürüldü.

- Türkçe Doğal Dil İşleme alanına katkı sağlamak amacıyla kod işaretlemesi (code markup) yaparak Jupiter Notebook dosyası hazırladık.
- Bu dosyamız ile bir veri kümesinin nasıl Türkçeye çevrileceğini izah ettik.

<https://github.com/IFL->

[Ellestirmenler/FilmAB/blob/main/VeriKumesininTurkcelestirilmesi.ipynb](https://github.com/IFL-Ellestirmenler/FilmAB/blob/main/VeriKumesininTurkcelestirilmesi.ipynb)

```
# Google Colab'ta Google Translator çeviricisinin nasıl kullanıldığına bir örnek görelim.  
from googletrans import Translator  
googleCevirici = Translator()  
cevrilecekVeri = 'This English text have been translated to Turkish by beeing used Google Translator'  
cevrilen = googleCevirici.translate(cevrilecekVeri,dest='tr').text  
print(cevrilen)
```

Bu İngilizce metin Google Translate kullanılarak Türkçe'ye çevrilmiştir.

- Çeviri yöntemimizi belirlemeden önce Huggingface Modellerinden dört tanesi, (Helsinki-NLP/opus-mt-en-mul, Helsinki-NLP/opus-mt-tc-big-en-tr, Helsinki-NLP/opus-tatoeba-en-tr ve Helsinki-NLP/opus-mt-en-trk) TXTAI ve Google Translate üzerinde çalıştık.

- Türkçe Doğal Dil İşleme alanına katkı sağlamak amacıyla Türkçe Film Özetleri içeren ve açık lisans (creative commons) ile paylaşılan bir veri kümesi oluşturduk.
- DOSYA ADI: *filmOzetleriVeriKumesi_TURKCE.csv*

Veri Kümesi

Veri kümesi olarak [wiki_movie_plots_deduped.csv](#) adlı dosyayı kullandık. Sütun açıklamaları aşağıda tablo halinde verilmiştir:

İsim	Açıklama	Satır Sayısı (Unknown)	Satır Sayısı (NaN)	Satır Sayısı (Dolu)
Çıkış Yılı	Filmin yayınlandığı yıl	0	0	34886
Orijinal İsmi	Film başlığı	2	0	34886
Yapıldığı Ülke	Filmin kökeni (ör. Amerikan, Bollywood, Tamil vb.)	0	0	34886
Yönetmeni	Yönetmen(ler)	1124	0	34886
Oyuncu Kadrosu	Baş aktör ve aktrisler	1	1422	33464
Türü	Film Tür(ler)i	0	0	34886
Veri Kaynağı	Konu açıklamasının çıkarıldığı Wikipedia sayfasının URL'si	0	0	34886
Olay Dizisi	Film konusunun uzun biçimli açıklaması (UYARI: Spoiler içerebilir!!!)	0	0	34886

- Türkçe Doğal Dil İşleme alanına katkı sağlamak amacıyla kod işaretlemesi (code markup) yaparak üç farklı Jupiter Notebook dosyası hazırladık.
- Bu üç dosyamız ile içerik tabanlı bir öneri sistemimin nasıl yapılacağını izah ettik.

<https://github.com/IFL-Elestirmenler/FilmAB/blob/main/toplam10filmlismindenOzetlerineBakipBenzerFilmlerinBulunmasi.ipynb>

<https://github.com/IFL-Elestirmenler/FilmAB/blob/main/filmOzetindenBenzerFilmlerinBulunmasi.ipynb>

https://github.com/IFL-Elestirmenler/FilmAB/blob/main/filmOzetindenBenzerFilmlerinBulunmasi_ZEMBEREK.ipynb

```
In [116.. def tavsiyeVer(filmBasligi, cosine_sim=cosine_sim):
            idx = indices[filmBasligi]
            sim_scores = list(enumerate(cosine_sim[idx]))
            sim_scores = sorted(sim_scores, key=lambda x: x[1], reverse=True)
            sim_scores = sim_scores[1:11]
            movie_indices = [i[0] for i in sim_scores]
            return df['Original İsmi'].iloc[movie_indices]

In [117.. print(tavsiyeVer(gereksizKelimelerCikarilmisArananFilmOzeti))
```

- Türkçe Doğal Dil İşleme alanına katkı sağlamak amacıyla proje kapsamında snowballstemmer, nltk.stem (PorterStemmer), TurkishStemmer, nltk.stem (WordNetLemmatizer), nltk.stem (LancasterStemmer) ve nltk.stem (RegexStemmer) ile ZEMBEREK'i karşılaştırdık. Başka bir deyişle, Stemming, yani kelime eklerini kaldırarak veya değiştirerek bir kelimenin ortak kök biçimini bulmak ve Lemmatization, yani bir kelimenin çekimli biçimlerinin temel biçimini bulmak üzerine çalıştık.

STEMMING Yöntemi	STEMMING Aracı	Test kelimeleri: ['kalem', 'ilişkilendiremediklerimiz', 'gözlük', 'gözlem']
Kök Bulma YÖNTEM 1	snowballstemmer	Çıktılar: ['kale', 'ilişkilendiremedik', 'gözlük', 'gözle']
Kök Bulma YÖNTEM 2	PorterStemmer	Çıktılar: ['kalem', 'ilişkilendiremediklerimiz', 'gözlük', 'gözlem']
Kök Bulma YÖNTEM 3	TurkishStemmer	Çıktılar: ['kalem', 'ilişkilendiremedik', 'gözlük', 'gözle']
Kök Bulma YÖNTEM 4	WordNetLemmatizer	Çıktılar: ['kalem', 'ilişkilendiremediklerimiz', 'gözlük', 'gözlem']
Kök Bulma YÖNTEM 5	LancasterStemmer	Çıktılar: ['kalem', 'ilişkilendiremediklerim', 'gözlük', 'gözlem']
Kök Bulma YÖNTEM 6	RegexStemmer	Çıktılar: ['kalem', 'ilişkilendiremediklerimiz', 'gözlük', 'gözlem']
Kök Bulma YÖNTEM 7	ZEMBEREK	Çıktılar: ['kale', 'ilişki', 'gözlük', 'gözlem']

TEKNİK ÇALIŞMALARIMIZ

Kaynaklar:

- <https://nlp.stanford.edu/IR-book/html/htmledition/stemming-and-lemmatization-1.html>
- https://www.tutorialspoint.com/natural_language_toolkit/natural_language_toolkit_stemming_lemmatization.htm

In [13]:

```
testEdilecekKelimeler = ['kalem','iliskilendiremediklerimiz','gözlük','gözlem']
print('Test edilecek kelimeler:',testEdilecekKelimeler)
print('Kök Bulma YÖNTEM 1 - snowballstemmer','\n',50*'-','sep='')
!pip install snowballstemmer
from snowballstemmer import TurkishStemmer
turkStem = TurkishStemmer()
testSonucu = []
for kelime in testEdilecekKelimeler:
    testSonucu.append(turkStem.stemWord(kelime))
print(testSonucu,'\n\n\n')

print('Kök Bulma YÖNTEM 2 - nltk','\n',50*'-','sep='')
!pip install nltk
from nltk.stem import PorterStemmer
ps=PorterStemmer()
testSonucu = []
for kelime in testEdilecekKelimeler:
    testSonucu.append(ps.stem(kelime))
print(testSonucu,'\n\n\n')

print('Kök Bulma YÖNTEM 3 - TurkishStemmer','\n',50*'-','sep='')
!pip install TurkishStemmer
from TurkishStemmer import TurkishStemmer
stemmer = TurkishStemmer()
testSonucu = []
for kelime in testEdilecekKelimeler:
    testSonucu.append(stemmer.stem(kelime))
print(testSonucu,'\n\n\n')

print('Kök Bulma YÖNTEM 4 - WordNetLemmatizer','\n',50*'-','sep='')
import nltk
nltk.download('wordnet')
nltk.download('omw-1.4')
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
testSonucu = []
for kelime in testEdilecekKelimeler:
    testSonucu.append(lemmatizer.lemmatize(kelime))
print(testSonucu,'\n\n\n')

print('Kök Bulma YÖNTEM 5 - LancasterStemmer','\n',50*'-','sep='')
import nltk
from nltk.stem import LancasterStemmer
stemmer = LancasterStemmer()
testSonucu = []
for kelime in testEdilecekKelimeler:
    testSonucu.append(stemmer.stem(kelime))
print(testSonucu,'\n\n\n')

print('Kök Bulma YÖNTEM 6 - RegexpStemmer','\n',50*'-','sep='')
import nltk
from nltk.stem import RegexpStemmer
stemmer = RegexpStemmer('tr')
testSonucu = []
for kelime in testEdilecekKelimeler:
    testSonucu.append(stemmer.stem(kelime))
print(testSonucu,'\n\n\n')
```

- Türkçe Doğal Dil İşleme alanına katkı sağlamak amacıyla proje kapsamında Stemming ve Lemmatization'ın TF / IDF Vektörlerinin Dizeyinin boyutu üzerine etkisi üzerinde çalıştık.

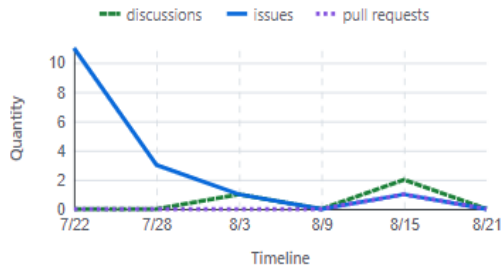
Öz Bulucu	TF IDF Nesnesinin Dizeyinin (Matrisinin) Boyutu (aranilacak_tfidfDizeyi.shape)
Veri ön işlemleri olmadan	(34886, 225577)
nltk.stem	(34886, 215685)
TurkishStemmer	(34886, 113148)
Zemberek	(34886, 88404)

Community insights

Period: Last 30 days ▾

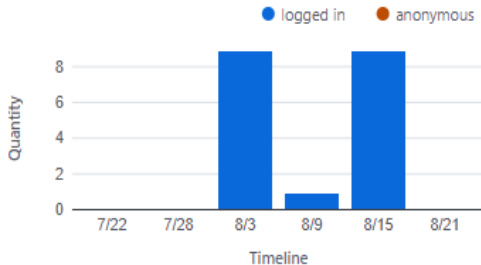
Contribution activity

Count of total contribution activity to Discussions, Issues, and PRs



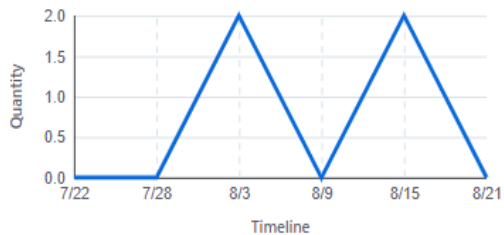
Discussions page views

Total page views to Discussions segmented by logged in vs anonymous users.



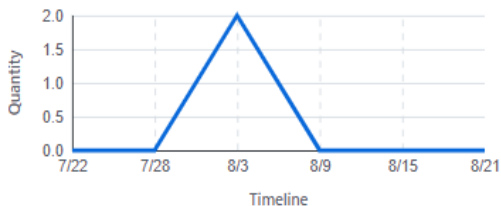
Discussions daily contributors

Count of unique users who have reacted, upvoted, marked an answer, commented, or posted in the selected period.



Discussions new contributors

Count of unique new users to Discussions who have reacted, upvoted, marked an answer, commented, or posted in the selected period.



TEKNİK
ÇALIŞMALARIMIZ

TEKNİK ÇALIŞMALARIMIZ

Community Standards

Here's how this project compares to [recommended community standards](#).

Checklist

<div><div></div></div>	
✓ Description	
✓ README	←
✓ Code of conduct	←
✓ Contributing	←
✓ License	
● Issue templates	<button>Add</button>
● Pull request template	
✓ Repository admins accept content reports	

What is the [community profile](#)?

TEKNİK ÇALIŞMALARIMIZ

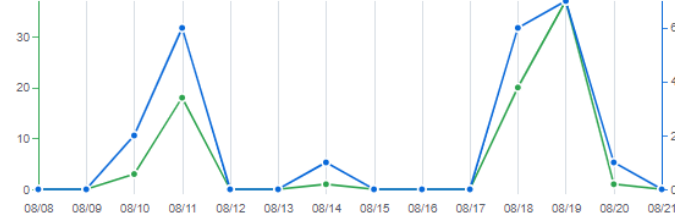
Popular content

Content	Views	Unique visitors
IFL-Eleştirmenler/FilmAB: Verilen fil...	244	35
Issues	49	7
Editing FilmAB/README.md at main	38	3
FilmAB/README.md at main	37	3
Proje Literatür Araştırması tamamlı...	21	3
FilmAB/filmOzetindenBenzerFilmle...	18	4
Forks	16	8
Pulse	15	3
Upload files	13	3
FilmAB/logo.png at main	11	4

Referring sites

Site	Views	Unique visitors
github.com	161	12
Google	10	4

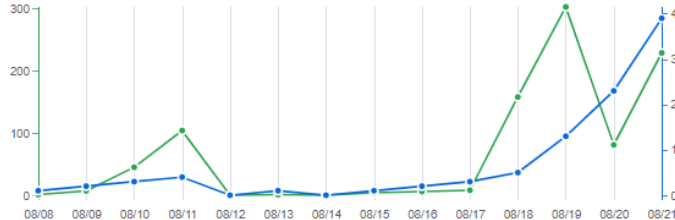
Git clones



80 Clones

8 Unique cloners

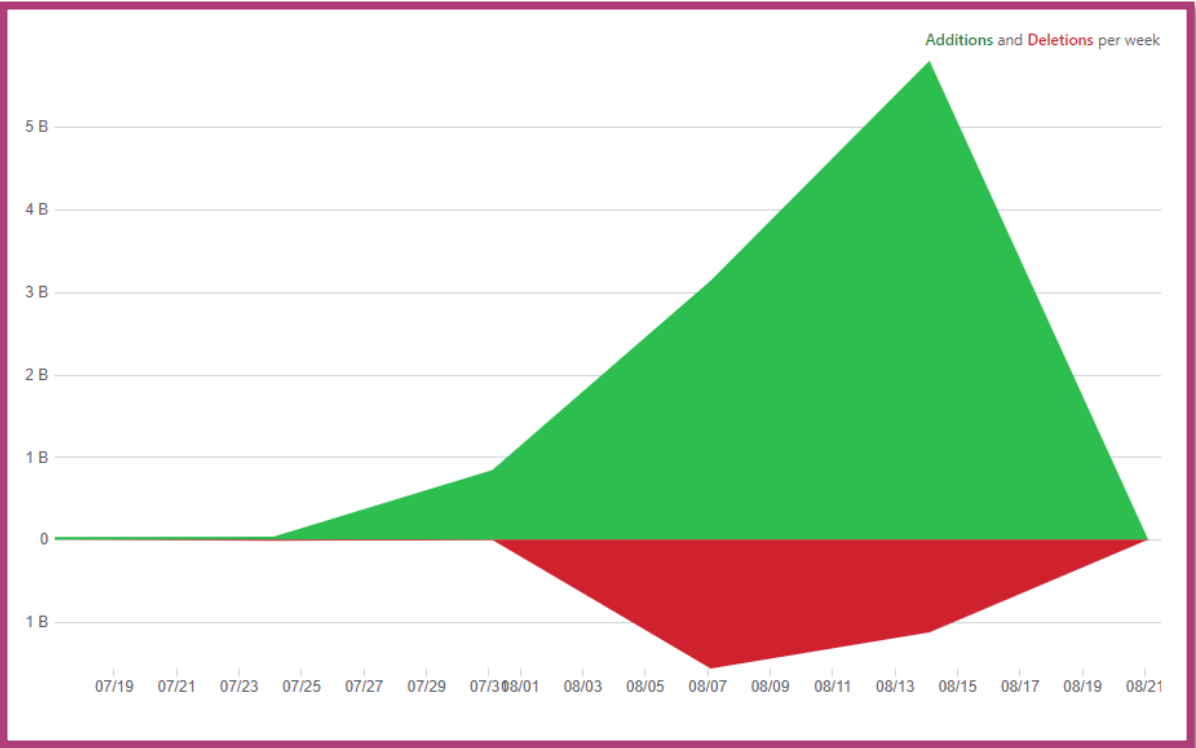
Visitors



947 Views

72 Unique visitors

TEKNİK ÇALIŞMALARIMIZ



PROJE İŞ AKIŞIMIZ

IFL-Eleştirmenler / Projects / FilmAB Projesi - Süreç Yönetim Tablosu

FilmAB Projesi - Süreç Yönetim Tablosu

Proje Akışı

Konuşulan Konular, Alınan Kararlar ve Görevli Listesi

+ New view

Title	Assignees	Status
1 README.md Dosyasını düzenleyelim.	allbattu and fahriye...	<div>Done</div>
2 Repo Etiketleri ve Açıklamaların Güncellenmesi		<div>Done</div>
3 Proje Dokümantasyon		<div>Done</div>
4 Avatar Seçimi ile görev listesi yönetimini uygulayarak öğrenelim.		<div>Done</div>
5 Mentörümüzle Haftalık Toplantı	allbattu and fahriye...	<div>Done</div>
6 Proje Literatür Araştırması tamamlandı mı?	allbattu and fahriye...	<div>Done</div>
7 Veri Kümesi ve Model Eğitimi ile ilgili çalışmamız hangi aşamada?	allbattu and fahriye...	<div>Done</div>
8 Hugging Face üzerinden aldığınız kursu bitti mi?		<div>Done</div>
9 "Doğal Dil İşleme- İsmail Arı Yaz Okulu 2018" videolarında durum nedir?		<div>Done</div>
10 Proje Ara Yüzünü tasarladık mı?	allbattu	<div>Done</div>
11 Saadin Oyucu - Mentör İle Tanışma		<div>Done</div>
12 EĞİTİM: Numpy ve Pandas Kütüphanelerinde eksiğimiz var mı?		<div>Done</div>
13 Sosyal Medya Görünümü - Logo, Etiketler ve Proje Açıklaması		<div>Done</div>
14 Proje Gelişim Aşamaları	allbattu and fahriye...	<div>Done</div>
15 Özetten Film Arama	allbattu and fahriye...	<div>Done</div>
16 Verilen Film İsimlerinden Arama	allbattu and fahriye...	<div>Done</div>

+ You can use

Ctrl

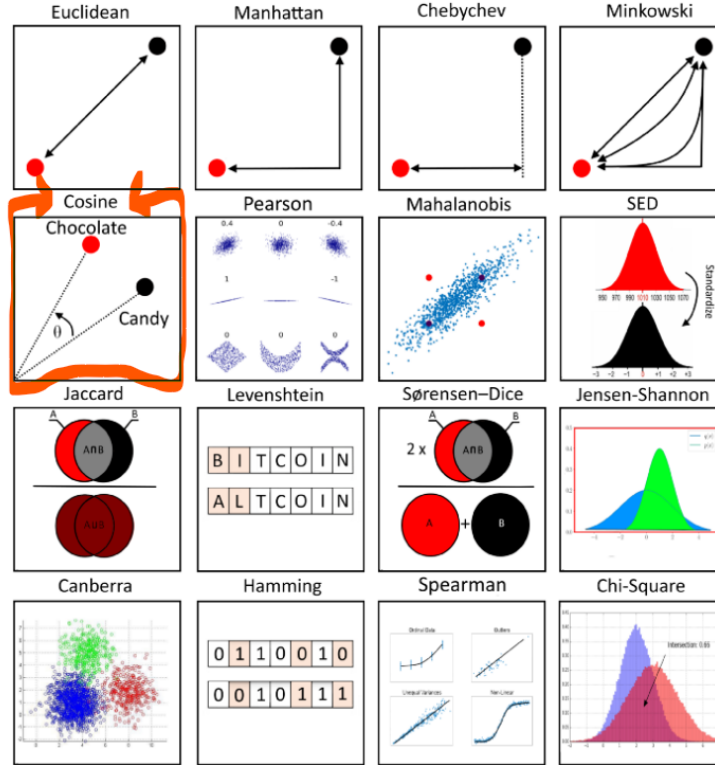
 +

Space

 to add an item

PROJE İLE İLGİLİ YOL HARİTAMIZ

- Olay Dizilerimiz ile arananFilmOzetinin benzerliği üzerinde yeterince çalışmadık.
- Google Colab PRO hesabı ile sadece cosinüs benzerliğinden faydalandık **diğer benzerliklerin karşılaştırmasını yapıp en verimlisini veya en performansı yüksek olanı belirleyemedik.**
- Hangi benzerliğin ve benzerliklerin öneri sistemlerinde kullanılmasının daha uygun olacağı üzerinde çalışmamız yerinde olacaktır.



Various ML metrics. Inspired by [Maarten Grootendorst](#).

- Alanyazındaki tavsiye sistemleri üzerinde yaptığımız okumalarda tavsiye sisteminin kullanıcıların davranışlarını kaydettiği ve daha sonra kullanıcıların davranışlarına göre tavsiye verdiğini gördük.
- Bizim kullanıcıdan toplayabileceğimiz veri kümesi
kullanıcıların istediği film senaryolarıdır.
İleride **senaryo yazarları için senaryo konusu veya senaryo özeti sunan bir çalışma** yapabiliriz.

PROJE ÖN GÖSTERİM VIDEOMUZ

<https://github.com/IFL->

[Elstirmenler/FilmAB/blob/main/FilmAB%20Projesi%20Demo%20-](https://github.com/IFL-Elstirmenler/FilmAB/blob/main/FilmAB%20Projesi%20Demo%20-%20Ekran%20Ara%20Y%C3%BCz%C3%BC%20Kullan%C4%B1m%C4%B1.mp4)

[%20Ekran%20Ara%20Y%C3%BCz%C3%BC%20Kullan%C4%B1m%C4%B1.mp4](https://github.com/IFL-Elstirmenler/FilmAB/blob/main/FilmAB%20Projesi%20Demo%20-%20Ekran%20Ara%20Y%C3%BCz%C3%BC%20Kullan%C4%B1m%C4%B1.mp4)

☰ README.md



FilmAB Projesi Demo - Ekran Ara Yüzü Kullanımı için [buraya](#) tıklayınız

FilmAB Projemiz

TeknoFest 2022 Samsun Türkçe Doğal Dil İşleme Yarışması kapsamında Serbest Kategori alanında Türkçe Veri Kümesi de oluşturduğumuz bir film öneri robotudur. Arayüzümüz açıldığında size iki seçenek sunar:

1. Özet girerek: Nasıl bir film izlemek istediğinizi belirterek,
2. Birkaç film ismi girerek: Verdiğiniz filmlere benzer film isteyerek, veri kümesinde uygun filmi bulabilirsiniz. (#Acikhack2022TDDİ)



ÖZEL TEŞEKKÜR

- Yarışma yönetimini temsilen Saygıdeğer Tuba ÖZTEPE hanımefendiye özel olarak teşekkür eder, saygılarımızı sunarız.
- Yarışmada akıl hocalığı yapan değerli uzmanları temsilen Sayın Saadin OYUCU hocamıza özel teşekkürlerimizi iletir, en derin saygılarımızı sunarız.
- Ve bize yarışma boyunca destek olduğu için Sertaç ATEŞ hocamıza da teşekkür ederiz.
- Bize hedef vererek okulumuzdaki yapay zeka çalışmalarını daha anlamlı hale getiren Türkçe Doğal Dil İşleme Yarışması ekibine sonsuz teşekkürlerimizi ve en kalbi saygılarımızı sunarız.

