Ders: Veri Madenciliği (FET445)

Proje Başlığı:

Amazon Ürün Yorumları ile Yapay Zeka Destekli Hibrit Öneri Sistemi

Takım Adı: Semicolon

Takım Üyeleri

Ad-Soyad	Öğrenci No	E-mail
Gays Harmuş	**22040301144**	gaysharmus@stu.topkapi.edu.tr
Ahmed Asfour	**22040301084**	ahmedasfour@stu.topkapi.edu.tr
Elisa Demir	**22040301152**	elisademir@stu.topkapi.edu.tr
Ayham Elmatar	**22040301011**	ayhamelmatar@stu.topkapi.edu.tr
BiBi Sanam Faizi	**22040301086**	bibisanamfaizi@stu.topkapi.edu.tr

GitHub Deposu:
(https://github.com/evil1341/Semicolon)

Dönem: 2025–2026

Problem Tanımı ve Motivasyon

Bu projenin amacı, Amazon müşterilerinin ürün yorumlarını analiz ederek kullanıcılara kişiselleştirilmiş ürün önerileri yapan hibrit bir yapay zeka sistemi geliştirmektir. Amazon'un All Beauty veri seti, yüz binlerce kullanıcı değerlendirmesi ve ürün bilgisi içerdiği için makine öğrenimi tabanlı bir öneri sistemi geliştirmeye uygundur.

Bu projede Content-Based Filtering ve Collaborative Filtering yaklaşımları birleştirilerek daha yüksek doğruluk sağlayan bir Hybrid Recommendation System oluşturulacaktır.

Bilimsel Soru:
Kullanıcıların geçmiş yorumlarına ve ürün içeriklerine bakarak, en doğru kişiselleştirilmiş ürün önerileri nasıl üretilebilir?

Görev Türü:
Recommender System + NLP

Hedef Değişken:

- Kullanıcı puanı (rating)
- Metin içeriği (text)
- Ürün özellikleri (product_title, main_category)

Başarı Kriterleri:

- F1 Score ≥ 0.80
- Precision, Recall karşılaştırmaları
- Model performans tabloları
- Hybrid modelin tüm modellerden daha yüksek doğruluk vermesi
- Recall@10 ≥ X
- NDCG@10 ≥ X
- RMSE ≤ X (SVD modelleri için)

Proje Yönetimi

Milestones & Timeline (Gantt Tarzı Plan)

Bu proje toplam **7 hafta** üzerinden planlanmıştır.

Hafta 1 (29 Eylül – 5 Ekim)

- Veri setinin seçilmesi
- Proje fikrinin netleştirilmesi
- Görev dağılımının yapılması

Sorumlu: Tüm ekip

Hafta 2 (6–12 Ekim)

- Veri Temizleme (Cleaning)
- Veri Hazırlama (Preparation)
- EDA (Keşifsel Veri Analizi)
- TF-IDF dönüşümü
- Train–Test Split

Sorumlu: *Gays*

Hafta 3

- Baseline modellerinin geliştirilmesi
- Popularity Baseline Model
- Basit Content-Based model (Title bazlı TF-IDF)
- Baseline Logistic Regression (TF-IDF text)

Sorumlu: *Ayham (baseline*

LR) + Elisa (baseline CBF) + Ahmed (popularity)

Hafta 4–5

Farklı model ailelerinin geliştirilmesi

Collaborative Filtering (User–Item Matrix)

- SVD model
- Item-Based KNN
  Sorumlu: Ayham

Content-Based Filtering (TF-IDF Cosine Similarity)

- Ürün başlıklarına göre CBF
- Yorum metnine göre CBF
  Sorumlu: *Elisa*

Hybrid Model (CF + CBF birleştirilmesi)

- Weighted Sum Hybrid
- Hybrid Stacking (Model-level fusion)
  Sorumlu: Gays
  Bibi: Baseline CBF + WordCloud + destek görseller

Hyperparameter Tuning

- SVD latent factors
- KNN k değeri
- TF-IDF max_features
- Hybrid α weight
  Sorumlu:

- Ayham → CF tuning
- Elisa → CBF tuning
- Gays → Hybrid tuning
- Ahmed → Metric comparison

Hafta 6

- Performans analizi
- Modellerin karşılaştırılması
- Hata analizi, Confusion Matrix, ROC Curve
- *Sorumlu:* Elisa + Gays

Hafta 7

- Final raporun hazırlanması
- Sunum slaytlarının oluşturulması
- GitHub düzenlemesi

Sorumlu*:* Tüm ekip

Roles & Responsibilities

Member	Responsibility
Gays Harmuş	Data cleaning, merging, EDA, NLP preprocessing, TF-IDF pipeline
Ahmed Asfour	Baseline Popularity Model, Model evaluation, metrics, confusion matrix, ROC curves, performance comparison, evaluation report
Ayham Elmatar	SVD Collaborative Filtering, Item-Based KNN CF, user–item matrix preparation, hyperparameter tuning
Elisa Demir	Content-Based Filtering (TF-IDF Title), Content-Based Filtering (TF-IDF Review Text), cosine similarity computation
BiBi Sanam Faizi	Baseline CBF model, WordCloud, text visualization, literature review contributions
Whole Team	Final report, presentation slides, GitHub organization, project documentation

Outputs:

- Final Project Report (PDF)
- Jupyter Notebook files
- final_clean_data.csv
- X_train_tfidf.npz, X_test_tfidf.npz
- Model files (.pkl)
- Presentation slides
- EDA charts

İlgili Çalışmalar (Mini Literatür Taraması)

Bu proje, Amazon ürün yorumlarını kullanarak **hibrit bir öneri sistemi** geliştirmeyi hedeflemektedir. Literatürde hem Collaborative Filtering hem de Content-Based Filtering üzerine birçok çalışma yapılmıştır. Aşağıda bu çalışmalardan birkaç örnek verilmiştir:

Çalışma 1 — Amazon Product Recommendation Using Collaborative Filtering

Yöntem: User-Based ve Item-Based CF
Veri: Amazon Reviews (Millions)
Sonuç: CF modelleri benzer kullanıcı davranışlarını yakalamada başarılıdır ancak soğuk başlangıç (cold start) sorununa karşı zayıftır.

Bizim farkımız:
Bu projede CF modeli yalnız başına kullanılmayacak, Content-Based ile birleştirilerek **Hybrid Model** oluşturulacaktır. Böylece cold-start etkisi azaltılır.

Çalışma 2 — Content-Based Recommendation Using TF-IDF and Cosine Similarity

Yöntem: Ürün başlıkları ve açıklamalardan TF-IDF çıkarımı
Veri: Ürün metinleri
Sonuç: Metin içeriği üzerinden yapılan öneriler özellikle yeni ürünlerde daha başarılıdır.

Bizim farkımız:
Biz sadece ürün açıklamalarını değil, **kullanıcı yorumlarını (review text)** de kullanıyoruz. Bu, daha derin içerik anlayışı sağlar.

Çalışma 3 — Hybrid Recommender Systems (CF + CBF)

Yöntem: Skor birleştirme, ağırlıklı ortalama, model stacking
Sonuç: En yüksek performans hibrit sistemlerde görülür.

Bizim farkımız:
Bizim modelimiz 5 farklı alt modelden alınan skorları karşılaştırıp optimize edecek.
Ayrıca TF-IDF + CF kombinasyonu Amazon veri seti üzerinde özel olarak uygulanmıştır.

Bu Projenin Doldurduğu Boşluklar

- Yalnızca CF veya yalnızca CBF yerine **çok modelli (5 model) hibrit sistem** kurulması
- Kullanıcı yorum metninin modele dahil edilmesi
- Büyük veri setinde (600K satır) TF-IDF ve CF birlikte kullanılması
- Ürün benzerliği ile kullanıcı benzerliğinin birlikte analiz edilmesi

Veri Tanımı ve Veri Yönetimi

Veri Seti

asin (kategorik)	Ürün kimliği
product_title (metin)	Ürün adı
main_category (kategorik)	Ürün kategorisi
average_rating (sayısal)	Ürün ortalama puanı
price (sayısal/metin)	Ürün fiyatı (eksik olabilir)
store (metin)	Satıcı bilgisi (eksik olabilir)

Bu projede Amazon'un açık kaynaklı **All Beauty** ürün kategorisine ait yorum ve ürün meta verileri kullanılmıştır.

- **Veri Seti Adı:** Amazon All Beauty Reviews + Metadata
- **Kaynak:** Amazon Product Reviews Dataset
- **Format:** JSONL → CSV
- **Lisans / Kullanım Hakkı:** Akademik amaçlı kullanım için uygundur. Veri anonimdir ve kişisel kimlik bilgisi içermez.

Şema (Schema) ve Değişkenler

1) Reviews (df) veri kümesi:

user_id	Yorum yapan kullancı kimliği
asin	Ürün kimliği
rating	kullanıcı puanı
text	Kullanıcı yorumu
timestamp	Yorum tarihi (ms)

2) Metadata (meta_df) veri kümesi:

Boyut (Size)

Temizleme sonrası birleştirilmiş veri:

- **Satır sayısı:** ~633.000 yorum
- **Sütun sayısı:** 9 (temizlenmiş ve birleştirilmiş)
- **Sınıf dengesi:** Rating dağılımı; 4–5 puan yoğunluğu beklenmektedir (EDA ile doğrulanacaktır).

Veri Erişim ve Saklama Planı

- JSONL dosyaları lokal ortamda okunmuştur.
- Veri temizleme sonrası **final_clean_data.csv** üretilmiştir.
- Model eğitimi için **TF-IDF sparse matrisleri**: `X_train_tfidf.npz`, `X_test_tfidf.npz`
- Veriler ve tüm çıktılar GitHub'a yüklenecektir.

Etik, Gizlilik ve Bias

- Veri seti anonimdir, hassas kişisel veri içermez.
- Kullanıcı kimlikleri `user_id` ile temsil edilmiştir, gerçek ad bulunmaz.
- Olası bias: Ratinglerin çoğunlukla olumlu olması öneri sisteminde popüler ürünlere kayma yaratabilir.
- Bu risk EDA ve değerlendirme aşamasında analiz edilecektir.

Keşifsel Veri Analizi (EDA) Planı

EDA aşaması, veri setinin yapısını anlamak ve sonraki modelleme adımlarında hata riskini azaltmak için kritik öneme sahiptir.

1. Veri Kalite Kontrolleri

Bu aşamada aşağıdaki kontroller yapılacaktır:

- **Eksik değer analizi:** `rating`, `text`, `product_title`, `main_category` gibi sütunlarda eksik veri var mı?

- **Çoğaltılmış kayıtlar (duplicates):** `user_id + asin + timestamp` kombinasyonuna

göre tekrar eden yorumları tespit etme.
- **Tutarsız değerler:** Örneğin, rating sütununun sadece 1–5 arasında olup olmadığı kontrol edilir.
- **Zaman sütunu kontrolü:** `timestamp` değerlerinin doğru şekilde tarihe çevrilebildiğini doğrulama.

2. Dağılımlar ve Denge Analizi

Verinin genel dağılımını anlamak için:

- **Rating dağılım grafiği (Histogram)** Kullanıcıların çoğunlukla hangi puanları verdiğini görme.
- **Yorum uzunluk dağılımı** Metinlerin ortalama uzunluğu, aşırı kısa/uzun yorumlar.

Kategori dağılımı
Hangi kategoriler altında kaç yorum olduğu.

3. Hedef Değişken ile İlişkiler

- **Rating vs. Yorum uzunluğu** Daha uzun yorumlar daha yüksek puanla mı geliyor?
- **Kategoriye göre Rating ortalamaları** Bazı kategori ürünleri daha mı yüksek puan alıyor?
- **Ürün popülerliği** En çok yorum alan ürünlerin analizi.

4. Görselleştirme Planı

EDA sırasında kullanılacak grafikler:

- Histogram (ratings)
- Bar chart (kategori dağılımı)
- Boxplot (rating'e göre yorum uzunluğu)
- WordCloud (en çok geçen kelimeler)
- Time-series plot (zamana göre yorum sayısı)

Bu grafikler final raporun **Appendix** kısmında yer alacaktır

Veri Hazırlama Planı (Data Preparation Plan)

Bu aşama, makine öğrenimi modellerinin doğru şekilde eğitilebilmesi için verinin düzenlenmesini içerir.

1. Temizleme (Cleaning)

Aşağıdaki adımlar uygulanmıştır ve rapora bu şekilde yazılır:

- Eksik değer içeren satırların kaldırılması (`rating`, `text` gibi kritik alanlarda)
- Liste veya sözlük yapısındaki verilerin string'e dönüştürülmesi
- Yinelenen kayıtların (`user_id + asin + timestamp`) silinmesi
- Gereksiz sütunların kaldırılması (`images`, `store`, `price`, `verified_purchase`, vb.)

2. İmputasyon Stratejisi (Missing Value Imputation)

Veri temizliğinde kullanılan strateji:

- Kritik alanlar (rating, text): **dropna** → silindi
- Ürün meta verilerinde eksik olanlar:
  → `"Unknown"` ile dolduruldu (örneğin `main_category`, `product_title`)

Bu seçim veri hatalarını en aza indirir.

3. Dönüşümler (Transformations)

- `rating` sütunu numerik tipe dönüştürüldü
- `timestamp` milisaniyeden datetime formatına çevrildi
- `asin`, `main_category` gibi sütunlar string olarak ayarlandı
- Metin temizleme yapılacak:
  - Küçük harfe çevirme
  - Noktalama işaretlerinin temizlenmesi
  - Stopwords temizliği
  - Lemmatization (gerekirse)

Bu dönüşümler TF-IDF ve ML modelleri için gereklidir.

4. Öznitelik Mühendisliği (Feature Engineering)

Bu projede yapılacak Feature Engineering:

- `review_length` → yorumdaki karakter sayısı
- `word_count` → kelime sayısı
- `rating_normalized` → 1–5 ölçeğinde normalize değer
- TF-IDF matrisleri → metni sayısal vektöre dönüştürme
- Ürün başlıklarından TF-IDF
- Kullanıcı–ürün rating matrisinin oluşturulması (CF modelleri için)

5. Özellik Seçimi (Feature Selection)

Bu projede Feature Selection mantığı:

- TF-IDF için: maksimum 5000 özellik
- Gereksiz sütunlar: tamamen kaldırıldı
- Yalnızca şu sütunlar modellemede kullanılacak:

`user_id/ asin/ rating/ text/ product_title/ main_category`

Bu, modelleri sade ve hızlı yapar.

6. Boyut Azaltma (Dimensionality Reduction) – Opsiyonel

- PCA gibi yöntemler TF-IDF sonrası aşırı maliyetli olacağı için kullanılmayacak.
- Ancak CF ve CBF modelleri boyut azaltma ihtiyacı duymadan çalışır.
  Bu seçim raporda belirtilecektir.

Modelleme Planı

Bu projede **10 farklı model** geliştirilecek ve performansları karşılaştırılacaktır.
Amaç, en doğru öneriyi veren modeli seçmek ve ardından hibrit yapıda birleştirmektir.

8.1 Baseline Modeller

İlk olarak basit referans modeller kurulacaktır:

1. **Popularity Baseline (Popülerlik Modeli)**
    - En çok puan alan veya en çok yorum yapılan ürünleri önerir.
    - Hibrit sistem için karşılaştırma noktasıdır.
2. **Basit TF-IDF + Cosine Similarity**
    - Sadece ürün başlığına göre öneri yapar.
    - Content-based sistemin temel versiyonudur.

8.2 Aday Modeller (10 Model)

Model 1 — Collaborative Filtering (SVD)

- Kullanıcı–ürün puan matrisi üzerinden öğrenir.
- Amaç: kullanıcının geçmiş puanlarına göre yeni ürün tahmini.

Model 2 — Item-Based KNN Collaborative Filtering

- Ürünler arası benzerlik hesaplanır.
- Kullanıcının sevdiği ürünlere benzeyen ürünler önerilir.

Model 3 — Content-Based TF-IDF (product_title)

- Ürün başlıkları TF-IDF ile vektörleştirilir.
- Cosine similarity ile benzer ürünler bulunur.

Model 4 — Content-Based TF-IDF (review text)

- Kullanıcı yorum metinlerinden TF-IDF çıkarılır.
- Kullanıcıların yazdığı içerik üzerinden ürün benzerliği çıkarılır.

Model 5 — Hybrid Recommendation Model

- CF (SVD + KNN) skorları ile CBF (Title + Text) skorları ağırlıklı şekilde birleştirilir.
- Amaç: cold-start ve sparsity problemini azaltmak.

8.3 Hyperparameter Tuning Planı

Her model için tuning yapılacaktır:

- **SVD:** latent factors, learning rate, regularization
- **KNN:** k değeri, similarity metric (cosine/pearson)
- **TF-IDF:** max_features, ngram_range
- **Hybrid:** ağırlık parametresi α (0–1 arası)

Tuning yöntemi:

- Grid Search veya Random Search

5-fold CV ile doğrulama (leakage yok)

Değerlendirme Tasarımı

Bu bölümde 5 modelin nasıl değerlendirileceği ve karşılaştırılacağı tanımlanır.

9.1 Metrikler (Birden Fazla Gerekir)

Bu bir öneri sistemi olduğu için farklı metrik türleri kullanılacaktır:

1) Sınıflandırma Metrikleri

- Precision
- Recall
- F1-Skor
- Accuracy

Model puan tahmini veya "beğendi/beğenmedi" tahmini yapıyorsa kullanılır.

2) Sıralama Metrikleri (Öneri sistemleri için en önemli grup)

- **NDCG@k/ MAP@k/ Hit Rate / Recall@k/ MRR**

Modelin önerdiği ürünleri ne kadar doğru sıraladığını ölçer.

3) Regresyon Metrikleri (SVD gibi modeller için)

- **RMSE/ MAE**

Sürekli puan tahmini yapan modellerde kullanılır.

9.2 Doğrulama Stratejisi

Train/Test Ayırma

Zaten yaptık:

- %80 eğitim + %20 test

Cross-Validation

Overfitting'i azaltmak için:

- **5-fold CV/** Sadece **eğitim verisi üzerinde** (leakage yok)
- → **Veri Sızıntısı Engellenir**
- TF-IDF yalnızca **train** üzerinde fit edilir.
- Özellik seçimi ve scaling **CV pipeline içinde** yapılır.
- Test verisi hiçbir işlemde kullanılmaz.

9.3 Hata Analizi

Sınıflandırma tahmini varsa:

Araçlar:

- Confusion Matrix
- ROC Curve
- PR Curve
- Classification Report

	member	task_type	stage	model	accuracy	f1	precision	recall	rmse	mae
0	Elisa	classification_base_models	Before_FS_DR	NaiveBayes	0.843700	0.901382	0.841343	0.970648	NaN	NaN
1	Elisa	classification_base_models	Before_FS_DR	KNN	0.743500	0.844215	0.763233	0.944422	NaN	NaN
2	Elisa	classification_base_models	After_FS(Chi2)	NaiveBayes	0.844100	0.901795	0.840535	0.972687	NaN	NaN
3	Elisa	classification_base_models	After_FS(Chi2)	KNN	0.761800	0.850283	0.791019	0.919147	NaN	NaN
4	Elisa	classification_base_models	After_FS_DR(SVD)	LogReg	0.846600	0.900892	0.858726	0.947411	NaN	NaN
5	Elisa	classification_base_models	After_FS_DR(SVD)	KNN	0.772800	0.844554	0.850489	0.838701	NaN	NaN
6	Ahmad	classification_base_models	Before_FS_DR	LogReg	0.890965	0.924572	0.907332	0.942479	NaN	NaN
7	Ahmad	classification_base_models	Before_FS_DR	NaiveBayes	0.868454	0.911511	0.871358	0.955543	NaN	NaN
8	Ahmad	classification_base_models	After_FS(MI)	LogReg	0.875674	0.914669	0.890879	0.939764	NaN	NaN
9	Ahmad	classification_base_models	After_FS(MI)	NaiveBayes	0.836601	0.894413	0.825369	0.976064	NaN	NaN
10	Ahmad	classification_base_models	After_FS_DR(SVD)	LogReg	0.850038	0.898051	0.866895	0.931529	NaN	NaN
11	Bibi	classification_base_models	Before_FS_DR	DecisionTree	0.793500	0.859667	0.859842	0.859492	NaN	NaN
12	Bibi	classification_base_models	Before_FS_DR	LogReg	0.866900	0.913307	0.877033	0.952711	NaN	NaN
13	Bibi	classification_base_models	After_FS_RFE	DecisionTree	0.793400	0.859780	0.858847	0.860715	NaN	NaN
14	Bibi	classification_base_models	After_FS_RFE	LogReg	0.863400	0.911160	0.873768	0.951896	NaN	NaN
15	Bibi	classification_base_models	After_FS_DR_PCA	DecisionTree	0.738600	0.822080	0.823537	0.820628	NaN	NaN
16	Bibi	classification_base_models	After_FS_DR_PCA	LogReg	0.850600	0.903538	0.860745	0.950809	NaN	NaN
17	Gays	classification_base_models	Before_FS_DR	LinearSVC	0.881362	0.918112	0.898930	0.938130	NaN	NaN
18	Gays	classification_base_models	Before_FS_DR	SGD_LogReg	0.861803	0.907679	0.862141	0.958297	NaN	NaN
19	Gays	classification_base_models	After_FS(Chi2)	LinearSVC	0.880605	0.917654	0.897799	0.938408	NaN	NaN
20	Gays	classification_base_models	After_FS(Chi2)	SGD_LogReg	0.861306	0.907362	0.861713	0.958119	NaN	NaN
21	Gays	classification_base_models	After_DR(SVD)	LinearSVC	0.844413	0.894564	0.860848	0.931029	NaN	NaN
22	Gays	classification_base_models	After_DR(SVD)	SGD_LogReg	0.836625	0.891522	0.842194	0.946989	NaN	NaN
23	Ayham	collaborative_filtering	CF	SVD_Optimized	NaN	NaN	NaN	NaN	1.404065	1.144130
24	Ayham	collaborative_filtering	CF	ItemKNN_Optimized	NaN	NaN	NaN	NaN	1.440094	1.179558

Sıralama modellerinde ise:

- Modelin yanlış önerdiği ürünler incelenir.
- Düşük puanlı ama önerilen ürünler analiz edilir.
- Kategori bazlı hata dağılımı çıkarılır

Son model seçilirken:

- Genel başarı + Sıralama performansı + CV tutarlılığı + Genel genelleme kabiliyeti değerlendirilir.

Riskler ve Önlemler

10.1 Veri Riskleri

Risk 1 — Veri kümesi çok büyük RAM'i aşabilir

Önlem:

- Yoğun olmayan **sparse npz** formatı kullanıldı.
- Gerekirse veriyi parça parça yükleme.
- Google Colab GPU kullanımı.

Risk 2 — Eksik veya tutarsız ürün bilgileri

Önlem:

- Eksikler "Unknown" ile doldurulduu.
- Kullanılamayan satırlar silindi.

Risk 3 — Rating dağılımı dengesiz

Önlem:

- F1 ve PR-AUC gibi metrikler kullanılacak.
- Gerekirse sınıf dengeleme uygulanacak.

10.2 Yöntemsel Riskler

Risk 4 — Overfitting

Önlem:

- Cross-validation
- Regularization
- Early stopping

Risk 5 — Model eğitimi çok uzun sürebilir

Önlem:

- TF-IDF sözcük sayısının sınırlandırılması
- Optimizasyonlu kütüphaneler kullanımı
- Colab GPU desteği

Risk 6 — Genelleme kötü olabilir

Önlem:

- Katı Train/Test ayrımı
- Veri sızıntısının engellenmesi
- CV ile doğrulama

10.3 Organizasyonel Riskler

Risk 7 — Grup içinde iş yükü dağılmayabilir

Önlem:

- Section 3'te net görev dağılımı yapıldı.

Risk 8 — GitHub dosya boyutu sorunları

Önlem:

- .gitignore ile büyük dosyaların hariç tutulması
- Büyük .npz dosyalarının Drive veya LFS ile paylaşılması

Tekrarlanabilirlik ve Araçlar

Bu bölüm projenin tamamen yeniden üretilebilir olmasını sağlar.

11.1 Ortam

- **Python Sürümü:** 3.11
- **Temel Kütüphaneler:** pandas, numpy, scikit-learn, scipy, matplotlib, surprise
- **Donanım:**
  - Minimum 8 GB RAM
  - Önerilen: Google Colab GPU

11.2 Tekrarlanabilirlik Adımları

1. GitHub deposunu klonla
2. `requirements.txt` dosyasını yükle
3. Notebook dosyalarını sırayla çalıştır:
   - Veri temizleme
   - TF-IDF + Train/Test bölme
   - Model eğitimi
   - Değerlendirme
4. Kaydedilmiş `.npz` ve `.csv` dosyalarını yükle
5. Modelleri yeniden eğit veya `.pkl` olarak yükle

11.3 Kod Yapısı

- project/
  - notebooks/
    - FET445_2204030 1144_Semicolon_ 1.ipynb
  - data/
    - final_clean_data.c sv
    - X_train_tfidf.npz
  - models/
    - svm_model.pkl

README.md

project_report.pdf

11.4 Rastgelelik Kontrolü

- `random_state=42`
- `np.random.seed(42)`
- Modellerde sabit seed

11.5 Çalışma Süreleri

- Veri temizleme: ~3 dk
- TF-IDF dönüşümü: ~8 dk
- Model eğitim süreleri: 1–6 dk arası

Beklenen Sonuçlar ve Görselleştirme Planı

Bu bölümde modellerin üretmesini beklediğimiz sonuçlar ve raporda kullanacağımız grafikler açıklanır.

12.1 Beklenen Çıktılar

Projenin sonunda şu çıktılar oluşturulacaktır:

A. Temizlenmiş Veri Dosyaları

- final_clean_data.csv
- X_train_tfidf.npz
- X_test_tfidf.npz
- y_train.csv, y_test.csv

B. Eğitilmiş Öneri Modelleri

- Collaborative Filtering
- Content-Based (TF-IDF)
- Hybrid model
- Ek sınıflandırma modelleri
- Modeller `.pkl` formatında kaydedilmiş halde

C. Değerlendirme Metriği Tabloları

- Precision
- Recall
- F1-score
- ROC-AUC
- PR-AUC
- RMSE

12.2 Kullanılacak Grafikler

Projede kullanılacak görseller:

1. Rating Dağılım Grafiği

Verideki puanların genel dağılımı.

2. Kelime Bulutu (Word Cloud)

Yorumlarda geçen en sık kelimeler.

3. Confusion Matrix

Sınıflandırma modellerinin hata analizini gösterir.

4. ROC ve PR Eğrileri

Modeller arası performans karşılaştırması.

5. Model Karşılaştırma Tablosu

Bütün metriklerin tek tabloda gösterimi.

6. Feature Importance / SHAP

Model tahminlerini açıklamak için.

7. User–Item Heatmap

Kullanıcı–ürün etkileşim yoğunluğu.

8. TF-IDF Benzerlik Tablosu

Bir ürün için en benzer ürünlerin listesi.

12.3 Yorumlama Planı

Sonuçlar aşağıdaki şekilde yorumlanacaktır:

- En iyi performans veren model hangisi?
- Hybrid model klasik modellere göre ne kadar daha iyi?
- Müşteri davranışında hangi örüntüler bulunuyor?
- Metindeki hangi özellikler en etkili?
- Kategorilere göre öneri kalitesi değişiyor mu?
- Veri dengesizliği veya yanlılık mevcut mu?

Kaynakça

Bu projede kullanılan bilimsel ve teknik kaynaklar aşağıda listelenmiştir.
Kaynaklar **IEEE formatındadır**.

[1] Amazon İnceleme Veri Seti (UCSD / Julian McAuley)

J. McAuley, R. Pandey, J. Leskovec, *"Inferring networks of substitutable and complementary products,"* KDD 2015.
Veri seti: https://cseweb.ucsd.edu/~jmcauley/datasets/amazon_v2/

[2] Scikit-Learn Belgeleri

F. Pedregosa et al., *"Scikit-learn: Machine Learning in Python,"* JMLR, 2011.
Belge: https://scikit-learn.org/

[3] Surprise Kütüphanesi (Collaborative Filtering)

N. Hug, *"Surprise: A Python library for recommender systems,"* 2017.
Belge: https://surpriselib.com/

[4] TF-IDF ve Metin Madenciliği

C. D. Manning, P. Raghavan, H. Schütze, *Introduction to Information Retrieval*, Cambridge Univ. Press, 2008.

[5] Hibrit Öneri Sistemleri Üzerine İnceleme

R. Burke, *"Hybrid Recommender Systems: Survey and Experiments,"* UMUAI, 2002.

[6] SHAP Açıklanabilirlik Framework'ü

S. Lundberg, S.-I. Lee, *"A Unified Approach to Interpreting Model Predictions,"* NIPS 2017.