Proje Adı: Makine Öğrenmesi ile Metin Sınıflandırma

Hazırlayan: Sedanur PEKER

Görev No: 2

Teslim Edilen Kurum: DFA Teknoloji

Teslim Tarihi: 25 Mayıs 2025

İletişim Bilgileri: sedanurpeker5@gmail.com /+90 506 180 4870

İçindekiler

1. Giriş

1.1 Problemin Tanımı

Günümüzde dijital platformlarda üretilen metin miktarı hızla artmakta ve bu durum, metin verilerinin otomatik olarak işlenmesini ve anlamlandırılmasını kritik hale getirmektedir. Haber portalları, sosyal medya ve müşteri geri bildirimleri gibi çok çeşitli kaynaklardan elde edilen metinlerin sınıflandırılması; içerik yönetimi, öneri sistemleri ve bilgi filtreleme gibi alanlarda önemli uygulamalara sahiptir. Bu bağlamda, metin sınıflandırma problemleri, doğal dil işleme (NLP) ve makine öğrenmesi alanlarında yaygın ve etkili bir araştırma konusu haline gelmiştir.

1.2 Proje Seçimi ve Motivasyon

Bu proje, DFA Teknoloji tarafından önerilen altı farklı görevden biri olan "Makine Öğrenmesi ile Metin Sınıflandırma" başlığı altında seçilmiştir. Diğer görevler arasında web tabanlı veri toplama ve fiyat karşılaştırma aracı, dosya yükleme ve yönetimi uygulaması, kira tahminleme sistemi, ocr tabanlı belge bilgi çıkarım sistemi ve blockchain tabanlı basit noter uygulaması yer almaktadır. Doğal dil işleme (NLP) ve makine öğrenmesine olan ilgim, veri analizi ve modelleme konularındaki yetkinliğimi geliştirme hedefim nedeniyle bu görev tercih edilmiştir.

1.3 Projenin Amacı

Bu çalışmanın temel amacı; kamuya açık, kısa haber metinlerinden oluşan bir veri seti üzerinde uygun ön işleme adımlarını uygulayarak, her bir metni ait olduğu kategoriye (örneğin: Spor, Ekonomi, Magazin, Teknoloji) otomatik olarak sınıflandırabilecek bir makine öğrenmesi sistemi geliştirmektir.

1.4 Kullanılan Yöntemler

Geliştirilen sistemde, veri temizleme ve tokenizasyon gibi temel ön işlemler gerçekleştirilmiş, ardından farklı sınıflandırma algoritmaları ile eğitim ve test süreçleri uygulanmıştır. Kullanılan başlıca modeller:

- **Naive Bayes:** Metin sınıflandırmada sıklıkla kullanılan, hızlı ve etkili bir yöntemdir.
- **Logistic Regression:** Özellikle iki veya çoklu sınıflandırma problemlerinde güçlü doğrusal ayrım yeteneğine sahiptir.

- **Support Vector Machine (SVM):** Yüksek boyutlu verilerde etkili olan ve iyi genelleme performansı sağlayan güçlü bir sınıflandırıcıdır.
- **Random Forest:** Karar ağaçlarından oluşan ansambl yöntemiyle, özellikle dengesiz veri setlerinde başarılı sonuçlar sunar.

1.5 Değerlendirme Ölçütleri

Her model; doğruluk oranı, karmaşıklık matrisi ve diğer performans metrikleri ile kıyaslanmıştır. Elde edilen sonuçlar doğrultusunda, sınıflandırma başarısı yüksek, ölçeklenebilir ve anlaşılır bir sistem ortaya konmuştur.

2. Veri Kümesi

Bu çalışmada kullanılan veri seti, Kaggle platformunda yer alan ve kamuya açık olarak sunulan "BBC Full Text Document Classification" veri setidir. Veri kümesi, İngilizce dilinde hazırlanmış 2225 adet kısa haber metninden oluşmaktadır ve haberler beş farklı kategoriye ayrılmıştır:

- Business (Ekonomi)
- Entertainment (Magazin / Eğlence)
- Politics (Gündem / Siyaset)
- Sport (Spor)
- Tech (Teknoloji)

Veri seti her bir metnin tam halini ve ait olduğu sınıf etiketini içermektedir. Dosya yapısı, bbc adlı bir klasör içinde kategori isimlerine göre alt klasörlere ayrılmış şekilde organize edilmiştir. Bu yapı sayesinde veri seti, doğrudan denetimli makine öğrenmesi modellerinde kullanılmaya uygun hale getirilmiştir.

Veri setinin başlıca avantajları şunlardır:

- Haberler kısa ve tek konulu olduğu için sınıflandırma açısından nettir.
- Beş farklı sınıf içermesi, çok sınıflı sınıflandırma algoritmalarının karşılaştırılmasına olanak tanır.
- İngilizce dilinde olması sayesinde pek çok hazır NLP kütüphanesi ile doğrudan çalışabilir.

Veri seti kullanılarak yapılan sınıflandırma işlemleri; metinlerin içeriklerine göre ilgili kategoriye atanmasını ve böylece bilgi organizasyonunun otomatikleştirilmesini sağlamaktadır. Bu yapı, gerçek dünya uygulamaları açısından oldukça değerlidir.

Veri Seti Kaynağı:

Kaggle - Shivam Kushwaha tarafından sağlanan "BBC Full Text Document Classification" veri seti

https://www.kaggle.com/datasets/shivamkushwaha/bbc-full-text-document-classification

3. Veri Ön İşleme ve Özellik Çıkarımı

Ham metin verileri, doğrudan makine öğrenmesi modelleri tarafından işlenemez. Bu nedenle metinler üzerinde çeşitli ön işleme adımları uygulanarak, veriler sayısal özelliklere dönüştürülür. Bu çalışmada aşağıdaki ön işleme adımları uygulanmıştır:

3.1 Temizlik İşlemleri

- **Küçük Harfe Dönüştürme:** Tüm metinler küçük harfe çevrilmiştir. Bu işlem sayesinde “Ekonomi” ve “ekonomi” gibi aynı anlama gelen kelimeler model tarafından farklı olarak algılanmaz.
- **Noktalama İşaretlerinin ve Sayıların Kaldırılması:** Metinlerden noktalama işaretleri (., ,, ?, ! vb.) ve sayılar çıkarılmıştır. Bu karakterler genellikle anlam taşımaz ve sınıflandırma modeline katkı sağlamaz.
- **Stopword Temizliği**: *the, and, is, in* gibi sık geçen ama çok az bilgi taşıyan kelimeler (stopwords) çıkarılmıştır. Bu sayede modelin önemli kelimelere odaklanması sağlanmıştır.
- **Boşluk ve Satır Temizliği:** Fazladan boşluklar, tab karakterleri ve gereksiz satır başı karakterleri temizlenmiştir.

Bu işlemler sonucunda verinin daha saf ve bilgi açısından yoğun hali elde edilmiştir.

3.2 Tokenizasyon ve Vektörleştirme

Ön işlemden geçirilen metinler, makine öğrenmesi modellerinin işleyebileceği sayısal formatlara dönüştürülmüştür. Bu işlem için aşağıdaki yöntem kullanılmıştır:

TF-IDF (Term Frequency – Inverse Document Frequency)

TF-IDF, metinlerdeki kelimelerin önemini belirlemek için kullanılan istatistiksel bir tekniktir. Sık kullanılan ama genel geçer olmayan kelimelere daha yüksek ağırlık verir. Bu sayede model, metne özgü anlamlı kelimelere daha fazla önem verir. TfidfVectorizer sınıfı kullanılarak uygulanmıştır.

Bu yöntemde her kelimenin ağırlığı aşağıdaki formül ile hesaplanır:

$$TF\text{-}IDF(t,d) = TF(t,d) \times IDF(t)$$

- **TF (Term Frequency):** Bir terimin ilgili belgede kaç defa geçtiğini ifade eder. Normalleştirilmiş hali:

$$TF(t,d) = f_{t,d} / \sum_k f_{k,d}$$

Burada ft,df_{t,d}ft,d, ttt teriminin ddd belgesindeki frekansıdır.

- **IDF (Inverse Document Frequency):** Terimin tüm belgeler arasında ne kadar yaygın olduğunu ölçer:

$$IDF(t) = \log(N / (1 + n_t)$$

Burada N, toplam belge sayısı, n_t ise terimin geçtiği belge sayısıdır.

TF-IDF, çok sık geçen ama anlam açısından düşük öneme sahip ("ve", "ile", "bu" gibi) kelimelere düşük ağırlık verirken, nadir ve ayırt edici terimlere daha yüksek ağırlık verir.

TF-IDF dönüşümünden sonra her metin, yüzlerce boyuttan oluşan bir vektörle temsil edilir. Bu vektörler daha sonra sınıflandırma modellerinin giriş verisi olarak kullanılır.

4. Modelleme ve Performans Değerlendirmesi

Bu bölümde, verinin ön işlenmesinden sonra TF-IDF ile dönüştürülmüş metin özellikleri kullanılarak dört farklı makine öğrenmesi sınıflandırma algoritması ile modeller kurulmuş ve karşılaştırılmıştır. Amaç, metinleri doğru kategorilere ayırmada hangi modelin daha başarılı olduğunu ortaya koymaktır.

4.1 Kullanılan Modellerin Tanıtımı

Bu projede dört farklı sınıflandırma algoritması kullanılmıştır:

- **Multinomial Naive Bayes (MNB):**
  Metin sınıflandırmada klasik ve etkili bir yöntemdir. Özellikle kelime frekansına dayalı olan bu model, metin verileri için oldukça hızlı çalışır.
- **Logistic Regression (LR):**
  TF-IDF ile elde edilen yüksek boyutlu verilerde oldukça başarılıdır. Lineer karar sınırları çizer ve olasılıksal sınıflandırma yapar.
- **Support Vector Machine (SVM):**
  Özellikle lineer olarak ayrılabilen veri kümelerinde yüksek doğruluk sağlar. Metin verilerinde sık kullanılan bir yöntemdir. LinearSVC kullanılmıştır.
- **Random Forest (RF):**
  Karar ağaçlarının topluluğudur. Karmaşık ilişkileri modellemede başarılıdır. Ancak metin sınıflandırma gibi yüksek boyutlu problemler için daha fazla kaynak tüketebilir.

4.2 Eğitim ve Test Süreci

- Veri, train_test_split() fonksiyonu ile %80 eğitim, %20 test olarak ikiye ayrılmıştır. Her model eğitim verisiyle eğitilmiş ve test verisiyle doğrulanmıştır.

4.3 Performans Ölçütleri

Her bir model için aşağıdaki metrikler hesaplanmıştır:

- **Accuracy (Doğruluk):**
  Tüm test örnekleri içinde doğru tahmin edilenlerin oranı.
- **Confusion Matrix (Karmaşıklık Matrisi):**
  Hangi sınıfın ne kada r doğru/yanlış tahmin edildiğini gösterir.
- **Classification Report:**
  Her sınıf için precision, recall ve f1-score değerlerini içerir.

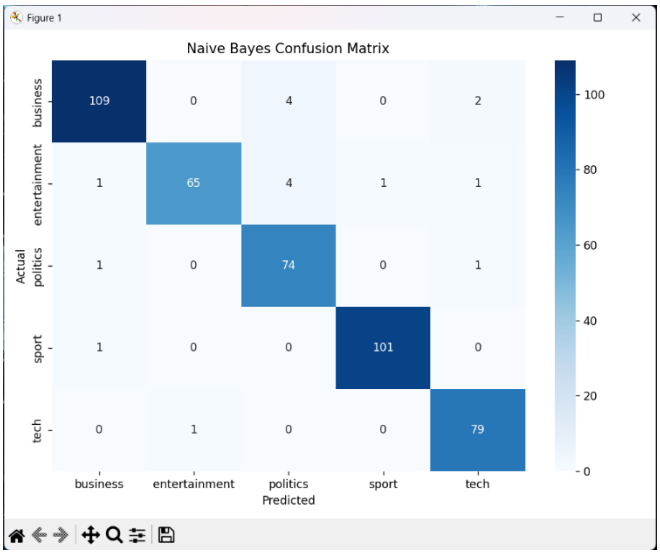
4.4 Model Sonuçları

- **Multinomial Naive Bayes**

```
========== Naive Bayes ==========
Accuracy: 0.9618
Classification Report:
               precision    recall  f1-score   support

     business       0.97      0.95      0.96       115
entertainment       0.98      0.90      0.94        72
     politics       0.90      0.97      0.94        76
        sport       0.99      0.99      0.99       102
         tech       0.95      0.99      0.97        80

     accuracy                           0.96       445
    macro avg       0.96      0.96      0.96       445
 weighted avg       0.96      0.96      0.96       445
```

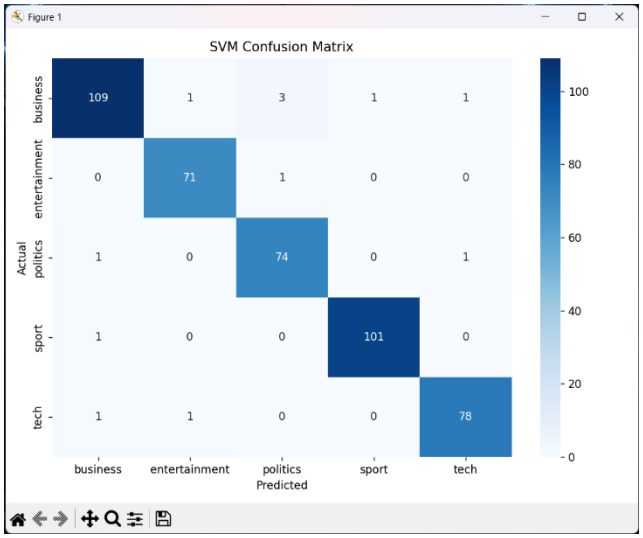


- **Support Vector Machine (SVM)**

```
========== SVM ==========
Accuracy: 0.9730
Classification Report:
               precision    recall  f1-score   support

     business       0.97      0.95      0.96       115
entertainment       0.97      0.99      0.98        72
     politics       0.95      0.97      0.96        76
        sport       0.99      0.99      0.99       102
         tech       0.97      0.97      0.97        80

     accuracy                           0.97       445
    macro avg       0.97      0.97      0.97       445
 weighted avg       0.97      0.97      0.97       445
```

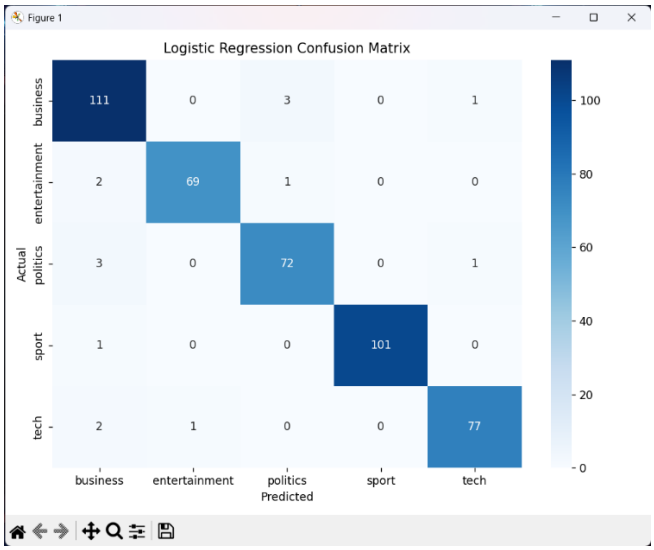


- **Logistic Regression**

```
========== Logistic Regression ==========
Accuracy: 0.9663
Classification Report:
              precision    recall  f1-score   support

    business       0.93      0.97      0.95       115
entertainment       0.99      0.96      0.97        72
    politics       0.95      0.95      0.95        76
       sport       1.00      0.99      1.00       102
        tech       0.97      0.96      0.97        80

    accuracy                           0.97       445
   macro avg       0.97      0.96      0.97       445
weighted avg       0.97      0.97      0.97       445
```

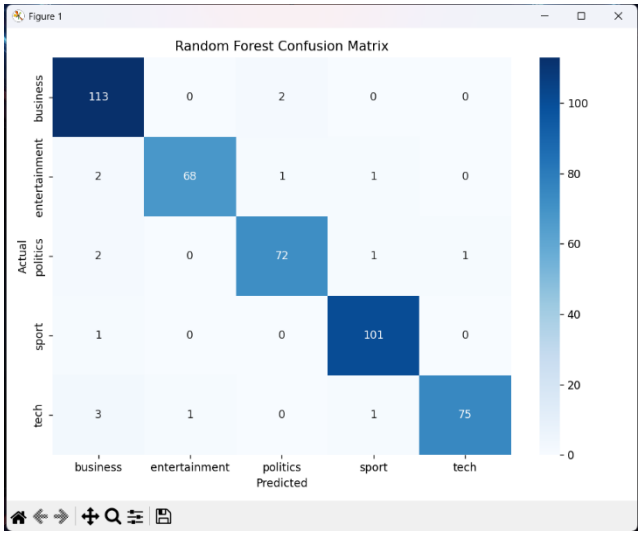


- **Random Forest**

```
========== Random Forest ==========
Accuracy: 0.9640
Classification Report:
              precision    recall  f1-score   support

    business       0.93      0.98      0.96       115
entertainment       0.99      0.94      0.96        72
    politics       0.96      0.95      0.95        76
       sport       0.97      0.99      0.98       102
        tech       0.99      0.94      0.96        80

    accuracy                           0.96       445
   macro avg       0.97      0.96      0.96       445
weighted avg       0.96      0.96      0.96       445
```



4.5 Modelerin Karşılaştırılması

Model	Accuracy (%)	Avantajlar	Dezavantajlar
Naive Bayes	96.18	Çok hızlı, düşük kaynak tüketimi, basit ve etkili	"entertainment" gibi sınıflarda düşük recall görülebilir.
Logistic Regression	96.63	Genelleme yeteneği yüksek, dengeli performans	Çok büyük verilerde eğitim süresi uzayabilir.
SVM (LinearSVC)	97.3	En yüksek doğruluk, özellikle "entertainment" ve "sport" sınıflarında mükemmel performans	Eğitim süresi diğerlerine göre daha uzun olabilir.
Random Forest	96.4	Doğru sınıflandırma oranı yüksek, overfitting'e karşı dirençli	Yüksek boyutlu TF-IDF verilerinde daha yavaş ve kaynak tüketimi fazla olabilir.

5. Sonuç ve Değerlendirme

Bu projede, BBC haber veri seti kullanılarak metin sınıflandırma problemi üzerine dört farklı makine öğrenmesi algoritması test edilmiştir: Naive Bayes, Logistic Regression, Support Vector Machine (SVM) ve Random Forest. Uygulanan ön işleme adımlarının ardından her model eğitilip test edilmiş ve doğruluk oranları ile sınıflandırma raporları üzerinden değerlendirme yapılmıştır.

Elde edilen sonuçlara göre en yüksek doğruluğa ulaşan model SVM olmuştur (%97.30). Özellikle entertainment ve sport kategorilerinde oldukça yüksek precision ve recall değerleri göstermiştir. Logistic Regression modeli de genel doğruluk açısından güçlü bir performans ortaya koymuş (%96.63) ve sınıflar arasında dengeli sonuçlar vermiştir. Random Forest, doğruluğu yüksek olmasına rağmen (%96.40), işlem süresi ve kaynak kullanımı açısından daha maliyetli bir model olarak değerlendirilmiştir. En basit ve en hızlı model olan Naive Bayes ise diğer modellere kıyasla biraz daha düşük doğruluk (%96.18) göstermiştir ancak bu fark oldukça küçük olup modelin verimli çalıştığını göstermektedir.

Genel olarak, metin sınıflandırma gibi doğal dil işleme problemleri için iyi bir ön işleme süreciyle birlikte klasik makine öğrenmesi yöntemlerinin bile oldukça başarılı sonuçlar verdiği görülmüştür. Bu projede, özellikle SVM ve Logistic Regression modellerinin, küçük ve dengeli veri setlerinde yüksek doğruluk sağlayabildiği, pratikte kullanılabilir modeller olduğu ortaya konmuştur.

6. Kaynakça

1. Kushwaha, S. (2019). *BBC Full Text Document Classification Dataset*. Kaggle. Erişim Tarihi: 21 Mayıs 2025.
2. Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (2nd ed.)*. O'Reilly Media.
3. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, 12, 2825–2830.
4. Jurafsky, D., & Martin, J. H. (2021). *Speech and Language Processing* (3rd ed. draft). Stanford University. https://web.stanford.edu/~jurafsky/slp3/
5. scikit-learn documentation. *Classification Metrics*. https://scikit-learn.org/stable/modules/model_evaluation.html
6. Akay, M. F. (2020). *Makine Öğrenmesi ve Veri Madenciliği Yöntemleriyle Sınıflandırma ve Tahminleme*. Sakarya Üniversitesi Yayınları.
7. Özdemir, M. (2016). *Veri Madenciliği: Kavramlar, Yöntemler, Uygulamalar*. Papatya Yayıncılık.
8. Akyokuş, S. (2022). "Doğal Dil İşleme ile Metin Sınıflandırması ve Uygulama Örneği", *Mühendislik Bilimleri ve Tasarım Dergisi*, 10(2), 732–743.