



**DENETİMLİ MAKİNE ÖĞRENMESİ ALGORİTMALARI İLE
TÜRKÇE SAHTE HABER TESPİTİ İÇİN BİR KARAR DESTEK SİSTEMİ**

Yasin ERDURAN

**YÜKSEK LİSANS
YÖNETİM BİLİŞİM SİSTEMLERİ ANA BİLİM DALI**

GAZİ ÜNİVERSİTESİ

BİLİŞİM ENSTİTÜSÜ

ARALIK 2022

Yasin ERDURAN tarafından hazırlanan “DENETİMLİ MAKİNE ÖĞRENMESİ ALGORİTMALARI İLE TÜRKÇE SAHTE HABER TESPİTİ İÇİN BİR KARAR DESTEK SİSTEMİ” adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile Gazi Üniversitesi Yönetim Bilişim Sistemleri Ana Bilim Dalında YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Danışman: Prof. Dr. Cevriye GENCER
Endüstri Mühendisliği, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....

Başkan: Prof. Dr. Hacer KARACAN
Bilgisayar Mühendisliği, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....

Üye: Doç. Dr. Muhammed Hanefi CALP
Yönetim Bilişim Sistemleri, Ankara Hacı Bayram Veli Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....

Tez Savunma Tarihi: 17/01/2023

Jüri tarafından kabul edilen bu tezin Yüksek Lisans Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

.....

Prof. Dr. Aslıhan TÜFEKÇİ
Bilişim Enstitüsü Müdürü

ETİK BEYAN

Gazi Üniversitesi Bilişim Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Yasin Erduran
17/01/2023

DENETİMLİ MAKİNE ÖĞRENMESİ ALGORİTMALARI İLE
TÜRKÇE SAHTE HABER TESPİTİ İÇİN BİR KARAR DESTEK SİSTEMİ
(Yüksek Lisans Tezi)

Yasin Erduran

GAZİ ÜNİVERSİTESİ

BİLİŞİM ENSTİTÜSÜ

Aralık 2022

ÖZET

Sosyal medya ve çevrim içi platformlar aracılığı ile her geçen gün daha fazla habere çok daha hızlı ulaşmaktayız. Bu haberlerin birçoğu bizim için önemli ve faydalı bilgiler içerirken, içlerinden bazı haberler bireye ve topluma zarar verme amacı ile yayılmaktadır. Bu haberler gerçekte hiçbir bağı olmayan, bir kişi ya da kurumun itibarını zedelemeyi, bir ideolojiyi desteklemeyi, kar sağlamayı ve en tehlikelisi alıcısının fikir ve düşüncelerini manipüle etmeyi amaçlayan sahte haberlerdir. Özellikle toplumun bilgi ihtiyacının yüksek olduğu doğal afet, seçim ya da kriz dönemlerinde sosyal medya üzerinden sahte haber yayılımı artmaktadır. Ortaya çıkan enformasyon insan gücüyle teyit edilebilecek boyutları geçmekte ve sahte haberin ortaya çıkardığı hasar büyümektedir. Sahte haberlerin doğuracağı zararların minimize edilmesi için bilgisayar bilim dalı olan makine öğrenmesi algoritmaları kullanılabilir. Makine Öğrenmesi, verilerden öğrenebilen, deneyimlerden bilgi edinebilen ve zaman içinde öğrenme davranışlarını iyileştirebilen algoritmalar. Çalışmada, doğrudan teyit uzmanları aracılığı ile sahte ve gerçek olarak etiketlenmiş Türkçe haber veri seti oluşturulmuş; bu haberler BoW, TF-IDF ve Doc2Vec özellik çıkarım algoritmaları ile öznitelik vektörlerine dönüştürülmüş; MLP, DT, SVM ve LGR makine öğrenme algoritmaları bu öznitelik vektörleri ile eğitilerek, ortaya çıkan 12 farklı modelin doğrulukları Test/Eğitim ayrımı ve K-fold yöntemleri ile denetlenmiştir. Denemeler için 8 çekirdek işlemcili, 16 GB bellekli bir bilgisayar kullanılmış olup, Python programa dili ile kodlanmış ve makine öğrenmesi modelleri için SciKitLearn kütüphanesi kullanılmıştır, en iyi sonuç TF-IDF özellik çıkartımı vektörleri ile beslenen LGR modeli olduğu tespit edilmiştir. Test/Eğitim ayrımı ile elde edilen sonuçların geçerliliği K-fold doğrulama yöntemi ile test edilmiştir. En iyi sonuç veren TF-IDF özellik çıkarımı algoritmasının vektörleri ile beslenen LGR makine öğrenmesi modeli ve diğer elde edilen modeller ile Türkçe sahte haber tespitine yönelik web tabanlı karar destek sistemi oluşturulmuştur.

Bilim Kodu : 114606

Anahtar Kelimeler : Sahte Haber, Dezenformasyon, Makine Öğrenmesi, TF-IDF, BoW, Doc2Vec, Karar Destek Sistemi

Sayfa Adedi : 104

Danışman : Prof. Dr. Cevriye GENCER

A DECISION SUPPORT SYSTEM FOR FAKE NEWS DETECTION IN TURKISH LANGUAGE WITH SUPERVISED MACHINE LEARNING ALGORITHMS

GAZİ UNIVERSITY

INSTITUTE OF INFORMATICS

Aralık 2022

ABSTRACT

The number of news is increasing day by day through social media and online platforms. While some of this news is beneficial and true, other news is destructive and fake. Fake news having no connection with reality, aims to damage the reputation of a person or institution, to support an ideology, to make profit and most dangerously to manipulate the ideas and thoughts of receivers. Fake news spreads more widely on social media during natural disasters, elections, and other times of crisis when there is a great need for knowledge in society. Truth of news cannot be verified by human power because of exponential growth of news, therefore the damage caused by fake news is expanding. Machine learning algorithms, a branch of computer science, can be used to minimize the damage caused by fake news. Machine Learning is algorithms that can learn from data, learn from experience, and improve learning behaviors over time. In the study, a Turkish news data set labeled as fake and real was created by direct factcheckers; these articles were converted into feature vectors with BoW, TF-IDF and Doc2Vec feature extraction algorithms; MLP, DT, SVM and LGR machine learning algorithms were trained with these feature vectors, and the accuracy of 12 different models were checked with Test/Training separation and K-fold methods. A computer with 8 core processors and 16 GB memory was used for the experiments, coded in Python programming language and SciKitLearn library was used for machine learning models, the best result was LGR fed with TF-IDF feature extraction vectors in Test/Training separation and K-fold validation method. A decision support system was developed for the study. In this decision support system, TF-IDF and LGR, which are the most successful models, were used. In addition to this model, other models were presented to the decision maker's preference on the decision support system.

Science Code : 114606

Key Words : Fake News, Disinformation, Machine Learning, TF-IDF, BoW, Doc2Vec, Decision Support System

Page Number : 104

Supervisor : Prof. Dr. Cevriye Gencer



İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT.....	v
İÇİNDEKİLER	vi
ÇİZELGELERİN LİSTESİ.....	ix
ŞEKİLLERİN LİSTESİ.....	xii
RESİMLERİN LİSTESİ	xiv
SİMGELER VE KISALTMALAR.....	xv
1. GİRİŞ.....	1
2. DEZENFORMASYON, SAHTE HABER TESPİTİ, MAKİNE ÖĞRENİMİ VE KARAR DESTEK SİSTEMİ LİTERATÜR ARAŞTIRMASI.....	4
2.1. Dezenformasyon Nedir?	4
2.2. Sahte Haber	8
2.3. Sahte Haberin Makine Öğrenmesi Algoritmaları ile Tespiti	10
2.3.1. Özellik çıkarma algoritmaları.....	13
2.3.1.1. Kelime çantası (BoW) algoritması.....	13
2.3.1.2. TF-IDF algoritması	14
2.3.1.3. Doc2Vec algoritması.....	14
2.3.2. Makine öğrenmesi algoritmaları	15
2.3.2.1. Çok katmanlı yapay sinir ağı (MLP).....	15
2.3.2.2. Karar ağaçları (DT)	15
2.3.2.3. Destek vektör makinaları (SVM).....	15
2.3.2.4. Lojistik regresyon (LGR).....	16

	Sayfa
2.3.3. Makine öğrenmesi modellerin doğrulanması.....	16
2.3.3.1. Eğitim/Test ayrımı	16
2.3.3.2. K katlı çapraz geçerlilik	17
2.3.4. F1, duyarlılık ve kesinlik performans değerleri.....	17
2.3.4.1. Doğruluk değeri (Accuracy)	19
2.3.4.2. Duyarlılık değeri (Precision).....	19
2.3.4.3. Kesinlik değeri (Recall)	19
2.3.4.4. F1 değeri	20
2.4. Karar Destek Sistemi.....	20
3. TÜRKÇE HABER METİNLERİNİN SINIFLANDIRILMASI	21
3.1. Türkçe Veri Setinin Oluşturulması.....	22
3.2. Metinlerin Ön İşlenmesi	25
3.2.1. Noktalama işaretleri ve rakamların kaldırılması	26
3.2.2. Büyük ve küçük harflerin tekdüze hale getirilmesi	26
3.2.3. Cümlelerin kelime boyutunda parçalanması	27
3.2.4. Kelimelerin köklerinin bulunması ve eklerinin atılması	27
3.3. Metinlerden Özellik Çıkarma	28
3.4. Modellerin Oluşturulması	29
3.5. Model Performanslarının Yorumlanması	30
3.6. Bulguların Yorumlanması.....	30
3.6.1. BoW ve MLP modeli	31
3.6.2. BoW ve DT modeli	34
3.6.3. BoW ve SVM modeli	37
3.6.4. BoW ve LGR modeli	40

	Sayfa
3.6.5. TF-IDF ve MLP modeli	43
3.6.6. TF-IDF ve DT modeli	46
3.6.7. TF-IDF ve SVM modeli	49
3.6.8. TF-IDF ve LGR modeli	52
3.6.9. Doc2Vec ve MLP modeli	55
3.6.10. Doc2Vec ve DT modeli	58
3.6.11. Doc2Vec ve SVM modeli	61
3.6.12. Doc2Vec ve LGR modeli	64
4. KARAR DESTEK SİSTEMİNİN GELİŞTİRİLMESİ	67
5. TARTIŞMA VE BULGULAR	71
6. SONUÇ VE ÖNERİLER	77
KAYNAKLAR	79
ÖZGEÇMİŞ	83

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 2.1.1. Dezenformasyonun yayılmasının motivasyonları ve tanımları	7
Çizelge 2.3.4.1. Karmaşıklık Matrisi	18
Çizelge 3.1.1. Mevcut veri setlerinin haber içeriğine göre kıyaslanması	23
Çizelge 3.1.2. Mevcut veri setlerinin sosyal bağlama göre kıyaslanması.....	24
Çizelge 3.1.3. Mevcut veri setlerinin mekân zamansal olarak kıyaslanması.....	24
Çizelge 3.1.4 FakeNewsNet Veri setinin haber içeriği sosyal bağlam ve mekân zamansal veri sayıları	25
Çizelge 3.6.1.1. Test/Eğitim ayrımı ile elde edilen eğitim ve test verisi ile BoW Özellik çıkartımı ve MLP makine öğrenmesi yönteminin 30 iterasyon sonuçları	31
Çizelge 3.6.1.2. BoW Özellik çıkartımı ve MLP makine öğrenmesi yönteminin K-fold sonucu.	32
Çizelge 3.6.2.1. Test/Eğitim yöntemi ile elde edilen eğitim ve test verisi ile BoW özellik çıkartımı ve DT makine öğrenmesi yönteminin 30 iterasyon sonuçları ..	34
Çizelge 3.6.2.2. BoW Özellik çıkartımı ve DT makine öğrenmesi yönteminin K-fold sonucu.....	35
Çizelge 3.6.3.1. Test/Eğitim yöntemi ile elde edilen eğitim ve test verisi ile BoW Özellik çıkartımı ve SVM makine öğrenmesi yönteminin 30 iterasyon sonuçları....	37
Çizelge 3.6.3.2. BoW Özellik çıkartımı ve SVM makine öğrenmesi yönteminin K-fold sonucu.....	38
Çizelge 3.6.4.1. Test/Eğitim yöntemi ile elde edilen eğitim ve test verisi ile BoW Özellik çıkartımı ve LGR makine öğrenmesi yönteminin 30 iterasyon sonuçları.....	40
Çizelge 3.6.4.2. BoW Özellik çıkartımı ve LGR makine öğrenmesi yönteminin K-fold sonucu.....	40
Çizelge 3.6.5.1. Test/Eğitim yöntemi ile elde edilen eğitim ve test verisi ile TF-IDF Özellik çıkartımı ve MLP makine öğrenmesi yönteminin 30 iterasyon sonuçları.....	43
Çizelge 3.6.5.2. TF-IDF Özellik çıkartımı ve MLP makine öğrenmesi yönteminin K-fold sonucu.....	44
Çizelge 3.6.6.1. Test/Eğitim yöntemi ile elde edilen eğitim ve test verisi ile TF-IDF Özellik çıkartımı ve DT makine öğrenmesi yönteminin 30 iterasyon Sonuçları.....	46

Çizelge	Sayfa
Çizelge 3.6.6.2. TF-IDF Özellik çıkartımı ve DT makine öğrenmesi yönteminin K-fold sonucu.....	47
Çizelge 3.6.7.1. Test/Eğitim yöntemi ile elde edilen eğitim ve test verisi ile TF-IDF Özellik çıkartımı ve SVM makine öğrenmesi yönteminin 30 iterasyon sonuçları	49
Çizelge 3.6.7.2. TF-IDF Özellik çıkartımı ve SVM makine öğrenmesi yönteminin K-fold sonucu.	50
Çizelge 3.6.8.1. Test/Eğitim yöntemi ile elde edilen eğitim ve test verisi ile TF-IDF Özellik çıkartımı ve LGR makine öğrenmesi yönteminin 30 iterasyon sonuçları	52
Çizelge 3.6.8.2. TF-IDF Özellik çıkartımı ve LGR makine öğrenmesi yönteminin K-fold sonucu	53
Çizelge 3.6.9.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile Doc2Vec ve MLP makine öğrenmesi yönteminin 30 iterasyon sonuçları.....	55
Çizelge 3.6.9.2. Doc2Vec Özellik çıkartımı ve MLP makine öğrenmesi yönteminin K-fold sonucu.	56
Çizelge 3.6.10.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile Doc2Vec ve DT makine öğrenmesi yönteminin 30 iterasyon sonuçları.....	58
Çizelge 3.6.10.2. Doc2Vec Özellik çıkartımı ve DT makine öğrenmesi yönteminin K-fold sonucu.	59
Çizelge 3.6.11.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile Doc2Vec ve SVM makine öğrenmesi yönteminin 30 iterasyon sonuçları.	61
Çizelge 3.6.11.2. Doc2Vec Özellik çıkartımı ve SVM makine öğrenmesi yönteminin K-fold sonucu	62
Çizelge 3.6.12.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile Doc2Vec ve LGR makine öğrenmesi yönteminin 30 iterasyon sonuçları.....	64
Çizelge 3.6.12.2. Doc2Vec Özellik çıkartımı ve LGR makine öğrenmesi yönteminin K-fold sonucu	65
Çizelge 5.1. Test/Eğitim ayrımı yöntemi ile elde 12 modelin 30, 60 ve 90 iterasyonun ortalama F1 skoru.	72
Çizelge 5.2. Test Eğitim ayrımı ile elde edilen modellerin 30 iterasyonda elde ettiği F1 Skorunun ortalama, standart sapma ve maksimum değerleri	73
Çizelge 5.3. Test/Eğitim ayrımı ile elde edilen modellerin 30 iterasyonda elde ettiği duyarlılık Skorunun ortalama, standart sapma ve maksimum değerleri	74

Çizelge	Sayfa
Çizelge 5.4. Test/Eğitim ayrımı ile elde edilen Modellerin 30 iterasyonda elde ettiği Kesinlik Skorunun ortalama, standart sapma ve maksimum değerleri.	75
Çizelge 5.4. Test/Eğitim ayrımı ile elde edilen Modellerin 30 iterasyonda elde ettiği Kesinlik Skorunun ortalama, standart sapma ve maksimum değerleri.	75
Çizelge 5.5. Test/Eğitim yöntemi ile elde edilen 12 modelin 30 iterasyonda elde ettiği test skorunun ortalama, standart sapma ve maksimum değerleri.	75



ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1.1. Türkiye’de haberlere erişim kaynakları.....	5
Şekil 2.1.2. Türkiye’de tercih edilen sosyal medya mecraları.....	5
Şekil 2.3.1. Sahte haber tespit teknikleri.....	12
Şekil 2.2.2. Sınıflandırma yaklaşımı için özellik çıkarım yöntemleri	13
Şekil 2.3.3.1.1. Veri setinin Test/Eğitim ayrımı yöntemine göre ayrımı.....	16
Şekil 2.3.3.1.2. Veri setinin K-fold yöntemine göre ayrımı	17
Şekil 3.1. Çalışma metodunun adımları.....	21
Şekil 3.2. Çalışma metodunun adımlarının detaylı gösterimi.....	22
Şekil 3.2.1. Çalışmada kullanılan metin ön işleme metotlarının örnek ile açıklanması .	26
Şekil 3.3.1. Metinden özellik çıkarımı ve kelime vektörlerinin elde edilmesi	29
Şekil 3.4.1 Makine Öğrenmesi Modellerinin Elde Edilmesi	30
Şekil 3.6.1.1. BoW Özellik çıkartımı ve MLP makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının kutu grafiği.	32
Şekil 3.6.1.2. BoW Özellik çıkartımı ve MLP makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının histogram grafiği.	33
Şekil 3.6.1.3 BoW Özellik çıkartımı ve MLP makine öğrenmesi yöntemi 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.....	33
Şekil 3.6.2.1. BoW Özellik çıkartımı ve DT makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının kutu grafiği.	35
Şekil 3.6.2.2. BoW Özellik çıkartımı ve DT makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının histogram grafiği.	36
Şekil 3.6.2.3 BoW Özellik çıkartımı ve DT makine öğrenmesi yöntemi 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.....	36
Şekil 3.6.3.1. BoW Özellik çıkartımı ve SVM makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının kutu grafiği.	38
Şekil 3.6.3.2. BoW Özellik çıkartımı ve SVM makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının histogram grafiği.	39

Şekil	Sayfa
Şekil 3.6.3.3 BoW Özellik çıkartımı ve SVM makine öğrenmesi yöntemi 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.	39
Şekil 3.6.4.1. BoW Özellik çıkartımı ve LGR makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının kutu grafiği.	41
Şekil 3.6.4.2. BoW Özellik çıkartımı ve LGR makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının histogram grafiği.	42
Şekil 3.6.4.3 BoW Özellik çıkartımı ve LGR makine öğrenmesi yöntemi 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.	42
Şekil 3.6.5.1. TF-IDF Özellik çıkartımı ve MLP makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının kutu grafiği.	44
Şekil 3.6.5.2. TF-IDF Özellik çıkartımı ve MLP makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının histogram grafiği.	45
Şekil 3.6.5.3 TF-IDF Özellik çıkartımı ve MLP makine öğrenmesi yöntemi 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.	45
Şekil 3.6.6.1. TF-IDF Özellik çıkartımı ve DT makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının kutu grafiği.	47
Şekil 3.6.6.2. TF-IDF Özellik çıkartımı ve DT makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının histogram grafiği.	48
Şekil 3.6.6.3 TF-IDF Özellik çıkartımı ve DT makine öğrenmesi yöntemi 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.	48
Şekil 3.6.7.1. TF-IDF Özellik çıkartımı ve SVM makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının kutu grafiği.	50
Şekil 3.6.7.2. TF-IDF Özellik çıkartımı ve SVM makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının histogram grafiği.	51
Şekil 3.6.7.3 TF-IDF Özellik çıkartımı ve SVM makine öğrenmesi yöntemi 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.	51
Şekil 3.6.8.1. TF-IDF Özellik çıkartımı ve LGR makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının kutu grafiği.	53
Şekil 3.6.8.2. TF-IDF Özellik çıkartımı ve LGR makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının histogram grafiği.	54
Şekil 3.6.8.3 TF-IDF Özellik çıkartımı ve LGR makine öğrenmesi yöntemi 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.	54

Şekil	Sayfa
Şekil 3.6.9.1. Doc2Vec Özellik çıkartımı ve MLP makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının kutu grafiği.	56
Şekil 3.6.9.2. Doc2Vec Özellik çıkartımı ve MLP makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının histogram grafiği.	57
Şekil 3.6.9.3 Doc2Vec Özellik çıkartımı ve MLP makine öğrenmesi yöntemi 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.	57
Şekil 3.6.10.1. Doc2Vec Özellik çıkartımı ve DT makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının kutu grafiği.	59
Şekil 3.6.10.2. Doc2Vec Özellik çıkartımı ve DT makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının histogram grafiği.	60
Şekil 3.6.10.3 Doc2Vec Özellik çıkartımı ve DT makine öğrenmesi yöntemi 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.	60
Şekil 3.6.11.1. Doc2Vec Özellik çıkartımı ve SVM makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının kutu grafiği.	62
Şekil 3.6.11.2. Doc2Vec Özellik çıkartımı ve SVM makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının histogram grafiği.	63
Şekil 3.6.11.3 Doc2Vec Özellik çıkartımı ve SVM makine öğrenmesi yöntemi 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.	63
Şekil 3.6.12.1. Doc2Vec Özellik çıkartımı ve LGR makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının kutu grafiği.	65
Şekil 3.6.12.2. Doc2Vec Özellik çıkartımı ve LGR makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının histogram grafiği.	66
Şekil 3.6.12.3 Doc2Vec Özellik çıkartımı ve LGR makine öğrenmesi yöntemi 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.	66
Şekil 4.1. Karar destek sisteminin makine öğrenmesi algoritması ile kullanımı.	67
Şekil 5.1. Kurulan modellerin ortalama F1 skoruna göre performanslarının değerlendirilmesi.	73
Şekil 5.2. Kurulan modellerin ortalama duyarlılık performans değerine göre performanslarının değerlendirilmesi.	74
Şekil 5.3. Kurulan modellerin ortalama kesinlik performans değerine göre performanslarının değerlendirilmesi.	75

RESİMLERİN LİSTESİ

Resim	Sayfa
Resim 4.1. Karar destek sisteminin ilk erişim sayfası.....	68
Resim 4.2. Etiketi bilinen bir haberin test amacı ile karar vericiye sunulması	69
Resim 4.3. Karar destek sisteminin bir haberi	69
Resim 4.4. Karar destek sisteminin uyumsuz haber uyarı ekranı.	70
Resim 4.5.a. Karar destek sisteminin detaylı sahte haber sorgulama ekranı.	70
Resim 4.5.b. Karar destek sisteminin detaylı sahte haber sorgulama ekranı.	71



SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile aşağıda sunulmuştur.

Kısaltmalar

TF-IDF

BoW

Doc2Vec

MLP

SVM

DT

LGR

K-fold

Açıklamalar

Doküman Frekansı ve Ters Doküman Frekansı

Kelime Çantası

Document to Vector

Çok Katmanlı Yapay Sinir Ağı

Destek Vektör Makinaları

Karar Ağaçları

Lojistik Regresyon

K Katlı Çapraz Doğrulama

1. GİRİŞ

En genel anlamda enformasyon belirli ve görelî bir konuya ilişkin derlenmiş bilgi parçasıdır. Günümüz bilişim sistemleri aracılığı ile, birçok iletişim kanalından aşırı miktarda enformasyona maruz kalınmaktadır. Bu enformasyonların birçoğu bizim için hayati önem taşıyan bilgiler içerirken, birçoğu da bizlerin düşünüş şekillerine kasti olarak zarar verme amacı taşıyan bilgiler içermektedir. Bu tarz enformasyonlara Dezenformasyon denilmektedir.

Dezenformasyon doğruluğu olmayan bilginin, belirli motivasyonlar altında kasti bir şekilde yayılmasıdır. Bu motivasyonlar bir ideoloji savunmaktan, haksız kazanç elde etme ve toplum bilincini manipüle etmeye kadar uzanabilmektedir (Meel ve Vishwakarma, 2020). 2016 ABD seçimleri ve Cambridge Analytica skandalı ise dezenformasyonun yayılma araçlarından biri olan sahte haberin toplumun bilincini bir ülkenin kaderini değiştirecek kadar manipüle etme ihtimalinin olduğunu gözler önüne sermiştir. Özellikle Covid-19 gibi salgın ve kriz dönemlerinde veya seçim gibi kritik zamanlarda sahte haber çok büyük bir risk oluşturmaktadır.

Sahte haber yaymanın en temel sonucu, akılcı çözülebilecek bir problemin halka yanlış bir şekilde yansıtılarak çözülememesidir. Savaşların ülkeler arasında silahlarda değil de daha çok siber ortamlarda yürütüldüğü ya da yürütülmesinin tercih edildiği günümüzde güvenlik anlayışı da değişim geçirmektedir. Oxford Üniversitesinde 2019 yılında yayınlanan “The Global Disinformation Order” adlı yayında ülkelerin de dezenformasyon kampanyaları yürüttüğü tespit edilmiştir (Bradshaw ve Howard, 2019). Klasik Güvenlik anlayışı bakış açısında kişiler, toplumlar ve uluslar, kendileri için tartışılmaz bir önem taşıyan güvenliği oluşturabilmek için hukuki düzenlemeler yapmış, polis ve ordu teşkilatları gibi kurumlar oluşturmuş, güvenlik ihlallerine karşı silahlanma gibi önlemler almışlardır. Bilişim sistemlerinin güvenliğinin sağlanmasının vatandaşların verilerinin de güvenliğini dolayısıyla ulusal güvenliğinde sağlaması olduğu anlayışı ise teknolojik gelişmeler ile siber güvenlik kavramı aracılığıyla gerçekleşmiştir. Siber saldırılar doğrudan bir bilişim sistemini durdurmak ya da işleyemez hale getirerek veri zafiyetleri ortaya çıkarmayı amaçlarken, yine bilişim sistemleri aracılığıyla yayılan sahte haberler toplumun düşünce yapısını hedef almaktadır. Bu saldırıların ciddi maddî sonuçları olduğu kadar kurum kuruluş ya da kişiler hakkında toplum görüşlerinde düzeltilemeyecek yaralar bırakmaktadır.

Sahte haberle mücadelede en önemli araçlardan biri teyit platformları olarak karşımıza çıkmaktadır. Bu platformlarda haberler uzmanlar aracılığı ile incelenip teyit edilmektedir. Ülkemizde başlıca Teyit.org ve Doğruluk Payı teyit kuruluşları hizmet vermektedir.

(Doğruluk Payı, 2022.; Teyit.org,2022) . Dünyada ise teyit organizasyonlarının öncüleri Politifact ve FactCheck gibi kuruluşlardır (Amazeen, 2020). Dünya genelinde teyit organizasyonları çoğalırken, 2015 yılında Poynter enstitüsü bünyesinde teyit organizasyonlarını bir arada tutan, onaylayan ve belirli kurallar altında tutan bir çatı vakıf olan Uluslararası Doğruluk Kontrol Ağı (IFCN) kurulmuştur. Teyit platformları kuşkusuz sahte haberler ile yayılan dezenformasyonu önlemenin önemli araçlarından biri olsa da sosyal medyanın ürettiği enformasyon miktarı karşısında sorunlar yaşamaktadır. TalkWalker adlı sosyal medya aracı Covid 19 salgının ilk dönemlerinde 6 gün içerisinde 40 milyon haber içerikli Tweet atıldığını tespit edilmiştir (Gottlieb ve Dyer, 2020). Bu miktarda verinin insan kaynağı ile işlenmesi sürdürülebilir görünmemektedir. Dezenformasyonun doğuracağı sonuçların minimize edilmesi için bilişim sistemlerinden yararlanılması gerekmektedir. Sahte haberler bilişim sistemleri aracılığı ile denetimli, denetimsiz makine öğrenmesi yöntemleri, anomali tespiti, ağ analizleri vb. yöntemler ile tespit edilebilmektedir (Meel ve Vishwakarma, 2020).

Çalışmanın amacı Türkçe dilinde Twitter üzerinden yayınlanan siyasi haber içerikli metinlerin denetimli makine öğrenmesi ile tespitine yönelik karar destek sisteminin hazırlanmasıdır. Türkçe ve İngilizce dilleri arasındaki farklılıklar bu konuda karşılaşılan ilk problem olan Türkçe veri seti eksikliğini ortaya koymaktadır. Türkçe dilinde haberlerin sahte ve gerçek olarak tespitinin yapıldığı bir çalışma Taşkın ve arkadaşları tarafından gerçekleştirilmiştir(Taşkın vd., 2022) ancak erişilebilir ve doğrudan teyit uzmanları aracılığı ile etiketlenmiş bir Türkçe veri seti bulunmamaktadır. Çalışma kapsamında öncelikle PolitiFact teyit platformu uzmanları tarafından sahte ve gerçek haber olarak etiketlenmiş İngilizce haber içerikli Tweet metinleri Türkçeye çevrilip [Kaggle](#) ve [Github](#) Platformları üzerinden açık kaynak olarak kullanıma sunulmuştur.

Türkçe ve İngilizce dilleri arasındaki farklılıklardan ortaya çıkan bir diğer problem, makine öğrenmesi algoritmalarının kullanımında karşımıza çıkmaktadır. Metinleri makine öğrenmesi algoritmaları ile kullanmamız için gerekli olan özellik çıkarma yöntemleri ve metnin içindeki gürültülü verileri temizlememizi sağlayan ön işleme adımları dil yapısına bağlı çalışmaktadır. Çalışmada Türkçe dilinin kullanılabilmesi için gerekli metot oluşturularak sınıflandırma işlemi gerçekleştirilmiştir.

Çalışmanın ikinci bölümünde dezenformasyon, sahte haber ve makine öğrenmesi yöntemleri ile sahte haber tespiti hakkında literatür araştırması; üçüncü bölümde Türkçe haber metinlerinin sınıflandırılmasına yönelik veri setinin oluşturulması, metinlerin ön işlenmesi, metinlerden özellik çıkartma, makine öğrenmesi modellerinin oluşturulması, oluşturulan modellerin yorumlanması, dördüncü bölümde elde edilen modeller ile tasarlanan karar destek sistemi ve son bölümde sonuç ve öneriler yer almaktadır.

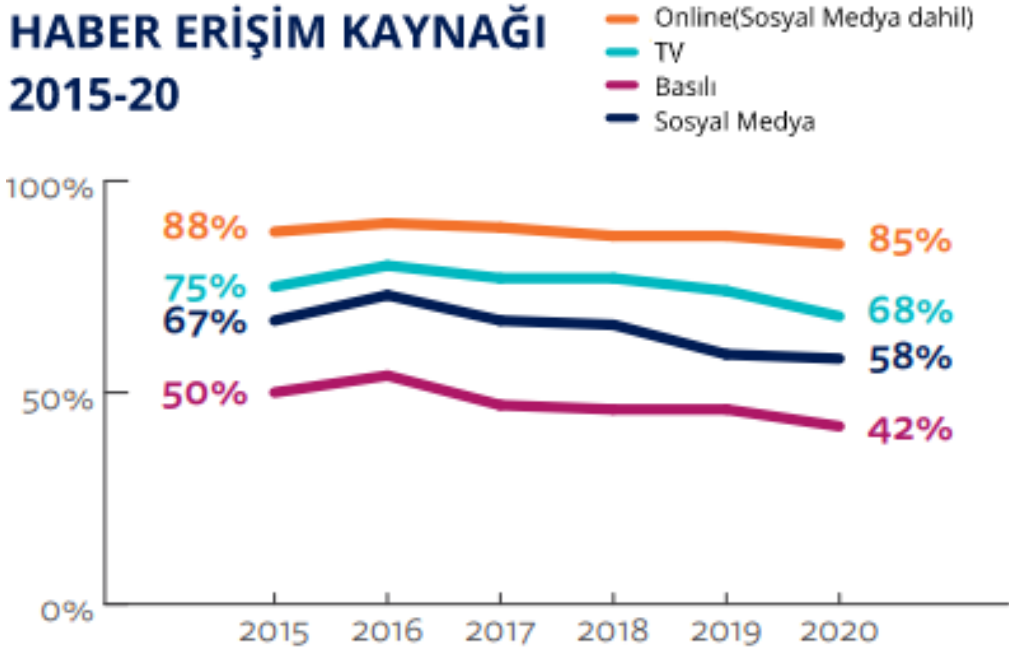
2. DEZENFORMASYON, SAHTE HABER TESPİTİ, MAKİNE ÖĞRENİMİ VE KARAR DESTEK SİSTEMİ LİTERATÜR ARAŞTIRMASI

2.1. Dezenformasyon Nedir?

Kitle iletişim insanlık tarihinde ve günümüzde hayatımızın önemli bir köşe taşı olarak görülmektedir. İnsanlar birçok kitlesel iletişim aracı ile (alfabe, matbaa, radyo, televizyon, internet vb.) çevrelerini tanımaktadır ve toplum hafızası bu kitle iletişim araçlarının taşıdığı bilgiler ile şekillenmektedir(Gökçen, 2010).

Klasik anlamdaki Medyanın (TV yayıncılığı gazete ve diğer basılı yayınlar) yapısı tek noktadan üretilen enformasyonun birden çok alıcıya gönderilmesi ile süre gelmektedir. Sosyal medya kavramı medyanın yapısını değiştirmektedir. Sosyal medya toplumdaki her bireye habercilik ve gazetecilik yapabilme imkânı tanımaktadır. Artık internet aracılığıyla tek noktadan üretilen değil her kullanıcının kendi enformasyonunu üretebildiği ve toplumun büyük bir kısmına rahatlıkla iletebildiği bir yapıya dönüşmüştür. Bu durum insanların haber alma ve haber iletme alışkanlıklarını da etkilemektedir.

Türkiye’de 2022 yılında yayınlanmış olan dezenformasyon yasası ile sosyal medya üzerinden bireylerin haber yaymaları kontrol altına alınmaya çalışılmaktadır. 2018 yılında yapılan “We Are Social 2018” adlı raporda her ülke için sosyal medya üzerinden haberlere erişim yüzdeleri incelenmiştir. Şekil 2.1.1. Türkiye’de Sosyal medyanın haberlere yüzdeleri görülmektedir. Sosyal medyanın baskısı altında kalan TV yayıncılığı, Radyo ve Gazete, Dergi ve benzeri yapılar varlıklarını, teknolojinin desteğini kullanarak her gün gelişen ve güncellenen sosyal medya mecralarına karşı halen korumaktadır ve doğru kararlar ile kendilerini güncel tutabilirler ise korumaya da devam edeceklerdir (Bulunmaz, 2011).



Şekil 2.1.1. Türkiye’de haberlere erişim kaynakları (Reuters, 2020.).

Şekil 2.1.1. incelendiğinde Türkiye özelinde haberlere erişimin, 85% çevrimiçi mecralarda gerçekleştiği görülmektedir. Bu oranda sosyal medyanın payı 58% olarak TV yayıncılığının gerisindedir. Sosyal medya 2015 yılından günümüze 9%’luk bir düşüş yaşamıştır.

2018 yılında yayınlanmış olan “We Are Social 2018” adlı raporda Türkiye için 58% sosyal medya kullanımının haber erişimi olarak platformlara yüzdelik dağılımı Şekil 2.1.2. de verilmektedir.

En Çok Tercih Edilen Sosyal Ağlar ve Mesajlaşma Servisleri

Sıralama	Platform	Haber Erişimi İçin	Toplam	Sıralama	Platform	Haber Erişimi İçin	Toplam
1	Facebook	49% (+2)	67%	4	WhatsApp	31% (-2)	67%
2	YouTube	45% (-4)	71%	5	Twitter	30% (-3)	44%
3	Instagram	41% (+8)	66%	6	Facebook Messenger	10% (-)	32%

Şekil 2.1.2. Türkiye’de tercih edilen sosyal medya mecraları (Reuters., 2018).

Şekil 2.1.2. incelendiğinde haberlere erişim Türkiye için 49%’lik pay ile Facebook, 45% pay ile YouTube ve 41% pay ile Instagram aracılığıyla sağlandığı görülmektedir.

Reuters tarafından yayınlanan “We are social 2020“raporuna göre Türkiye’de 2020 yılında günde internet kullanım süresi ortalaması 7 saat 29 dakika olarak gerçekleşmiş olup; bir önceki yılın ortalama verisinden yaklaşık olarak 14 dakika fazladır (Reuters., 2020). Bu durum her yıl daha fazla enformasyona maruz kalındığını ve topluma bilişsel olarak aşırı yüklenildiğini göstermektedir.

Örneğin sosyal medya veri toplama aracı olarak kullanılan TalkWlaker adlı yazılımın, 12 Mayıs 2020 ve 18 Mayıs 2020 tarihleri arasında çevrim içi ortamlarda Covid-19 hakkında çeşitli bilgilerin 40,2 Milyon defa bahsedildiğini ortaya çıkarmıştır (Gottlieb ve Dyer, 2020). Dolayısıyla bahsedilen 40,2 milyon haber taşıyan metnin içerisinde yararlı ve zararlı bilgiler bir arada bulunmaktadır. Veri miktarının artması ortaya çıkan enformasyonun filtrelenmesini zor bir hale getirmektedir. Enformasyon miktarı büyüdükçe gürültü artmakta; bu durum gerçekçi ve alakalı enformasyonun tespitini zorlaştırmakta ve enformasyonun kötü amaçlar uğruna büyük bir kitleyi yanlış yönlendirmek için kasıtlı olarak yayılan dezenformasyon için temel oluşturmaktadır.

Dezenformasyonun engellenememesi durumunda gerçekleşen vaka ve sonuçlarına örnek olarak, 2011 yılında “Japonya’da radyasyon kaçağı deniz suyunu ve deniz tuzunu kirletti, iyotlu tuz insanları nükleer radyasyondan koruyabilir.” şeklinde yayılan haber sonucunda Çinin Beijing şehrinde süper marketlerde tuz stokları tükenmesi ve market sahiplerinin tuz fiyatlarını 10 kat artması; 2013 yılında önemli bir haber kanalının Twitter hesabının ele geçirilmesi ile “Son dakika Beyaz sarayda iki patlama, Barack Obama yaralandı.” şeklinde yayılan haber ile 10 milyar dolar kayıp yaşanması; “2018 Ağustos ayının ikinci haftasında Chicago’da altı yüz cinayet işlendi” şeklinde yayılan haberde, gerçekte yalnızca bir cinayet olmasına rağmen halkta panik ve anksiyete yaratılması verilebilir (Meel ve Vishwakarma, 2020).

Dezenformasyonun TDK tanımı yanlış veya doğruluğu bulunmayan ve kasıtlı olarak yayılan bilgidir. Meel ve Vishwakarma göre dezenformasyon birden çok formatta yayılabilmektedir bunların en önemlileri; sahte haber, dedikodu, yanlış yönlendirme, tık tuzağı, hiciv, propaganda ve komplo teorileridir (Meel ve Vishwakarma, 2020). Dezenformasyonun yayılmasında motivasyon önemli bir yer tutmaktadır. Dezenformasyon yaymanın motivasyonları Çizelge 2.1.1. üzerinde sunulmuştur. Toplum görüşünü siyasi, finansal vb. yönlerden yanlı bir şekilde etkileme motivasyonlarına sahip dezenformasyon saldırıları gerçekleşebilmektedir.

Çizelge 2.1.1. Dezenformasyonun yayılmasının altında yatan motivasyonlar ve tanımları. (Meel ve Vishwakarma, 2020).

Motivasyon	Tanım
Politik Niyet	Sahte haberin yayılma amacı politik niyetle bir politikacı ya da insanın veya farklı bir grubun toplum gözündeki imajını zedelemek ya da hedef alınan kişi ve grupların aleyhine imajlarının değişmesi olabilmektedir.
Finansal Kazanç	Finansal kazanç motivasyonu ile büyük ölçekli yatırımlar tetiklenebilir veya hisse senetleri üzerinde olumsuz ya da haksız olumlu etkiye sebep olabilir. Yanlı ürün değerlendirmeleri yaparak ürün satışlarını arttırmak için kasıtlı olarak yayılabilirler.
Bir İdeolojiyi Destekleme Arzusu	Bir felsefe ya da bir kişi üzerine fanatik kişi/kişiler veya kuruluşların görüşlerini yayma arzusu.
Eğlence	Hiciv siteleri (Zaytung vb.) zevk ve eğlence için esprili içerikli haberler oluşturmaktadır. Oluşturulan bu haberler kötü bir niyet ile üretilmemiş olsalar da gerçek haberi çarpıtıp yanlış anlaşılmalara sebep olabilmektedir.
Müşteri Tabanını Büyütme	İnternet tabanlı habercilik çağında okuyucu kitlesi edinmek oldukça rekabet gerektirmektedir. Böylelikle yazarlar milyonlarca okunma için okuyucuyu cezbederek kendi platformlarına çekecek doğruluk kontrolü yapılmamış doğruluğu/gerçekliği sorgulanabilecek haberler paylaşabilmektedirler. Online habercilik gerçeklik ve sorumluluktan uzaklaşırken anlayışları “Önce yayınla, sonra kontrol et” haline gelmektedir.
En Son Haber Başlığına Yığılma	
Reklamcılık Geliri Üretme	Sahte haber üreticileri 2016 yılı Amerika başkanlık seçimlerinde AppNexus, Facebook Ads ve Google Adsense gibi otomatik reklam motorlarından ciddi miktarda gelir elde etmişlerdir (Reilly, 2018).
Teknolojik Nedenler	Algoritmalar doğru olanı değil popüler olan içerikleri desteklemekte ve öne çıkarmaktadır (Reilly, 2018). Daha fazla insanın ilgisini çekmek için tasarlanmış olan sahte haberlerin yapısını desteklemekte yayılımını hızlandırmaktadır.
Kamuoyunu Etkileme	Seçimler, bir firmanın kaderi ve bir ülkenin ekonomisi toplumsal kabullere bağlıdır. Toplumun görüşleri sahte dezenformasyon ile manipüle edilebilmektedir.

2.2. Sahte Haber

Sahte haber terim olarak 19. Yüzyıl Amerika'sına dayanmaktadır. 2016 yılında gerçekleşen Amerika Birleşik Devleti seçimlerde yükseliş yaşamış ve Oxford Sözlüğü tarafından 2016 yılının kelimesi "hakikat sonrası" olarak çevrilen "post truth" kavramı olarak belirlenmiştir. Kocabay ve Şener 2018 yılında yaptıkları çalışmada İngilizce literatürde "Fake News" olarak geçmekte olan terimin Türkçe literatürde karşılığının "Sahte Haber" veya "Yalan Haber" olarak yer aldığını ifade etmişlerdir(Kocabay ve Şener, 2018).

Sahte haber kavramının anlaşılması için öncelikle haber tanımının yapılması doğru olacaktır. Şener 2018 yılındaki çalışmasında haber tanımını "*Öncelikle belirtmek gerekir ki en basit tanımıyla haber, belirli bir olay hakkında kamuya bilgi vermenin daha ötesinde bir yapıya ve işleve sahiptir. Medyanın dünyada meydana gelen olaylar arasından seçerek habere dönüştürdükleri aynı zamanda topluma yansıtmak istedikleri ve toplumun düşünmesini istedikleridir.*" şeklinde ifade etmiştir (Şener, 2018). Haberler toplumun üzerinde önemli bir etkiye sahiptir, haberlerin anlatış şekilleri seçimleri ve sunuluşları toplum algısı üzerinde önemli bir değişim gücü olarak karşımıza çıkmaktadır. Bu noktada hedeflediği kitle ve kitlenin bireylerinde oluşturabileceği hasar konusunda diğer dezenformasyon yayılma formatları arasında sahte haber kendini öne çıkarmaktadır.

Sahte haber enformasyonun kasıtlı bir şekilde değiştirilerek belirli bir kurum, kişi ya da grupların aleyhine olabileceği gibi yalnızca kişisel çıkarlar doğrultusunda yayınlanabilen gerçekte meydana gelmemiş bir kurgu çalışması yapılarak sunulmasıdır. Sahte haber dezenformasyonun yayılması için kullanılan önemli bir format olmakla birlikte, ortaya çıkaracağı etki bir kurum, kuruluş veya kişi için finansal ya da politik kayıp olarak ortaya çıkmaktadır. 2.1. Dezenformasyon Nedir? bölümünde de bahsedildiği üzere sosyal medya mecraları veya çevrim içi mecraların kullanımının yaygınlaşması ile insanların haber kaynakları da televizyon ve basılı medyadan bu mecralara kayma eğilimi göstermektedir. Çevrim içi mecralarda yer alan platformlar aracılığıyla insanlar bir haberi çok az maliyetle, anlık olarak büyük bir kitle ile paylaşabilmektedirler. Reuters'in 2020 yılında yaptığı bir çalışmaya göre Türkiye de çevrim içi platformlarda bir kişi gördüğü haberi 57% oranında bir ya da birden fazla kişiyle paylaşılmaktadır. Çevrim içi platformların ve sosyal medya mecralarının kullanımı ve sağladığı olanaklar sahte haberin etkilerini ciddi noktalara taşıma potansiyeli barındırmaktadır (Reuters, 2020). Ayrıca Reuters 2018 yılında yayınladığı

çalışmasına göre Türkiye araştırmaya konu olan 37 ülke içerisinde sahte habere en fazla maruz kalan ülke olarak gösterilmiştir (Newman vd., 2018). Medya ve medyanın yayınladığı haberler toplumun görüşlerini doğrudan etkilediği düşünüldüğünde sahte haber önemli bir tehdit olarak karşımıza çıkmaktadır. Sahte haberin oluşturabileceği tehditleri sahte haberin oluşturulmasının altında yatan motivasyona bakarak anlamak mümkündür. Meel ve Vishwakarma'nın 2020 yılında yaptığı çalışmaya göre bilgi kirliliğinin altında yatan nedenler; Politik Niyet, Finansal Kazanç, Bir ideolojiyi Destekleme Arzusu, Eğlence, Müşteri Tabanını Büyütme, En Son Haber Başlığına Yığılma, Reklamcılık Geliri Üretme, Teknolojik Nedenler, Kamuoyunu Etkileme şeklinde sınıflandırılmıştır (Meel ve Vishwakarma, 2020). Sahte haberler de bir dezenformasyon yayılma formatı olduğundan ötürü aynı motivasyonlar altında üretilip yayılmaları muhtemel görülmektedir.

Çizelge 2.1.1. incelendiğinde dezenformasyonun yayılmasının altında yatan olası her motivasyonun sonuçları kişi ya da toplum için bir tehdit olarak nitelendirileceği öne sürülebilir. Sahte haber kavramının 2016 yılı Amerika başkanlık seçimi ile tekrar gündeme gelmesi dikkate alınır ise politik niyetle sahte haber üretmenin ve yaymanın daha yaygın olduğunu düşünebiliriz.

Aksoy ve Türkölmez 'in 2020 yılında gerçekleştirdikleri çalışmada 2016 yılında Amerika başkanlık seçimlerinde ve daha birçok siyasi seçimde etkisi olduğu düşünülen Cambridge Analytica skandalı, sahte haber ile yayılan dezenformasyonun ne kadar tehlikeli olabileceğini gözler önüne sermişlerdir. Şirketin Facebook üzerinden tüm kullanıcılara açık bir kişilik testi aracılığıyla kişilerin rızasıyla topladığı verinin aslında testi çözen kişinin tüm arkadaşlarının da verisinin toplandığı ortaya çıkartılmıştır. Şirket toplanan bu veriler aracılığıyla insanları siyasi görüşlerine göre gruplandırılmış ve her bölgede insanların ne düşündüğünün bir haritası çıkarılmıştır. Kişilerin verileri rızaları olmayan bir alanda işlenmiş hatta bu veriler doğrultusunda siyasal görüşlerinin değiştirilmesi amaçlı olarak tüketilmesi için içerikler sunulmuştur. Şirket iddiaları reddetmiş ve iş hayatını sonlandırmıştır (Aksoy ve Türkölmez, 2020).

Oxford Üniversitesinin 2019 yılında yayınladığı "The Global Disinformation Order" adlı raporda 70 ülkede dezenformasyon kampanyası yapıldığı ve yedi ülkenin Hindistan, Pakistan, Rusya, Çin, İran, Sudi Arabistan, Venezuela dezenformasyon kampanyası yürüttüğü belirtilmiştir (Bradshaw ve Howard, 2019). Bilgisayar ağları ve verileri güçlü

güvenlik duvarları ve sağlam politikalar ile koruma altına alınabilmektedir. Sınırlar silah gücü ile korunurken; toplumsal hafıza için silah gücü, güvenlik politikaları ve güvenlik araçları bulunmamaktadır. Toplumsal hafıza; bir topluluğu, toplum olarak nitelendirmemizi sağlayan medya tarafından şekillenmektedir (Gökçen, 2010). Günümüz sosyal medya yapısının ise dezenformasyon ile iç içe olduğunu kabul etmek zorlama bir yaklaşım olmayacaktır. Sahte haberin motivasyonları ve Türkiye'nin dezenformasyon konusundaki verileri ele alındığında hem ulusal güvenlik hem de uluslararası güvenlik açısından önem verilmesi gereken bir alan olduğu aşikardır.

2.3. Sahte Haberin Makine Öğrenmesi Algoritmaları ile Tespiti

Sahte haberle mücadele, ilgilenilmesi gerekli olan veri miktarı düşünüldüğünde oldukça güç bir alan olarak karşımıza çıkmaktadır. Sosyal medyalar aracılığıyla büyük miktarda kontrol edilmesi gereken veri ortaya çıkmaktadır. İngilizce dilinde makine öğrenme algoritmaları kullanılarak sahte haberlerin tespitinin yapılması üzerine birçok çalışma bulunmaktadır.

Ruchansky ve arkadaşları 2017 yılında gerçekleştirdikleri çalışmada, haberlerin hem içerik hem de durum tabanlı özelliklerinden faydalanarak hibrit bir makine öğrenmesi algoritması geliştirmişlerdir. Haberin metni, haberi yayan kullanıcı ve haberin ulaştığı kullanıcıları modele dahil etmişlerdir. (Ruchansky vd., 2017).

Al Asaad ve Eraşcu 2018 yılında gerçekleştirdikleri çalışmada sahte ve gerçek haber metinleri ile Scikit-Learn kütüphanesini kullanarak makine öğrenmesi modelleri geliştirilmiştir. Çalışmalarında özellik çıkartımı yöntemi olarak TF-IDF, BoW ve Bi-Gram yöntemlerini kullanmışlardır. En iyi başarımları TF-IDF yöntemi ile elde etmişlerdir (al Asaad ve Erascu, 2018).

Shabani ve Shokn 2018 yılında gerçekleştirdikleri çalışmada hiciv ve sahte haberi birbirinden ayırmak için makine öğrenmesi algoritmalarından faydalanmışlardır. Çalışmanın amacı teyit uzmanlarına yardımcı olmaktır. Özellik çıkarma yöntemi olarak TF-IDF ve paralingustik bir yöntem olan LIWC yöntemini kullanmışlardır. En başarılı sonuçlarını SVM makine öğrenmesi yöntemi ile elde etmişlerdir(Shabani ve Sokhn, 2018).

Chang ve arkadaşları 2018 yılında gerçekleştirdikleri çalışmada, e-ticaret sitelerine yapılan anormal yorumların tespitini gerçekleştirmek için özellik çıkarma yöntemi Doc2Vec ve kümeleme yöntemi kullanarak model geliştirmişlerdir (Chang vd., 2018).

Fard ve arkadaşları 2019 yılında gerçekleştirdikleri çalışmada, dezenformasyonun formatlarından biri olan söylenti tespiti için model geliştirmişlerdir. Model için hem içerik hem de durum/bağlam tabanlı 33 parametreden yararlanmışlardır (Fard vd., 2019).

Sadiq ve arkadaşları 2019 yılında gerçekleştirdikleri çalışmada hiciv, söylenti ve sahte haber tespiti için doğal dil işleme yöntemlerine alternatif bir özellik çıkartım yöntemi ortaya sunmuşlardır (Sadiq vd., 2019).

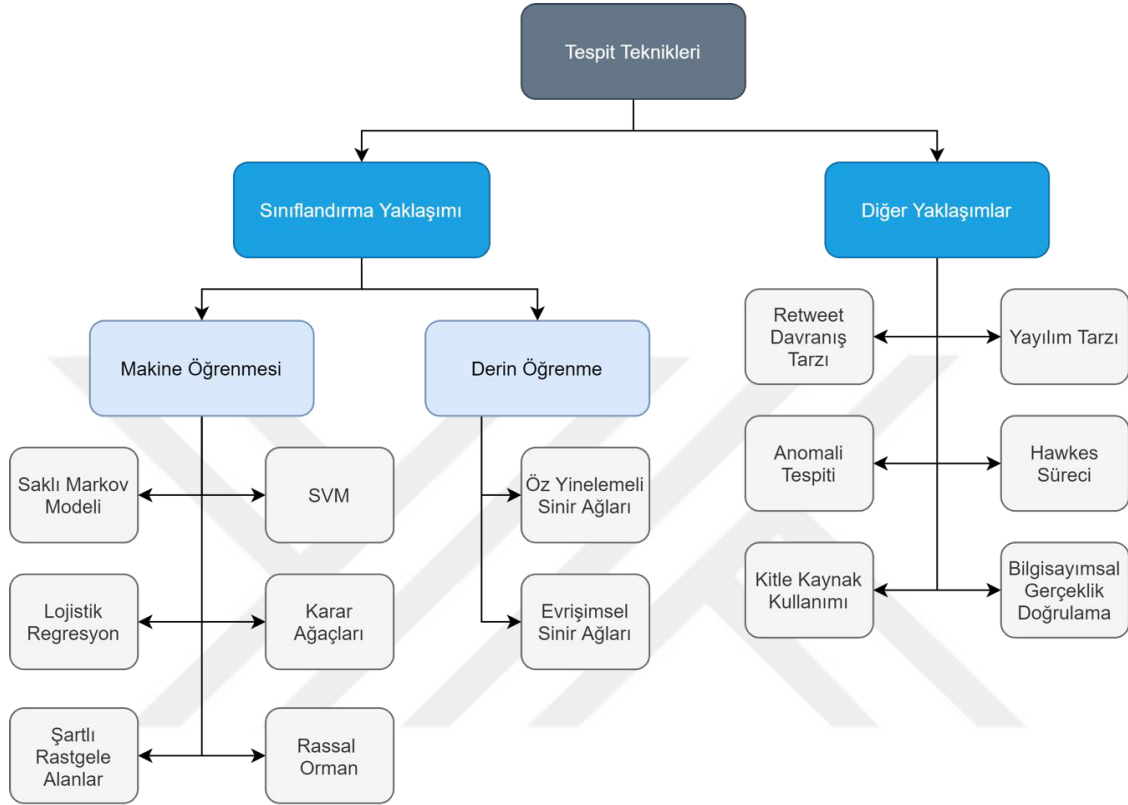
Agarwal ve arkadaşları 2019 yılında gerçekleştirdikleri çalışmalarında içerik tabanlı özellik çıkartım yöntemleri ve makine öğrenmesi yöntemleri ile sahte haber sınıflandırması gerçekleştirmişlerdir. Özellik çıkartım yöntemi olarak TF-IDF, BoW ve N-Gram yöntemleri kullanmışlardır. Sonuçlarını karmaşıklık matrisi ile yorumlamışlardır. SVM makine öğrenmesi yöntemi ile en başarılı sonuçlarını elde etmişlerdir (Agarwal vd., 2019).

Isaakidou vd. 2021 yılında gerçekleştirdikleri çalışmada Covid-19 sürecinde yayılan sahte haberlerin engellenmesi için karar ağaçları ve TF-IDF özellik çıkarma yöntemleri ile tespitini gerçekleştirmişlerdir (Isaakidou vd., 2021).

Taşkın ve arkadaşları 2022 yılında gerçekleştirdikleri çalışmada Türkçe dilindeki haberleri sahte ve gerçek olarak makine öğrenmesi algoritmaları aracılığı ile sınıflandırmışlardır. Çalışmada Twitter scrapper api aracılığı ile 3 konu üzerinden 18000 mesaj toplanmış; metinden özellik çıkarma yöntemi olarak TF-IDF ve Doc2Vec yöntemleri kullanılmıştır. Çalışmada hem denetimli hem de denetimsiz makine öğrenmesi yöntemleri kullanılarak sonuçlar elde edilmiştir (Taşkın vd., 2022).

Sahte haberleri insanlar aracılığı ile kitle kaynak kullanımı veya bilgisayar algoritmaları aracılığıyla doğal dil işleme ve veri madenciliği yöntemleri kullanarak tespit etmek mümkündür. Bondielli ve Marcelloni, 2019 çalışmalarında sahte haber tespit yaklaşımlarını sınıflandırma ve diğer yaklaşımlar olarak Şekil 2.3.1. üzerinde sunulduğu şekilde ikiye ayırmışlardır. Sınıflandırma yaklaşımları veri madenciliği, doğal dil işleme ve makine

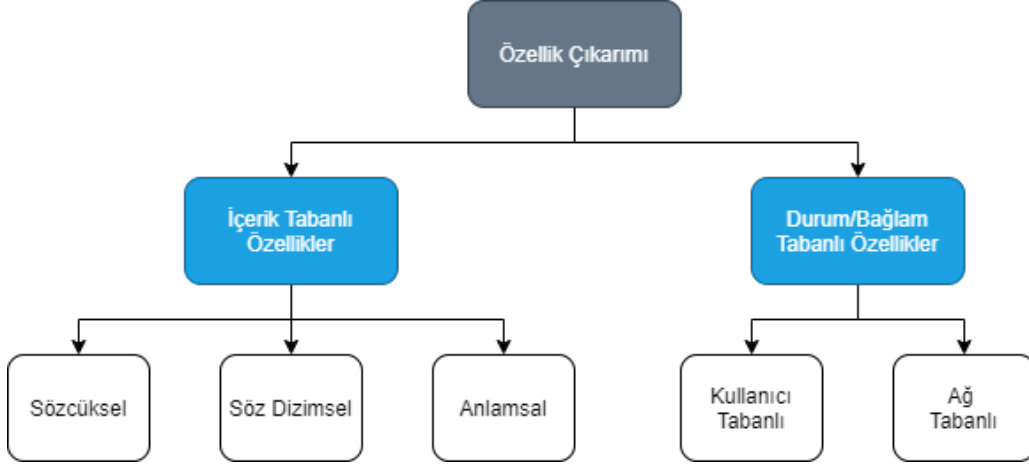
öğrenmesi gibi yapay zekâ algoritmaları yardımı ile sahte haber tespitinin sağlandığı yaklaşımları içerisinde barındırırken diğer yaklaşımlar daha çok teyit platformlarının da dâhil olduğu kitle kaynak kullanımı, aykırılık tespitleri ve yayılış tarzları ile sahte haber tespitini sağlayan yaklaşımlardır.



Şekil 2.3.1. Sahte haber tespit teknikleri (Bondielli ve Marcelloni, 2019)

Sahte haber tespit yöntemlerinin kullanılabilmesi için kontrol edilmek istenen haberden belirli özelliklerin çıkartılması gerekmektedir. Özellikle Şekil 2.3.1. de yer alan sınıflandırma yaklaşımı altında yer alan makine öğrenimi ve derin öğrenme yöntemleri, sinir ağları, girdi olarak öznitelik vektörleri gerektirir (Ye vd., 2020). Haberlerin bilgisayar aracılığıyla tespitinde metinsel ifadelerin sayısal özelliklere çevrilerek bahsedilen algoritmalara öznitelik vektörü olarak sunulması gerekmektedir.

Bondielli ve Marcelloni 2019 yılında yapmış oldukları çalışmada özellik çıkarımı yöntemlerini Şekil 2.2.2. üzerinde verildiği gibi sınıflandırılmıştır.



Şekil 2.2.2. Sınıflandırma yaklaşımı için özellik çıkarım yöntemleri (Bondielli ve Marcelloni, 2019)

Şekil 2.2.2. de görülen içerik tabanlı özellik çıkarım yöntemleri haberlerin sınıflandırılması için doğrudan haber metinlerine odaklanan yöntemlerdir. En çok kullanılan içerik tabanlı özellik çıkarım yöntemleri Kelime Çantası (BoW), Doküman Frekansı ve Ters Doküman Frekansı (TF-IDF), LDA, N-Gram ve Doc2Vec'dir.

Şekil 2.2.2. görülen durum/bağlam tabanlı özellik çıkartım yöntemleri, gerçek haberleri veya sahte haberleri çevreleyen verileri kullanan yöntemlerdir. Özellikle, en çok kullanılan bağlam özellikleri, kullanıcıların analizi, söylentilerin veya haberlerin kaynakları, sosyal medyadaki bilgilerin yayılma yapıları ve diğer kullanıcıların haberlere ilişkin tepkileri ile ilgilidir.

Çalışmada içerik tabanlı özellik çıkarma algoritmalarından BoW, TF-IDF, Doc2Vec algoritmaları kullanılmıştır.

2.3.1. Özellik çıkarma algoritmaları

2.3.1.1. Kelime çantası algoritması (BoW)

1994 yılında Harris tarafından keşfedilen Kelime Çantası (Bag of Words) algoritması başarılı sonuçlar veren, görece uygulaması ve anlaşılması kolay bir algoritma olması sebebi ile doğal dil işleme alanında oldukça yaygın olarak kullanılmaktadır (Harris, 1954). Kelime Çantası (BoW) modeli, metni temsil etmek için sabit uzunlukta bir vektör kullanır. Bu vektör, kelime çantası olarak ifade edilir. Kelime çantasında bulunan her bir kelimenin metin içerisindeki frekansının hesaplanmasıyla metin vektörü elde edilir. Model kelime

dizisi bilgisini dikkate almasa da uzun vadeli kelime-sözcük bağıntılarını ve konu bilgilerini yakalamada başarılı bir modeldir. Belge sınıflandırma ve bilgi alma gibi doğal dil işleme görevlerinde kullanılmaktadır (Hofman, 1999; Wei ve Croft, 2006). Şekil 3.3.1. Üzerinde gösterilen metinden özellik çıkarımı işlemi BoW algoritması ile gerçekleştirilmiştir.

2.3.1.2. TF-IDF algoritması

Terim Frekansı ve Ters Doküman Frekansı (Term Frequency – Inverse Document Frequency -TF-IDF) algoritma istatistik tabanlı bir öznitelik vektörü ağırlıklandırma yöntemidir (Ali vd., 2018; Ramos, 2003). TF-IDF bir kelimenin bir veri seti içerisinde bulunan bir belge için ne kadar önem ifade ettiğini hesaplar. Örneğin; birçok siyasi haber içeren veri seti içerisinde seçim kelimesinin öncelikle metin içerisindeki tekrarına daha sonra bütün haberler içerisindeki tekrar sayısına bakarak seçim kelimesinin ilgili haber için ne kadar ayırt edici olduğu TF-IDF algoritması ile belirlenebilir. Eğer seçim kelimesi ilgili haberde sık bir şekilde kullanılmış ve veri setinin tamamında çok seyrek kullanılmış ise, seçim kelimesi ilgili haber için belirleyiciliği yüksek bir kelime olarak karşımıza çıkacaktır. Özellikle metinler üzerinde arama işlemlerinde kullanılan bir yöntemdir.

$$TF - IDF(t, d) = 1 + \log(tf) * \log_{10}\left(\frac{N}{df_t}\right) \quad (1)$$

Eşitlik 1 üzerinde t ağırlıklandırmak istenilen kelime, d veri setini, tf değeri ise ağırlıklandırılmak istenilen kelimenin belge içerisindeki tekrar sayısının doküman da bulunan kelime sayısına bölümü ile elde edilir. N değeri veri setinde bulunan kelime sayısını ifade ederken df_t değeri ise ilgili kelimenin bütün veri setindeki tekrar sayısı olarak karşımıza çıkmaktadır.

2.3.1.3. Doc2Vec algoritması

Google mühendisleri tarafından geliştirilen Word2Vec algoritmasının bir türevidir. Sözcük ilişkileri yerine cümleler arasındaki ilişkileri inceler. Bir cümleyi sabit boyutlu bir vektöre dönüştürebildiği için makine öğrenimi için oldukça verimli bir yöntemdir. Diğer öznitelik çıkarma yöntemlerinden farklı olarak, doc2vec kelimelerin sırasını, birlikte kullanım sıklıklarını ve kelimeler arasındaki ilişkileri hesaplar (Chang vd., 2018; Chen ve Sokolova, 2018). Veri seti büyüdükçe Doc2Vec performansı da aynı oranda artmaktadır.

2.3.2. Makine öğrenmesi algoritmaları

Makine öğrenimi, bilgisayarların doğrudan programlanmadan "öğrenmesini" sağlamayı amaçlayan bir bilgisayar bilimi dalıdır. Makine öğrenimi, verilerden öğrenebilen, deneyimlerden bilgi edinebilen ve zaman içinde öğrenme davranışlarını iyileştirebilen algoritmaların tasarlanması ve geliştirilmesi için zekayı anlama ile ilgilidir. Makine Öğrenmesinin amacı insan tarafından erişilemeyen, verilerdeki yüksek boyutlu boşluklarda gizlenen ilgili yapısal ve/veya zamansal kalıpları ("bilgi") keşfetmektir (Holzinger, 2017).

Sahte haber tespitinde kullanılan başlıca makine öğrenmesi algoritmaları: Naive Bayes, Lojistik Regresyon, Karar Ağaçları, Destek Vektör Makinaları, Çok Katmanlı Yapay Sinir Ağı, Random Forest'dir (Albahr ve Albahr, 2020; Mahmud vd., 2021; Poddar vd., 2019).

Çalışmada denetimli makine öğrenmesi algoritmalarından MLP, DT, SVM, LGR algoritmaları kullanılmıştır.

2.3.2.1. Çok katmanlı yapay sinir ağı (MLP)

Çok katmanlı algılayıcı, bir yapay sinir ağı modelidir. Doğrusal ve doğrusal olmayan problemleri çözmek için kullanılır. Bir MLP, girdi katmanı, gizli katman ve çıktı katmanlarından oluşur. MLP, tahmin için etiketlenmiş verilerle hatayı geriye yayarak öğrenmeyi gerçekleştirir. Görünmeyen gözlemleri tahmin edebilen güçlü bir araçtır (Caminhas vd., 2003).

2.3.2.2. Karar ağaçları (DT)

Karar ağaçları, ayırt etme ve kestirimci modelleme için önemli olan özellikleri keşfeder ve büyük veritabanlarındaki modelleri çıkarır. Bir karar ağacı kök düğüm, yaprak düğüm ve iç düğümden oluşur. Gelen verilere göre kök düğüm ve iç düğümler öznitelikleri temsil eder ve örnek uzayı iki veya daha fazla parçaya böler. Her yaprak düğüm bir sınıfı temsil eder (Quinlan, 1996).

2.3.2.3. Destek vektör makinaları (SVM)

Destek Vektör Makinesi, genellikle sınıflandırma problemlerinde uygulanan denetimli öğrenme yöntemlerinden biridir. Bir düzlemde noktaları ayırmak için bir çizgi çizer. Her iki

sınıfın puanları için çizgiler arasındaki maksimum mesafeyi hedefler. SVM, küçük veya orta ölçekli veri kümeleri için uygundur ve normalleştirmeye duyarlıdır (Cortes, 1995; Tong ve Koller, 2001).

2.3.2.4. Lojistik regresyon (LGR)

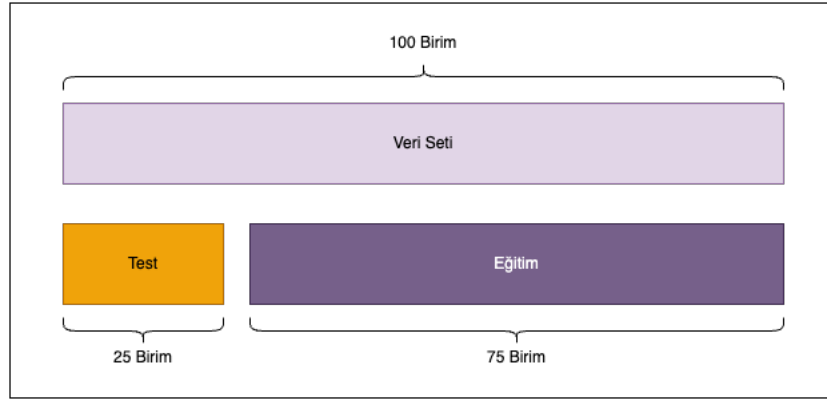
Lojistik regresyon, bir veri setini bir veya daha fazla bağımsız değişkenle analiz eden istatistiksel bir yöntemdir. Sonuç ikili bir değişkenle ölçülür (yalnızca iki olası sonuç vardır). Lojistik regresyonun amacı, iki yönlü bir özellik ile bir dizi ilgili bağımsız değişken arasındaki ilişkiyi açıklamak için en uygun modeli bulmaktır (Indra vd., 2017; Pranckevicius ve Marcinkevicius, 2017; Tolles ve Meurer, 2016).

2.3.3. Makine öğrenmesi modellerin doğrulanması

Belirli bir amaç için eğitilen makine öğrenmesi algoritmalarının sonuçlarının daha iyi anlaşılması ve yorumlanması için uygun doğrulama yöntemi ile test edilmiş olması gerekmektedir. Makine öğrenmesi algoritmalarının eğitimi ve test edilmesinde kullanılan verilerin seçiminde yanlılık oluşması veya yanlış doğrulama yönteminin seçilmesi durumunda tarafsız bir performans yorumlaması gerçekleştirilemez (Horvat vd., 2020). Literatürde iki doğrulama yöntemi sıklıkla kullanılmaktadır. Bu doğrulama yöntemleri Eğitim/Test ayrımı ve K-fold geçerlilik yöntemleridir.

2.3.3.1 Eğitim/Test ayrımı

En temel sınıflandırıcı makine öğrenme algoritması doğrulama yöntemidir. Şekil 2.3.3.1.1. üzerinde de görüleceği üzere verilerin belirli bir kısmının eğitim, belirli bir kısmının ise test verisi olarak ayrılmasına dayanır. Bu yaklaşımın en temel yararı, modelin daha önce görmediği verilere nasıl tepki verdiğini görebilmemizdir.



Şekil 2.3.3.1.1. Veri setinin Test/Eğitim ayırımı yöntemine göre ayırımı

Eğitim ve test verileri veri setinden rastgele seçilir. Ancak bu rastgele seçim örnekleme yanlılığına sebep olabilir. Örnekleme yanlılığı (Sampling Bias): veri setinin içerisinde yer alan sınıfların eğitim ve test verisine homojen bir şekilde dağılmadığı durumdur. Başka bir deyişle örnekleme yanlılığı, bir popülasyonun rastgele olmayan bir örneğinden kaynaklanan sistematik hatadır ve popülasyonun bazı üyelerinin dahil edilme olasılığının diğerlerinden daha az olmasına neden olarak taraflı bir örnekleme sonuçlanır.

2.3.3.2 K katlı çapraz geçerlilik

K katlı çapraz geçerlilik doğrulama (K Fold Cross Validation/K-fold) yöntemi örnekleme yanlılığını en aza indirmek için kullanılan bir yöntemdir. Yöntem örnekleme yanlılığını en aza indirmek için öncelikle verileri k boyutunda parçalara ayırır. Bu parçalardan birini test için ayırır ve geriye kalan k -1 parçayı eğitim için kullanır. Avantajı, tüm gözlemlerin hem eğitim hem de doğrulama için kullanılması ve her gözlemin (k parçacığının) doğrulama için bir kez kullanılmasıdır.

Yöntemin veri setini k adet parçaya ayırması ve her bir parçanın eğitim ve test verisi olarak kullanımı Şekil 2.3.3.2.1. üzerinde gösterilmiştir. Her bir kombinasyonun sonucunda elde edilen başarımların ortalaması alınarak makine öğrenmesi algoritmasının sınıflandırma performansı yorumlanabilir.



Şekil 2.3.3.2.1. Veri setinin K-fold geçerlilik yöntemi için eğitim ve test verisi olarak ayrımı.

K-fold geçerlilik modelinde k değeri literatürde yaygın olarak işlem karmaşıklığı ve doğrulamanın ideal oranda gerçekleştiği düşünülen 5 veya 10 olarak seçilmektedir.

2.3.4. F1, duyarlılık ve kesinlik performans değerleri

Modellerin performansları test veri setine hangi oranda doğru cevap verdikleri ile ölçülebilir. Bu değere doğruluk değeri (Accuracy) denir. Doğruluk değeri temel olarak modelin performansı hakkında önemli bilgiler verse de çok önemli yanıtlara sebebiyet verebilir. Örneğin dengesiz veri setleri ile çalışıldığında test verilerininin büyük bir bölümü tek bir sınıftan oluşabilmektedir. Bu durum veri setinin eğitim ve test verisi olarak ayrıldığında test verisinin çok büyük bir bölümünün tek sınıftan oluşmasına sebebiyet vermektedir. Oluşturulan model de yanlış bir şekilde bütün verileri ağırlıklı sınıfta olduğunu tahmin ettiğinde doğruluk oranı çok yüksek olarak karşımıza çıkacaktır. Ancak gerçekte herhangi bir öğrenme işlemi gerçekleşmemiş olacaktır (He vd., 2012).

Bir ikili sınıflandırıcı modelde, iki tip hataya karşılaşılmaktadır. Bu hatalar Tip1 ve Tip 2dir. Tip 1 hata gerçek olan bir haberi sahte olarak etiketlemektir. Tip 2 hata ise gerçekte sahte olan bir haberi gerçek olarak etiketlemektir (He vd., 2012; Zhao vd., 2015). Sahte bir haberin gerçek olarak etiketlenmesi, dezenformasyonun daha da fazla yayılmasına sebebiyet verebilecekken gerçek bir haberin sahte haber olarak sınıflandırılması ise doğru bir haberin ulaşması gereken kitleye ulaşamamasına yani medyanın baskılanmasına sebebiyet

verecektir. Toplumun bilgi alma kanallarından olan haber söz konusu olduğunda bahsi geçen iki tip hatanın da minimum seviyede tutulması gerekmektedir.

Bir sınıflandırıcının performansının doğru bir şekilde değerlendirilmesi için yalnızca doğruluk değil, duyarlılık, keskinlik ve F1 skorlarının da incelenmesi gerekmektedir. Bu değerlerin hesaplanabilmesi için öncelikle Çizelge 2.3.4.1. ile gösterilen karmaşıklık matrisinin (confusion matrix) oluşturulması gerekmektedir (G. Zhu vd., 2021). Bir karmaşıklık matrisi sınıflandırıcının bir haberi doğru ya da yanlış olarak etiketlemesi ile haberin gerçekte etiketlenmiş olduğu değerlerin kıyaslanması ile elde edilir.

Çizelge 2.3.4.1. Karmaşıklık Matrisi

		Gerçek Değerler	
		Pozitif	Negatif
Tahmin Edilen Değerler	Pozitif	Doğru Pozitif (TP)	Yanlış Pozitif (FP)
	Negatif	Yanlış Negatif (FN)	Doğru Negatif (TN)

Doğru Pozitif (True Positive): Gerçek değeri pozitif olan bir verinin sınıflandırıcı tarafından da pozitif olarak tahmin edilmesi.

Yanlış Pozitif (False Positive): Gerçek değeri negatif olan bir verinin sınıflandırıcı tarafından pozitif olarak tahmin edilmesi. Tip 2 hata örneğidir.

Yanlış Negatif (False Negative): Gerçek değeri pozitif olan bir verinin sınıflandırıcı tarafından negatif olarak tahmin edilmesi. Tip 1 hata örneğidir.

Doğru Negatif (True Negative): Gerçekte değeri negatif olan bir verinin sınıflandırıcı tarafından da negatif olarak tahmin edilmesi.

2.3.4.1. Doğruluk değeri (Accuracy)

Modelin doğru tahminlerinin test verisi içerisindeki bütün tahminlere oranını ifade etmektedir. Doğruluk değeri Modelin yanlışlarına değil, sadece doğrularının veri setine oranını göz etmektedir. Test skoru olarak da ele alınmaktadır.

$$Doğruluk = \frac{TP+TN}{TP+FP+TN+FN} \quad (2)$$

2.3.4.2. Duyarlılık değeri (Recall)

Duyarlılık değeri pozitif olarak sınıflandırılması gereken verinin ne kadarının doğru bir şekilde sınıflandırılabilirdiğini kontrol eden bir değerdir. Modelin negatif bir veriyi tespit edebilme kapasitesi olarak da ifade edilebilir. Sahte bir haberin model tarafından gerçek haber olarak sınıflandırılması dezenformasyonun etkilerini daha fazla arttıracaktır. Değer ne kadar büyük olursa model o denli duyarlı olacaktır.

$$Duyarlılık = \frac{TP}{TP+FN} \quad (3)$$

2.3.4.3. Kesinlik değeri (Precision)

Kesinlik değeri pozitif verilerin, negatif olarak etiketlenmesini denetler. Diğer bir deyiş ile doğru verilerin ne kadar doğru sınıflandırıldığıdır. Örneğin aslen gerçek olan bir haberin model tarafından sahte haber olarak etiketlenip, doğru bilginin alıcısına ulaşmaması ihtimalini ifade eder.

$$Kesinlik = \frac{TP}{TP+FP} \quad (4)$$

2.3.4.4. F1 Değeri

F1 Değeri diğer bir ifade ile F1 skoru duyarlılık ve kesinlik değerlerinin harmonik ortalamasıdır. Tüm hata maliyetlerini içerir ve uç durumları göz ardı eder. Bu açıdan bakıldığında doğruluk değeri sadece doğru tahminlerin, tüm tahminlere oranına bakarken, F1 değeri tüm hata maliyetlerini içermektedir.

$$F1 = 2 * \frac{Duyarluluk * Kesinlik}{Duyarluluk + Kesinlik} \quad (5)$$

2.4. Karar Destek Sistemi

Karar destek sistemi (KDS), karar verme aşamasına karar vericiye kararı hakkında yardımcı olabilmek için interaktif bir kullanıcı ara yüzü ile geliştirilmiş olan yazılım tabanlı sistemdir. Karar destek sistemleri, geçmişi Simon ve arkadaşlarının 1950 ve 1960 yıllarında gerçekleştirdiği çalışmalara dayanan, üzerinde oldukça çalışılmış bir alandır. Simon'ın öne sürdüğü karar destek modelinin zekâ, tasarım ve karar olmak üzere üç ana unsuru bulunmaktadır (Hosack vd., 2012).

Spargue'ye göre bir karar destek sistemini tanımlamak için gerekli unsurlar; karar probleminin az yapılandırılmış olması, analitik teknikler ile veriyi birleştirmesi, kullanıcı dostu olması ve yeni problemlere uyarlanabilir olmalıdır (Sprague, 1980).

Karar destek sistemlerinin operasyon, finansal yönetim, stratejik karar verme, optimizasyon ve simülasyon gibi birçok alanda kullanıldığı görülmekle birlikte çalışma kapsamında sadece sahte haber tespitine yönelik karar destek sistemleri değerlendirilmiştir.

Janze ve Risius 2017 yılında Twitter üzerinde yayılan sahte haberleri otomatik olarak tespit etmek üzerine bir karar destek çalışması gerçekleştirmişlerdir. Kurdukları model doğrudan sınıflandırma yapmayı BuzFeed'in bir haberin ne kadar sahte olduğunu ölçülmesi için geliştirdiği sınıflara yakınlık üzerinden yapmıştır. Özellik vektörü olarak sosyal bağlam verilerden yararlanmışlardır (Janze ve Risius, 2017).

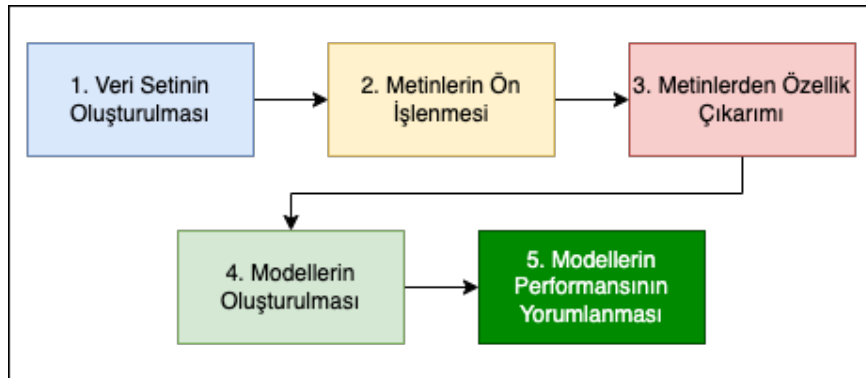
Okoro vd. 2018 yılında gerçekleştirdikleri çalışmada sahte haberlerin tespitinde teyit uzmanları ve makine öğrenme yöntemleri sentezleyerek hibrit bir model geliştirmişlerdir. Önerilen hibrit sistem bir haberi doğrudan sahte ya da gerçek haber olarak etiketlemeyip sahte ya da gerçek haber olabilme yakınlığını vermektedir (Okoro vd., 2018).

3. TÜRKÇE HABER METİNLERİNİN SINIFLANDIRILMASI

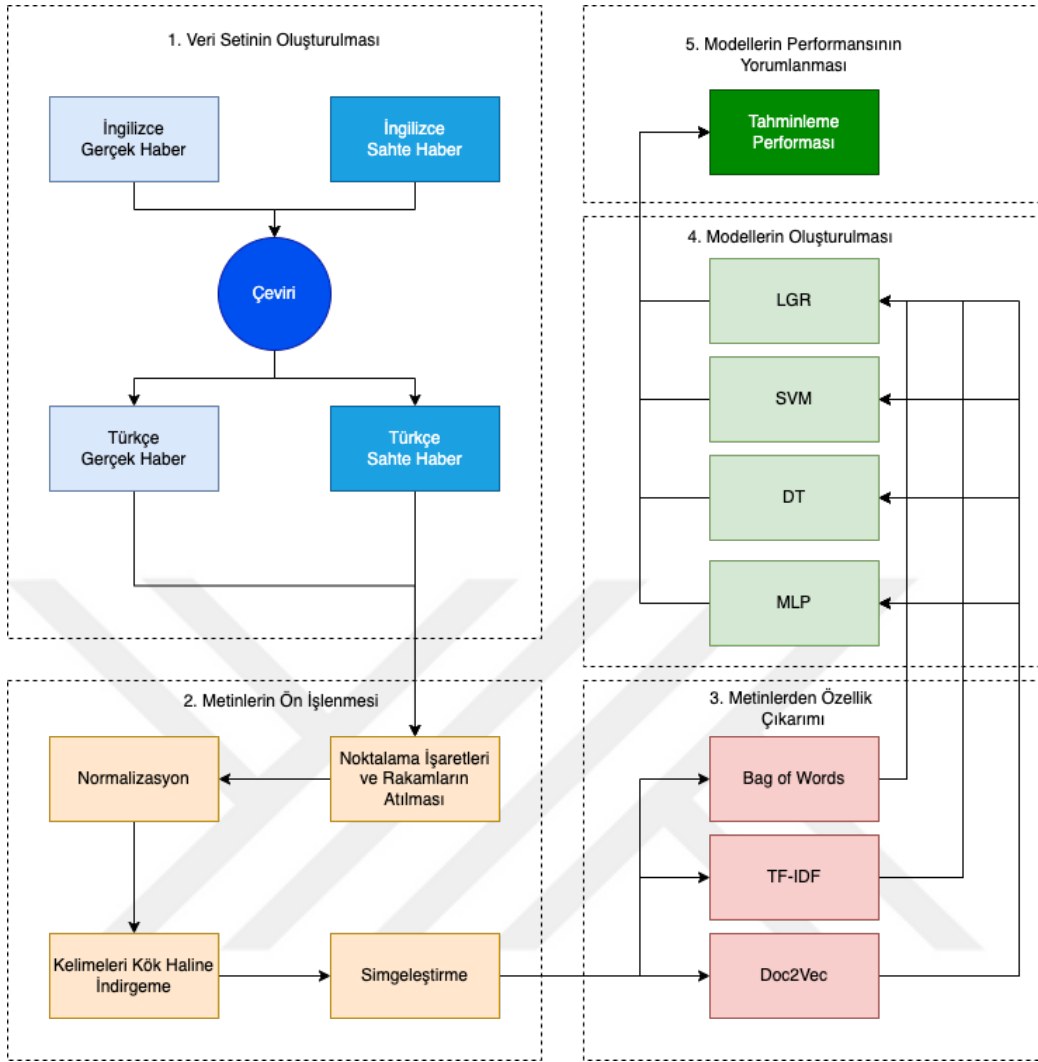
Haber metinlerinin sınıflandırılmasında makine öğrenimi algoritmalarının kullanılması literatürde sıklıkla çalışılan bir konu olup, sınıflandırma aşamasında İngilizce veri setlerinin kullanıldığı görülmektedir. Ancak Türkçe sahte haber tespitinde Türkçe ve İngilizce dilleri arasındaki farklılıklar nedeni ile metin ön işleme adımlarında ve özellik çıkarma yöntemlerinde sorunlar meydana gelmektedir. Bu sorunların temel nedeni diller arasındaki karakter farklılıkları, gramer yapısı, kelimelerin ek alış şekillerinin ayrı olmasından kaynaklanmaktadır.

Çalışmada, doğrudan teyit uzmanları aracılığı ile sahte ve gerçek olarak etiketlenmiş Türkçe haber veri seti oluşturulmuş; bu haberler BoW, TF-IDF ve Doc2Vec özellik çıkarma algoritmaları ile öznitelik vektörlerine dönüştürülmüş; MLP, DT, SVM ve LGR makine öğrenme algoritmaları bu öznitelik vektörleri ile eğitilerek, ortaya çıkan 12 farklı modelin performans verileri ortaya konulmuştur.

Türkçe haber metinlerinin sınıflandırılmasında içerik tabanlı özellik çıkarma yöntemleri tercih edilmiş olup çalışma metodunun genel adımları Şekil 3.1.'de; her bir adımın içerisinde barındırdığı detaylar ise Şekil 3.2. üzerinde gösterilmiştir.



Şekil 3.1. Çalışma metodunun adımları.



Şekil 3.2. Çalışma metodunun adımlarının detaylı gösterimi.

3.1. Türkçe Sahte Haber Veri Setinin Oluşturulması

İngilizce dilinde makine öğrenmesi ve doğal dil işleme aracılığıyla tespit edilmesi yoğun bir şekilde çalışılmış bir konu olup birçok veri seti bulunmaktadır. Türkçede sahte haber tespitinin makine öğrenmesi aracılığı ile yapıldığı yalnızca bir çalışma bulunmakta ve bu çalışmada da haber verileri Twitter üzerinden WebScrapper API kullanılarak elde edilmiş haberler geliştirilen ara yüz ile teyit.org platformundan yararlanılarak yazarlar tarafından etiketlenmiştir (Taşkın vd., 2022). Dolayısıyla doğrudan teyit uzmanları aracılığıyla etiketlenmiş Türkçe dilinde gerçek ve sahte haber içeren açık kaynak erişimli bir veri seti bulunmamaktadır. Bu nedenle öncelikle İngilizce açık kaynak olarak erişilebilen, güvenilir teyit uzmanları aracılığı ile etiketlenmiş dengeli siyasi haberler içeren bir veri seti olan FakeNewsNet Türkçeye çevrilmiş, [Kaggle](#) ve [Github](#) platformlarından açık kaynak kullanımına sunulmuştur.

Bondielli ve Marcelloni 2019 yılında yaptıkları çalışmada, Bir haberin sahte veya gerçek olarak sınıflandırılması için haber içeriğine, sosyal bağlama veya mekân zamansal verilere dayalı sınıflandırıcı makine öğrenmesi algoritmaları kullanılacağını ifade etmişlerdir (Bondielli ve Marcelloni, 2019).

Haberlerin sahte ve gerçek olarak sınıflandırılmasında haber metni ve haberin paylaşıldığı görsel ciddi bir önem taşımaktadır. Shu vd. 2018 yılında gerçekleştirdikleri çalışmada Çizelge 3.1.1.de verilen 7 veri setinin dilsel ve görsel veriler açısından karşılaştırılması sunulmuştur.

Çizelge 3.1.1. Mevcut veri setlerinin haber içeriğine göre kıyaslanması (Shu vd., 2018).

Veri seti	Haber İçeriği	
	Dilsel	Görsel
BuzzFeedNews	✓	
LIAR	✓	
BS Detector	✓	
CREDBANK	✓	
BuzzFace	✓	
FacebookHoax	✓	
FakeNewsNet	✓	✓

Çizelge 3.1.1. incelendiğinde haber içeriğinin dilsel ve görsel veriler açısından iki niteliği de karşılayan veri setinin FakeNewsNet olduğu görülmektedir.

Sahte haberler içerdiği metin ve görsel kadar, yayıldığı kaynak ve haberi yayan kitlenin sınıflandırılması ile de tespit edilebilmektedir (Meel ve Vishwakarma, 2020). 7 veri seti haberi yayan kullanıcı, gönderileri ve bu gönderilere aldığı yanıtlar ve ağ içindeki dağılımı Çizelge 3.1.2. üzerinde karşılaştırılmıştır.

Çizelge 3.1.2. Mevcut veri setlerinin sosyal bağlama göre kıyaslanması (Shu vd., 2018).

Veri seti	Sosyal Bağlam			
	Kullanıcı	Gönderi	Yanıt	Network
BuzzFeedNews				
LIAR				
BS Detector				
CREDBANK	✓	✓		
BuzzFace		✓	✓	
FacebookHoax	✓	✓	✓	
FakeNewsNet	✓	✓	✓	✓

Çizelge 3.1.2. incelendiğinde kullanıcı, gönderi, yanıt ve network sosyal bağlamı açısından FakeNewsNet veri setinin hepsini karşıladığı görülmektedir.

Sahte haberin tespitinde kullanılan bir diğer yöntem ise haberlerin yayıldığı zaman aralığı ve mekanların sınıflandırılmasıdır (Bondielli ve Marcelloni, 2019). Çizelge 3.1.3. üzerinde 7 adet veri seti mekânsal ve zamansal veriler içermeleri bakımından karşılaştırılmıştır.

Çizelge 3.1.3. Mevcut veri setlerinin mekân zamansal olarak kıyaslanması (Shu vd., 2018).

Veri seti	Mekân Zamansal	
	Mekânsal	Zamansal
BuzzFeedNews		
LIAR		
BS Detector		
CREDBANK	✓	✓
BuzzFace		✓
FacebookHoax		
FakeNewsNet	✓	✓

Çizelge 3.1.3. incelendiğinde mekânsal ve zamansal açıdan incelendiğinde FakeNews veri setinin iki niteliği karşıladığı görülmektedir.

Sahte haber tespitinde kullanılacak veri setinin seçiminde çok önemli bir kriter daha bulunmaktadır. Bu kriter haberlerin sahte ve gerçek olarak etiketlenmesinde hangi yöntemin kullanıldığıdır. İnsanlar aracılığı ile sahte haber tespiti yapılırken, sahte haber teyidi yapan kişinin bu alandaki uzmanlığı ve tarafsızlığı çok büyük önem taşımaktadır. FakeNewsNet veri seti Uluslararası Teyit Ağı (International Fact-Checking Network) tarafından onaylanmış PolitiFact adlı teyit platformu tarafından etiketlenmiş 624 gerçek 432 sahte haberlerden oluşturulmuştur (Shu vd., 2018).

Çizelge 3.1.4. üzerinde çevirisi gerçekleştirilen FakeNewsNet veri setinin içerdiği sahte ve gerçek haberlerinin dilsel, görsel, kullanıcı, gönderi, yanıt, network, mekânsal ve zamansal kategorilerindeki veri sayıları sunulmuştur.

Çizelge 3.1.4 FakeNewsNet Veri setinin haber içeriği sosyal bağlam ve mekân zamansal veri sayıları (Shu vd., 2018).

	Kategori	Özellik	Sahte Haber	Gerçek Haber
Haber İçeriği	Dilsel	Haber Başlığı	432	624
	Görsel	Haber Görseli	336	447
Sosyal Bağlam	Kullanıcı	Tweet'i Paylaşan	95553	249887
	Gönderi	Haber İçeren Tweet Sayısı	164.892	399237
	Yanıt	Retweet Sayısı	23489	67035
	Network	Takipçi Sayısı	1299.98	982.67
Mekân Zamansal	Mekânsal	Tweetin Gönderildiği Bölge	217379	719331
	Zamansal	Tweetin Gönderim Zamanı	171301	669641

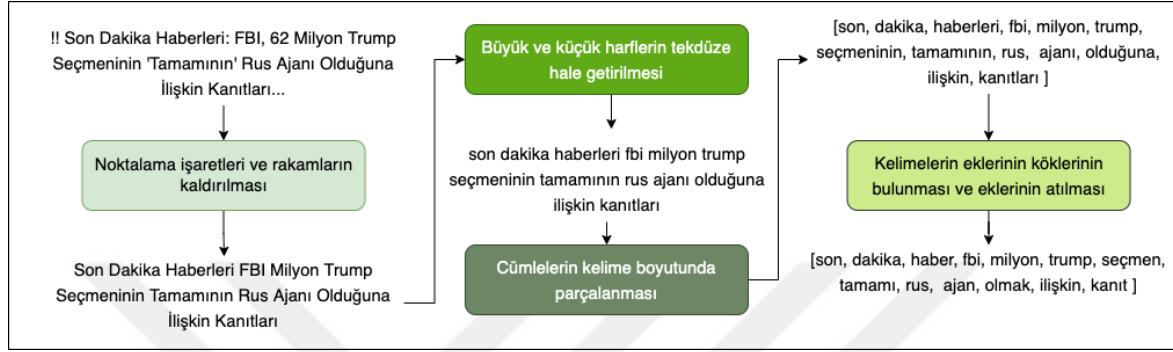
FakeNewsNet veri seti hem dengeli miktarda veri içermesi, açık kaynak olarak erişime sunulmuş ve erişilebilir olması açısından hem de verinin etiketlerinin teyit uzmanları aracılığı ile yapılmış olması nedeni ile çalışmada kullanılacak veri seti olarak seçilmiştir.

3.2. Metinlerin Ön İşlenmesi (Text Pre-Processing)

Veri setinin elde edilmesinin ardından veriler ön işleme adımlarından geçirilmiştir. Metin madenciliği, bir belge içerisinde yapılandırılmamış metinlerden veri elde edilmesidir. Özellikle metinler üzerinde makine öğrenmesi yöntemlerinin efektif bir şekilde kullanılabilmesi için bir karakter akışının anlamlı parçalara ayrılması gerekmektedir. Bu ayrımın doğru bir şekilde gerçekleşebilmesi için metinlerin belirli ön işleme adımlarından geçmesi gerekmektedir.

Metinler doğası gereği birçok gürültülü veri içermektedir. Metinlerin ön işlemenin amacı metin madenciliği tekniklerinin uygulanmasına hazır hale getirmek ve verimliliklerini arttırmaktır. Metin ön işleme sadece metin madenciliği tekniklerini değil makine öğrenme algoritmalarının performansını da olumlu yönde etkilemektedir.

Şekil 3.2.1. de çalışmada Türkçe dilinde sahte haber tespiti yapılırken kullanılan 4 adet ön işleme yöntemi gösterilmiştir. Bunlar; Noktalama İşaretlerinin ve rakamların kaldırılması (Removing punctuation and digits), büyük ve küçük harflerin tekdüze hale getirilmesi (Normalization), cümlelerin kelime boyutunda parçalanması (Tokenization) ve kelimelerin köklerinin bulunması ve eklerin atılması (Stemming) işlemleridir.



Şekil 3.2.1. Çalışmada kullanılan metin ön işleme metodlarının örnek ile açıklanması

3.2.1. Noktalama işaretleri ve rakamların kaldırılması

Sahte haber tespiti yaparken metinler içerisinde bulunan noktalama işaretleri ve rakamlar özellik çıkarma yöntemleri ve makine öğrenmesi yöntemlerinin performansını olumsuz olarak etkiler. Her bir noktalama işareti ve her bir farklı rakam modellere ek bir parametre daha eklenmesine sebep olacak vektör uzunluğunu artırır. Bu durum modelin aşırı öğrenmesine ya da önemli girdilere gerekli önem seviyesinin verilememesine sebebiyet verir.

3.2.2. Büyük ve küçük harflerin tekdüze hale getirilmesi

Metinler içerisinde birçok büyük ve küçük karakter içerebilir. Bu durum aynı iki kelimenin yazımından ötürü farklı kelimeler olarak ele alınmasına sebebiyet verebilmektedir. Bahsedilen problemin önlenmesi için Şekil 3.2.1. üzerinde sunulan örnekte olduğu gibi karakterlerin tekdüze hale indirilmesi gerekmektedir. Şekil 3.2.1. üzerinde verilen örnekte noktalama işaretleri ve rakamların kaldırılması metni anlam bakımından etkilemiş de olsa kendinden sonra gelen ön işleme yöntemleri için temel oluşturmuştur.

3.2.3. Cümlelerin kelime boyutunda parçalanması

Doğal dil işlemedeki en önemli adımlardan biri metinleri anlamlı parçalara ayırabilmektir. Tokenizasyon cümlelerin veya metinlerin belirli bir boyutta parçalara ayrılması işlemi genellikle metin işleminin en önemli adımlarındandır. Bu adımın temel amacı, bir karakter akışını belirli boyutlarda parçalara dönüştürmektir. Bu parçalar ayrıştırılmadan herhangi bir analiz veya üretim yapmak mümkün değildir (Webster ve Kit, 1992). Parçalara ayırma işlemi karakter akışını belirli belirteçlere göre ayırma işlemidir. Bu belirteç belirli bir karakter sayısı olabileceği gibi kelime veya cümle de olabilir. Bu çalışmada metinler kelime boyutunda boşluk karakteri kullanılarak parçalara ayrılmıştır. Bu yöntem deyimlerin karmaşık yapısını veya gramer yapısını gözden kaçırma da kelimelileri tanımlamada basit ve efektif bir yöntem olarak karşımıza çıkmaktadır.

3.2.4. Kelimelerin köklerinin bulunması ve eklerinin atılması

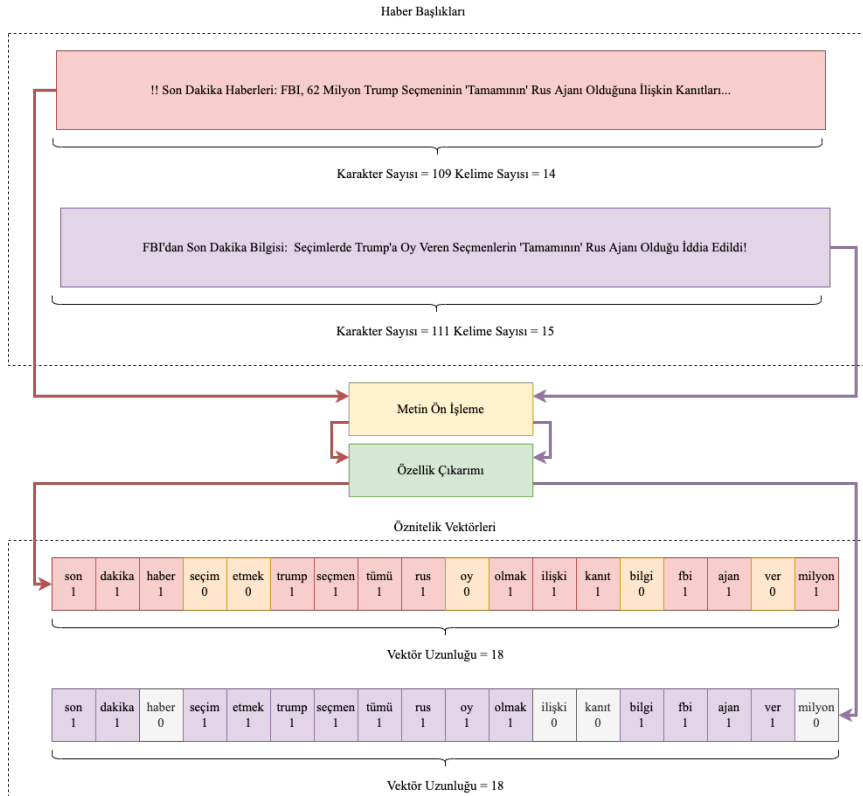
Metinler kelime boyutunda parçalanarak doğal dil işleme ve özellik çıkartım yöntemleri tarafından işlenebilecek formata getirilirler. Ancak bu adımda doğrudan oluşturulacak bir metin vektörünün çok fazla yakın anlamlı fakat ekleri ile farklılaşmış kelime içerecektir. Metin vektörünün uzunluğunun artması makine öğrenmesi algoritmalarına öğrenmesi için verilecek olan öz nitelik miktarının artması anlamına gelmektedir. Kelimelerin köküne indirgenmesi bir kelimeye verilmesi gereken öz nitelik öneminin ekler yüzünden azalmaması ve kelimenin köküne odaklanması için makine öğrenmesi algoritmalarının aşırı öğrenme probleminden kaçınılmasını sağlamaktadır (H. Zhu vd., 2018).

Her dilin kendine özgü bir karakteristiği ve grameri vardır. Bu durum köke indirgeme işlemini her dil için farklı tekniklerle gerçekleştirilmesini gerektirmektedir. Türkçe dili de dahil olmak üzere birçok köke indirgeme algoritması bulunmaktadır. İlk ve çok ünlü bir algoritma olan Lovins algoritması çok hızlı çalışmasına karşın birçok ekte sorun yaşamaktaydı (Lovins, 1968.). Lovins algoritmasının ardından birçok algoritma geliştirilmiştir. Özellikle Porter algoritması 10.000 kelimeyi 8,1 saniyede köke indirmeyi başaran ufak, basit ve hızlı bir algoritmadır. İngilizce dilinde en çok kullanılan köke indirgeme algoritmalarından biridir (Porter, 1980). Çalışmada ise Türkçe metinler üzerinde Porter ve Lovins algoritmalarına benzer yapıda sözlük ve kural tabanlı çalışarak köke indirme işlevi gören TRNLP adlı Python kütüphanesi kullanılmıştır.

3.3. Metinlerden Özellik Çıkarımı (Feature Extraction)

Ön işleme adımlarının ardından her biri farklı kelime grupları içeren ve farklı uzunluktaki haber metnininin, makine öğrenme algoritmalarının kullanabileceği eşit uzunluktaki ağırlıklandırılmış öznitelik vektörlerine dönüştürülmeleri gerekmektedir. Bu Adım metinlerden özellik çıkarımı olarak adlandırılmaktadır. Özellik çıkarım algoritmaları bir metni kullanım sıklıkları, belge içerisindeki kullanımları ve birbirleri arasındaki ilişkiyi gözeterak ağırlıklandırma işini üstlenmektedirler.

Ön işleme adımları ile anlamlı parçalara ayrılmış ve temizlenmiş metnin makine öğrenmesi algoritmalarının kullanılabileceği bir biçimde ifade edilmesi gerekmektedir. Bu biçime öznitelik vektörü denmektedir. Haber metinleri ele alındığında, her bir haber birbirinden farklı uzunlukta ve sayıda kelimedenden oluşmaktadır. Bu durum makine öğrenmesi algoritmalarının kullanımını kısıtlamaktadır. Makine öğrenmesi algoritmaları eğitilirken ve sonrasındaki süreçte aynı uzunlukta öznitelik vektörlerinin kullanımı gerektirmektedir. Bu sebeple haber metnininin tamamının aynı uzunlukta vektörler ile ifade edilmeleri gerekmektedir. Metinlerden özellik çıkartımı (Feature Extraction) yöntemleri, metinleri aynı uzunlukta öznitelik vektörlerine dönüştürülmesini sağlar. Şekil 3.3.1. Üzerinde farklı uzunluklardaki metinlerin aynı uzunluktaki öznitelik vektörüne dönüşümü gösterilmektedir.



Şekil 3.3.1. Metinden özellik çıkarımı ve kelime vektörlerinin elde edilmesi

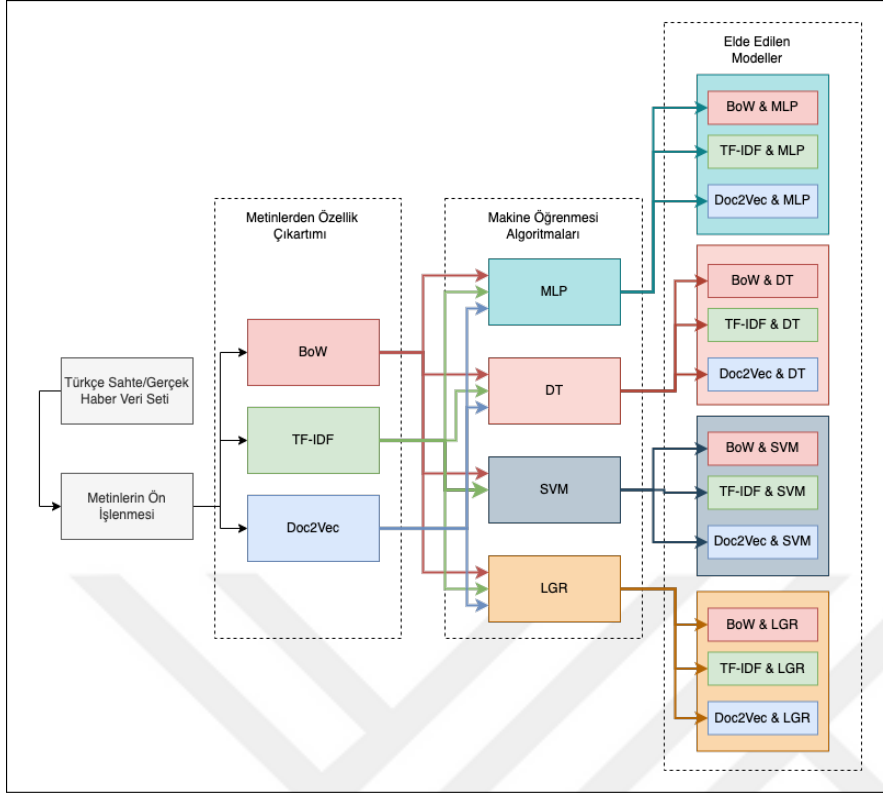
Metinlerin aynı uzunlukta öznitelik vektörlerine çevrilmesi için çalışmada 3 adet metinden özellik çıkarımı algoritmasından yararlanılmıştır. Bu algoritmalar BoW, TF-IDF ve Doc2Vec'dir (Algoritmalar Bölüm 2.2.1.'de detaylandırılmıştır).

3.4. Modellerin Oluşturulması

Özellik çıkartım yöntemleri ile öznitelik vektörü haline getirilmiş olan haber metinleri, ham metinlerin aksine makine öğrenmesi yöntemleri tarafından işlenebilmektedir. Bu aşamada her bir özellik çıkartım yöntemi ile elde edilmiş olan öznitelik vektörü MLP, DT, SVM ve LGR algoritmaları olmak üzere 4 adet makine öğrenmesi algoritmasının eğitiminde kullanılmıştır (Algoritmaların detayları bölüm 2.2.2. de verilmektedir.).

Türkçe gerçek ve sahte haber metinleri ön işleme adımlarından geçirildikten sonra, özellik çıkarma algoritmaları ile makine öğrenimi algoritmalarının işleyebileceği öznitelik vektörleri haline dönüştürülmüştür. Çalışmada kullanılan her makine öğrenmesi modeli BoW, TF-IDF ve Doc2Vec özellik çıkartım yöntemleri ile elde edilmiş olan öznitelik vektörleri ile eğitilerek Şekil 3.4.1. üzerinde de görüleceği gibi 12 farklı kombinasyonda model elde edilmiştir. Modeller Python programlama dilinde kodlanmış olup, makine öğrenmesi algoritmaları için SciKitLearn Kütüphanesi kullanılmıştır. Modeller 8 çekirdek işlemcili ve 16 GB bellekli bilgisayar üzerinde koşturulmuştur.

Her bir modelin eşit şartlarda performansının izlenebilmesi için, elde edilen bütün modeller aynı eğitim ve test verisine tabi tutulmuştur. Veri seti içerisinde bazı kelimeler çok yüksek ayırt ediciliğe sahip olmaktadır. Bu kelimelerin yalnızca eğitim veya test verisinde kalması model performanslarını doğrudan etkilemektedir. Eğitim ve test verisi 30 defa karıştırılarak, her bir iterasyon için ayrı ayrı eğitim ve test işlemi gerçekleştirilmiştir. Oluşturtulan modeller örnekleme yanlılığı oluşumunu kontrol etmek için K-fold yöntemi ile doğrulanmıştır.



Şekil 3.4.1 Makine Öğrenmesi Modellerinin Elde Edilmesi

3.5. Model Performansının Yorumlanması

Sahte haber tespiti için elde edilen 12 farklı modelin başarımlarının kıyaslanması gerekmektedir. Modelleri değerlendirmek için birçok yöntem bulunmaktadır. Bu yöntemlerin en bilineni doğruluk değeridir. Ancak doğruluk değeri tek başına bir model hakkında kanıya varmak için yeterli değildir. Karmaşıklık matrisi aracılığı ile elde edilen duyarlılık, kesinlik ve F1 değerleri ile elimizdeki bütün modellerin incelenmesi gerekmektedir (Karmaşıklık matrisi, duyarlılık, kesinlik ve F1 değerleri bölüm 2.2.3. de detayları ile verilmiştir).

3.6. Bulguların Yorumlanması

Bulguların yorumlanmasında iki adet yöntemden yararlanılmıştır. Model performans verileri Test/Eğitim yöntemi ile elde edilmiş ve K-fold yöntemi ile geçerlilikleri doğrulanmıştır.

Eğitim/Test ayırımı yöntemi sahte ve gerçek haber verileri 30, 60, 90 kez 80% eğitim ve 20% test verisi olarak rastgele ayrılmış ve her iterasyonda öznelik vektörleri yeniden oluşturulmuş ve modeller en baştan tekrar eğitilmiştir. Çalışmanın metodunun adımları Şekil 3.2. üzerinde sunulmuştur. Modellerin eğitimleri her iterasyon için eğitim skoru, test skoru, duyarlılık, kesinlik, F1 skoru verileri toplanmıştır.

K-fold geçerlilik yöntemi için k değeri 10 seçilmiştir. Her bir model K-fold geçerlilik yöntemine tabi tutulup her bir k parçacığının eğitimden elde edilen eğitim skoru, test skoru, duyarlılık, kesinlik, F1 skoru verileri toplanmıştır. Sonuçlar her bir kombinasyonun ortalaması alınarak elde edilmiştir.

3.6.1. BoW ve MLP Modeli

BoW özellik çıkartım yöntemi ile öznitelik vektörü haline dönüştürülmüş olan sahte ve gerçek haber metinleri MLP makine öğrenmesi algoritması eğitilerek sınıflandırılmıştır. Çizelge 3.6.1.1. üzerinde verilmiş olan performans değerleri dikkate alındığında Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilmiş model 30 iterasyonda ortalama 0.86, maksimum 0.96 eğitim performansı elde edilmiştir.

3.6.1.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile BoW Özellik çıkartımı ve MLP makine öğrenmesi yönteminin 30 iterasyon sonuçları.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
count	30.000000	30.000000	30.000000	30.000000	30.000000
mean	0.863672	0.734444	0.719784	0.720733	0.716374
Std	0.063293	0.038517	0.085502	0.050409	0.071052
Min	0.591017	0.590476	0.295238	0.500000	0.371257
25%	0.851359	0.719048	0.715072	0.708154	0.703986
50%	0.862884	0.740476	0.731884	0.729464	0.731207
75%	0.894504	0.760714	0.752934	0.742756	0.748616
Max	0.962175	0.785714	0.788793	0.781977	0.777222

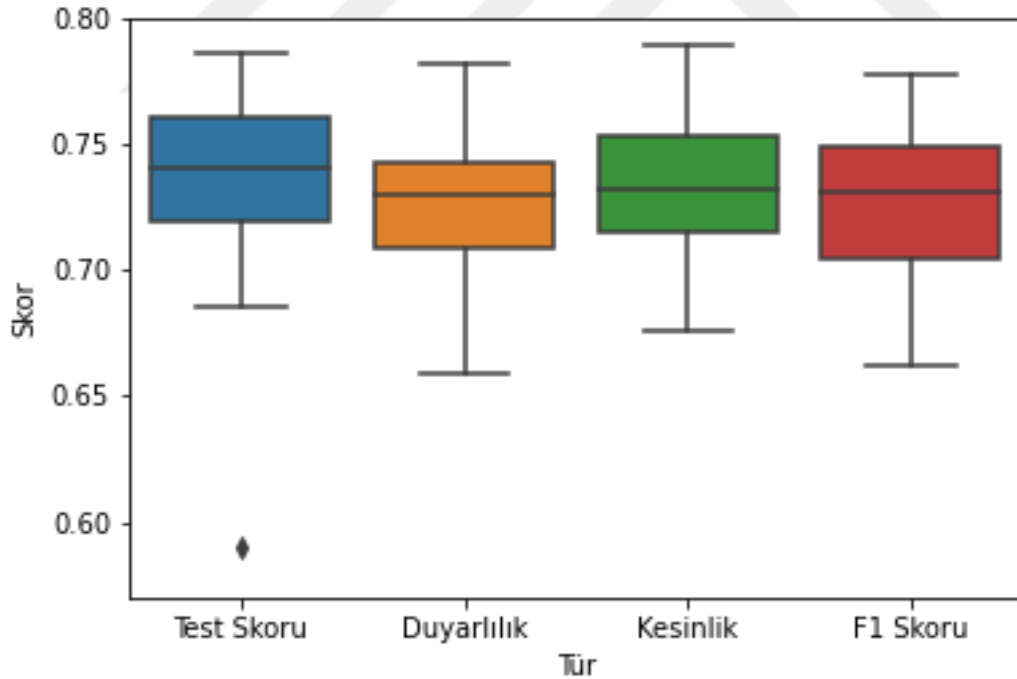
Şekil 3.6.1.1. incelendiğinde Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test, duyarlılık, kesinlik ve F1 skorunun minimum ve maksimum performans değerlerinde dalgalanmalar olsa da ortalama performans değerlerinin birbirine yakın seyrettiği görülmektedir. En düşük performans değeri kesinlik değerinde 0,29 olarak gerçekleşmiştir.

Model K-fold yöntemi ile BoW özellik çıkarımı ve MLP makine öğrenmesi algoritması ile elde edilen modelin doğrulama gerçekleştirildiğinde elde edilen eğitim, test, kesinlik ve F1 skoru değerleri Çizelge 3.6.1.2. de gösterilmiştir. Test/Eğitim ayrımı yöntemi ile elde edilmiş BoW MLP modelinin güvenilirliği K-fold yönteminin sonuçları ile desteklendiği ve modelde örnekleme yanlılığının olmadığı görülmektedir.

Çizelge 3.6.1.2. BoW Özellik çıkartımı ve MLP makine öğrenmesi yönteminin K-fold sonucu.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
K-Fold	0.891471	0.734103	0.698694	0.721200	0.706102

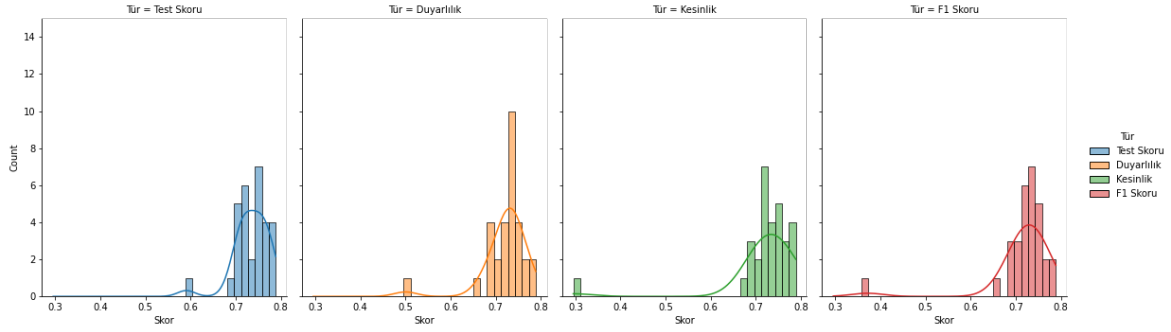
Şekil 3.6.1.1. İncelendiğinde Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin yalnızca test değerlerinde aykırı bir değer oluştuğu görülmektedir.



Şekil 3.6.1.1. Test/Eğitim ayrımı ile elde edilen verilerin BoW özellik çıkartımı ve MLP makine öğrenmesi yöntemi 30 iterasyonun sonuçlarının kutu grafiği.

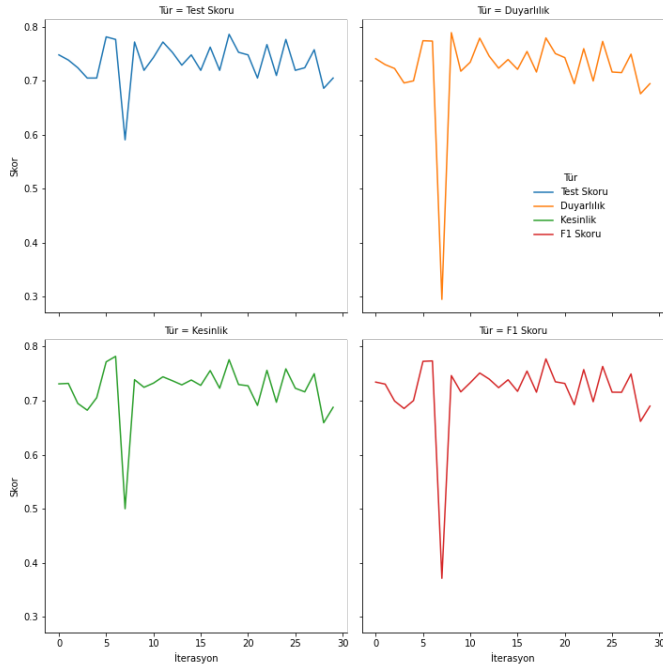
Şekil 3.6.1.2. üzerinde eğitim ve test verisi Test/Eğitim ayrımı yöntemi ile elde edilmiş BoW Özellik çıkarımı ve MLP makine öğrenmesi algoritması ile oluşturulmuş olan modelin

sonuçlarının histogram grafiği sunulmuştur. Test, kesinlik ve duyarlılık verileri Kolmogorov-Smirnov testine göre normal dağılmıştır.



Şekil 3.6.1.2. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş BoW Özellik çıkartımı ve MLP makine öğrenmesi yönteminin 30 iterasyonun sonuçlarının histogram grafiği.

Test/Eğitim ayrımı ile elde edilen eğitim ve test verileri ile eğitilen modelin 30 iterasyonda elde ettiği performans değerlerini Şekil 3.6.1.3. üzerinden incelediğimizde 7. İterasyon da gerçekleşen eğitim test verisi kombinasyonu dışındaki dağılımlarda ciddi dalgalanmalar göstermediği görülmektedir. Kesinlik performans değeri ortalama 0,71 olarak gerçekleşmiş ve standart sapması 0,09 olarak gerçekleşmiştir. Duyarlılık performans değeri ortalama 0,72 olarak gerçekleşmiş ve standart sapması 0,05 olarak gerçekleşmiştir. F1 Skoru ortalama 0,72 olarak gerçekleşmiş ve standart sapması 0,07 olarak gerçekleşmiştir.



Şekil 3.6.1.3 Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş BoW Özellik çıkartımı ve MLP makine öğrenmesi modelinin 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.

3.6.2. BoW ve DT Modeli

BoW özellik çıkartım yöntemi ile öznitelik vektörü haline dönüştürülmüş olan sahte ve gerçek haber metinleri DT makine öğrenmesi algoritması eğitilerek sınıflandırılmıştır. Çizelge 3.6.2.1. üzerinde verilmiş olan performans değerleri dikkate alındığında Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile model 30 iterasyonda ortalama 0,99, maksimum 1,0 eğitim performansı elde edilmiştir.

Çizelge 3.6.2.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile BoW özellik çıkartımı ve DT makine öğrenmesi yönteminin 30 iterasyon sonuçları.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
count	30,000000	30,000000	30,000000	30,000000	30,000000
mean	0,998936	0,769206	0,767653	0,751475	0,755693
std	0,001046	0,031257	0,035249	0,028588	0,030693
min	0,994090	0,690476	0,688821	0,695143	0,687350
25%	0,998818	0,758333	0,754785	0,733238	0,739532
50%	0,998818	0,771429	0,771523	0,753001	0,758988
75%	0,998818	0,784524	0,785529	0,765379	0,770831
max	1,000000	0,828571	0,838462	0,808515	0,816434

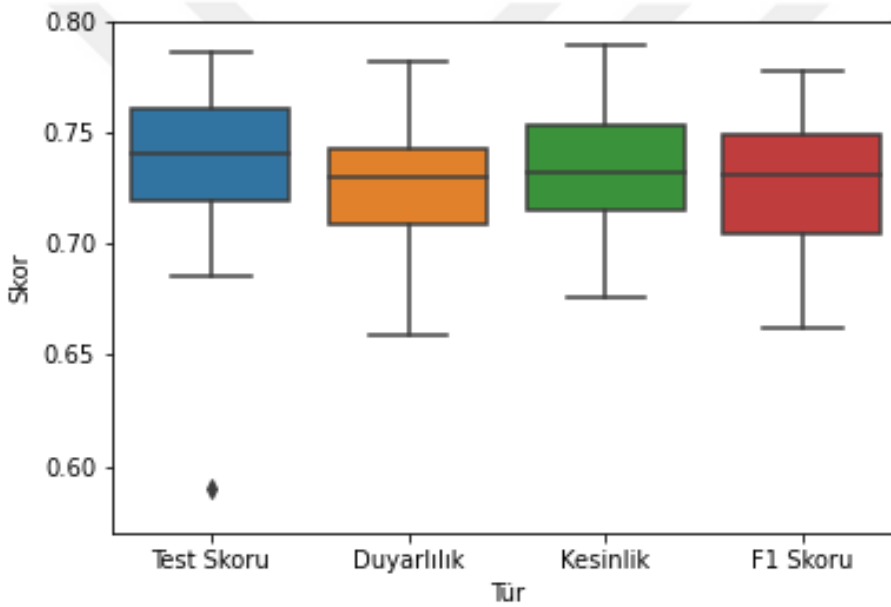
Çizelge 3.6.2.1. incelendiğinde Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin kesinlik değerinin duyarlılık değerinden 0.1 daha iyi performans sergilediği görülmektedir. Modelin 30 iterasyonda elde ettiği en iyi değer 0,81 ortalama 0,75 olarak gerçekleşmiştir.

Model K-fold yöntemi ile BoW özellik çıkarımı ve DT makine öğrenmesi algoritması ile elde edilen modelin doğrulama gerçekleştirildiğinde elde edilen eğitim, test, kesinlik ve F1 skoru değerleri Çizelge 3.6.2.2. de gösterilmiştir. Test/Eğitim ayrımı yöntemi ile elde edilmiş BoW DT modelinin güvenilirliği K-fold yönteminin sonuçları ile desteklendiği ve modelde örnekleme yanlılığının olmadığı görülmektedir.

Çizelge 3.6.2.2. BoW Özellik çıkartımı ve DT makine öğrenmesi yönteminin K-fold sonucu.

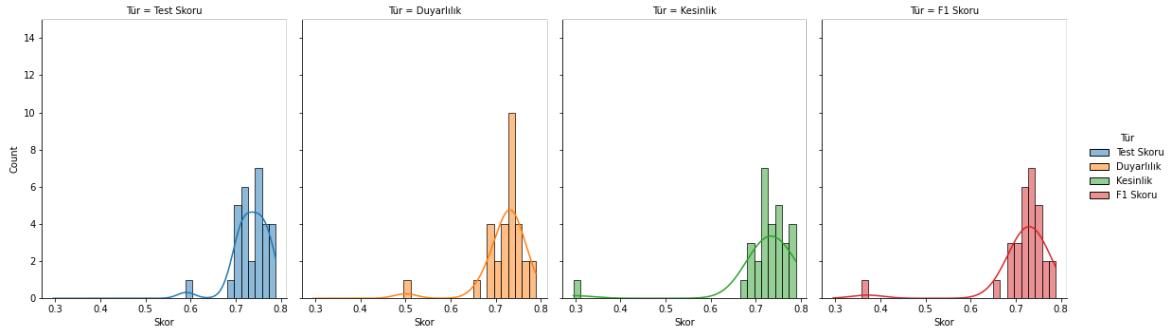
	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
K-Fold	0.998842	0.760502	0.758177	0.747412	0.747412

Şekil 3.6.2.1. İncelendiğinde Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin yalnızca test değerlerinde aykırı bir değer olduğu görülmektedir.



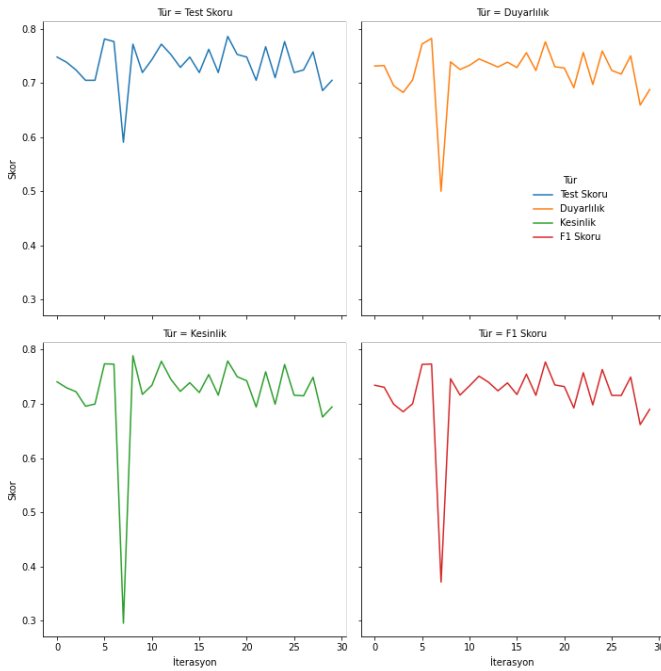
Şekil 3.6.2.1. Test/Eğitim ayrımı ile elde edilen verilerin BoW Özellik çıkartımı ve DT makine öğrenmesi yönteminin 30 iterasyonda elde ettiği sonuçlarının kutu grafiği.

Şekil 3.6.2.2. üzerinde eğitim ve test verisi Test/Eğitim ayrımı yöntemi ile elde edilmiş BoW Özellik çıkarımı ve DT makine öğrenmesi algoritması ile oluşturulmuş olan modelin sonuçlarının histogram grafiği sunulmuştur. Test, kesinlik ve duyarlılık verileri Kolmogorov-Smirnov testine göre normal dağılmıştır.



Şekil 3.6.2.2. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş BoW özellik çıkartımı ve DT makine öğrenmesi yönteminin 30 iterasyonun sonuçlarının histogram grafiği.

Test/Eğitim ayrımı ile elde edilen eğitim ve test verileri ile eğitilen modelin 30 iterasyonda elde ettiği sonuçları Şekil 3.6.1.2. üzerinden incelendiğinde, 4. İterasyon da gerçekleşen eğitim test verisi kombinasyonu dışındaki dağılımlarda pozitif ve negatif olarak 0,10 miktarında dalgalanma göze çarpmaktadır. Duyarlılık Skoru ortalama 0,77 olarak gerçekleşmiş ve standart sapması 0,04 olarak gerçekleşmiştir. Kesinlik skoru ortalama 0,75 olarak gerçekleşmiş ve standart sapması 0,03 olarak gerçekleşmiştir. F1 Skoru ortalama 0,76 olarak gerçekleşmiş ve standart sapması 0,03 olarak gerçekleşmiştir. Modelin kesinlik değerinde dikkate değer bir performans sapması görülmektedir. Bu durum kurulan bu modelin gerçek haberleri sahte haber olarak etiketleme hatasına yatkın olduğunu göstermektedir.



Şekil 3.6.2.3. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş BoW Özellik çıkartımı ve DT makine öğrenmesi modelinin 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.

3.6.3. BoW ve SVM Modeli

BoW özellik çıkartım yöntemi ile öznitelik vektörü haline dönüştürülmüş olan sahte ve gerçek haber metinleri SVM makine öğrenmesi algoritması eğitilerek sınıflandırılmıştır. Çizelge 3.6.3.1. üzerinde verilmiş olan değerler dikkate alındığında Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilmiş model 30 iterasyonda ortalama 0,76, maksimum 0,77 eğitim skoru elde edilmiştir.

Çizelge 3.6.3.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile BoW Özellik çıkartımı ve SVM makine öğrenmesi yönteminin 30 iterasyon sonuçları.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
count	30,000000	30,000000	30,000000	30,000000	30,000000
mean	0,761584	0,734127	0,752822	0,695402	0,698903
std	0,005537	0,018969	0,025116	0,021748	0,024259
min	0,749409	0,685714	0,697110	0,639441	0,635417
25%	0,756797	0,719048	0,733487	0,678803	0,680869
50%	0,761820	0,733333	0,756250	0,696455	0,700957
75%	0,766844	0,747619	0,771936	0,715374	0,720728
max	0,768322	0,771429	0,815201	0,729839	0,736979

Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test skoru Çizelge 3.6.3.1. incelendiğinde ortalama eğitim değerinden 0,03 azalarak ortalama 0,73, standart sapması 0,01 olarak gerçekleşmiştir. Maksimum 0,77 test performans değeri elde edilmiştir.

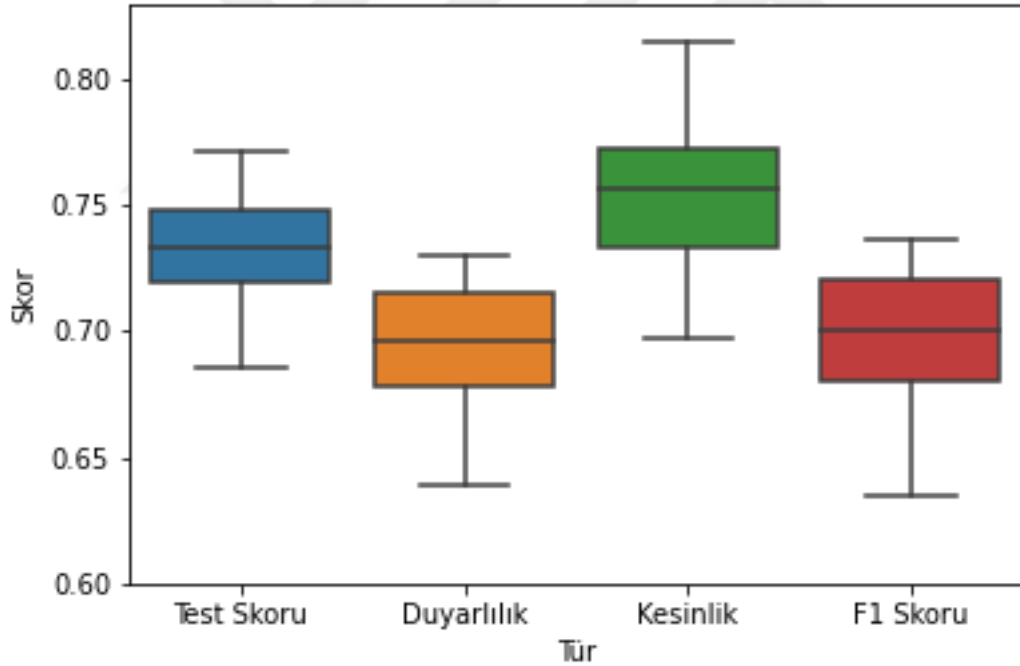
Model K-fold yöntemi ile BoW özellik çıkarımı ve SVM makine öğrenmesi algoritması ile elde edilen modelin doğrulama gerçekleştirildiğinde elde edilen eğitim, test, kesinlik ve F1 skoru değerleri Çizelge 3.6.3.2. de gösterilmiştir. Test/Eğitim ayrımı yöntemi ile elde edilmiş BoW SVM modelinin güvenilirliği K-fold yönteminin sonuçları ile desteklendiği

ve modelde örnekleme yanlılığının olmadığı görülmektedir.

Çizelge 3.6.3.2. BoW Özellik çıkartımı ve SVM makine öğrenmesi yönteminin K-fold sonucu.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
K-Fold	0.761680	0.728104	0.736664	0.692256	0.695110

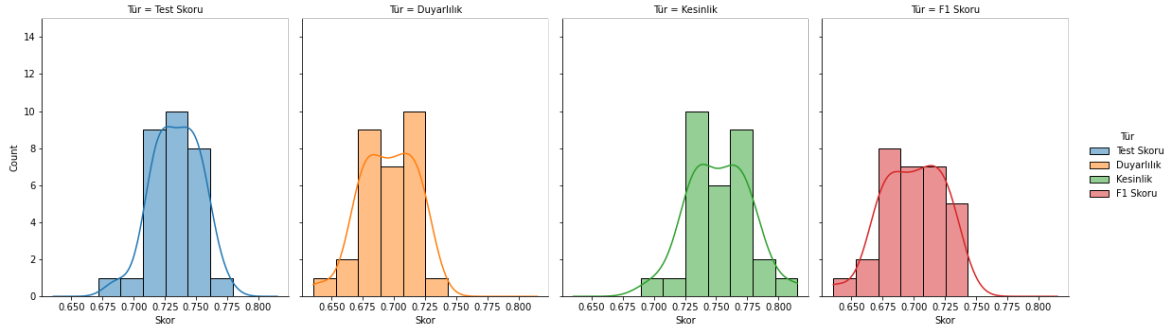
Şekil 3.6.3.1. incelendiğinde Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test, duyarlılık, kesinlik ve F1 skorunun minimum ve maksimum performans değerlerinde dalgalanmalar görülmektedir. Modelin kesinlik değeri, duyarlılık değerinden daha başarılı performans elde etmiştir. En düşük ortalama performans değeri duyarlılık değerinde 0,69 olarak gerçekleşmiştir.



Şekil 3.6.3.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen BoW özellik çıkartımı ve SVM makine öğrenmesi yönteminin 30 iterasyon sonuçlarının kutu grafiği.

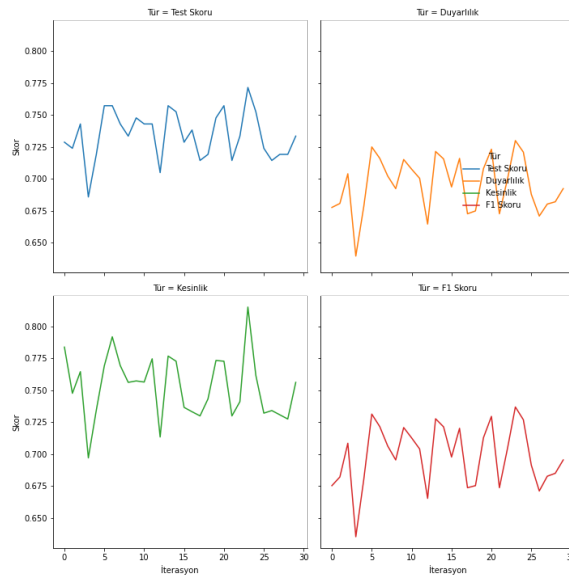
Şekil 3.6.3.2. üzerinde eğitim ve test verisi Test/Eğitim ayrımı yöntemi ile elde edilmiş BoW Özellik çıkarımı ve SVM makine öğrenmesi algoritması ile oluşturulmuş olan modelin

sonuçlarının histogram grafiği sunulmuştur. Test, kesinlik ve duyarlılık verileri Kolmogorov-Smirnov testine göre normal dağılmıştır.



Şekil 3.6.3.2. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş BoW Özellik çıkartımı ve SVM makine öğrenmesi yönteminin 30 iterasyonun sonuçlarının histogram grafiği.

Test/Eğitim ayrımı ile elde edilen eğitim ve test verileri ile eğitilen modelin 30 iterasyonda elde ettiği performans skorları Şekil 3.6.3.3. üzerinden incelendiğinde, 3. İterasyon da gerçekleşen eğitim test verisi kombinasyonu dışındaki dağılımlarda pozitif ve negatif olarak 0,07 miktarında dalgalanma göze çarpmaktadır. Kesinlik performans değeri ortalama 0,75 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. Duyarlılık performans değeri ortalama 0,69 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. F1 Skoru ortalama 0,69 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. Modelin duyarlılık değerinde dikkate değer bir performans sapması görülmektedir. Bu durum kurulan bu modelin sahte haberleri gerçek haber olarak etiketleme hatasına yatkın olduğunu göstermektedir.



Şekil 3.6.3.3. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş BoW Özellik çıkartımı ve SVM makine öğrenmesi modelinin 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.

3.6.4. BoW ve LGR Modeli

BoW özellik çıkartım yöntemi ile öznitelik vektörü haline dönüştürülmüş olan sahte ve gerçek haber metinleri LGR makine öğrenmesi algoritması eğitilerek sınıflandırılmıştır. Çizelge 3.6.4.1. üzerinde verilmiş olan değerler dikkate alındığında Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilmiş model 30 iterasyonda ortalama 0,99, maksimum 0,99 eğitim skoru elde edilmiştir.

Çizelge 3.6.4.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile BoW Özellik çıkartımı ve LGR makine öğrenmesi yönteminin 30 iterasyon sonuçları.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
count	30.000000	30.000000	30.000000	30.000000	30.000000
mean	0.990623	0.783810	0.778368	0.775244	0.775615
std	0.008165	0.022391	0.023056	0.022661	0.022986
min	0.958629	0.752381	0.744934	0.731995	0.737828
25%	0.989362	0.766667	0.759199	0.760221	0.757464
50%	0.992317	0.780952	0.774122	0.771099	0.771360
75%	0.995272	0.795238	0.796314	0.788400	0.787924
max	0.998818	0.833333	0.828811	0.825019	0.826729

Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test skoru Çizelge 3.6.4.1 incelendiğinde ortalama eğitim değerinden 0,21 azalarak ortalama 0,78, standart sapması 0,02 olarak gerçekleşmiştir. Maksimum 0,83 test performans değeri elde edilmiştir.

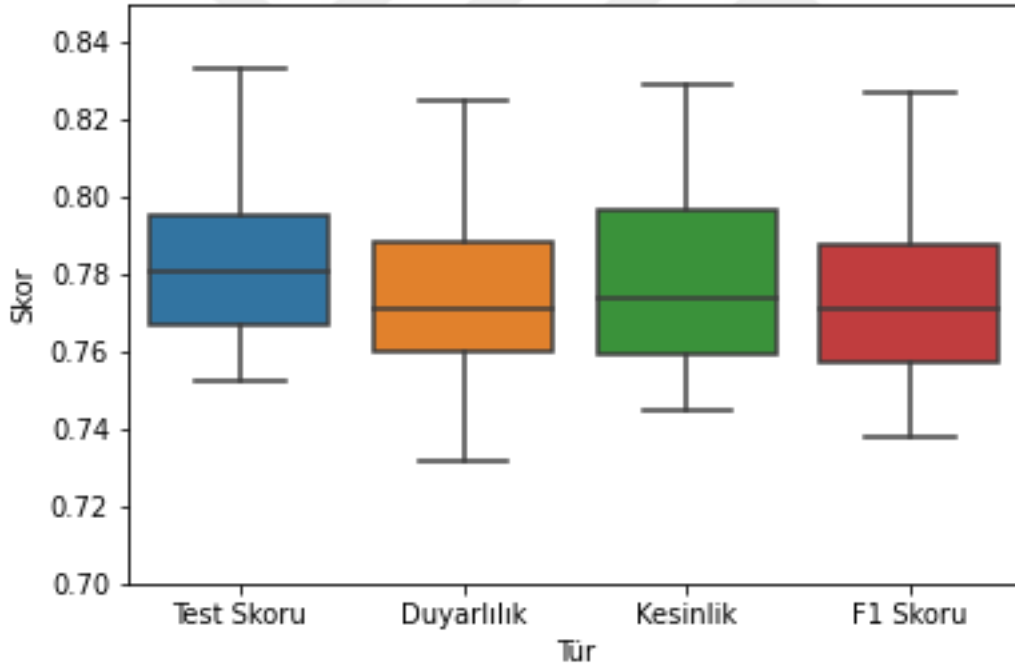
Model K-fold yöntemi ile BoW özellik çıkarımı ve LGR makine öğrenmesi algoritması ile elde edilen modelin doğrulama gerçekleştirildiğinde elde edilen eğitim, test, kesinlik ve F1 skoru değerleri Çizelge 3.6.4.2. de gösterilmiştir. Test/Eğitim ayrımı yöntemi ile elde edilmiş BoW LGR modelinin güvenilirliği K-fold yönteminin sonuçları ile desteklendiği

ve modelde örnekleme yanlılığının olmadığı görülmektedir.

Çizelge 3.6.4.2. BoW Özellik çıkartımı ve LGR makine öğrenmesi yönteminin K-fold sonucu.

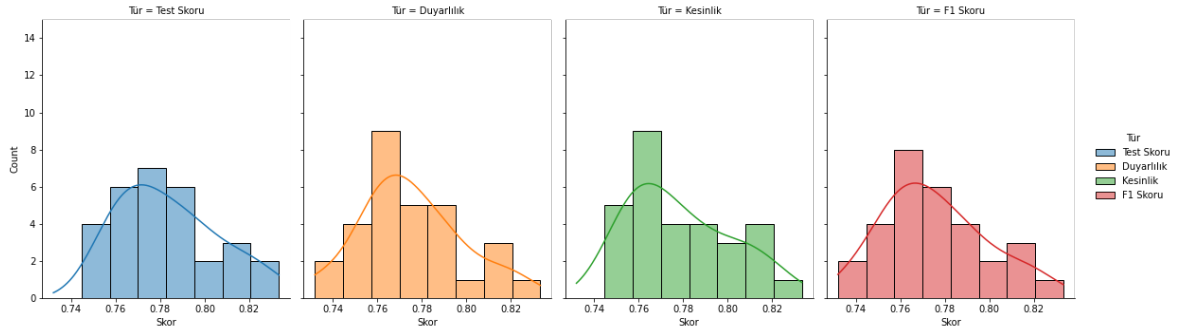
	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
K-Fold	0.980110	0.797270	0.793144	0.787800	0.789004

Şekil 3.6.4.1. incelendiğinde Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test, duyarlılık, kesinlik ve F1 skorunun minimum ve maksimum değerlerinin birbirine yakın seyrettiği görülmektedir. En düşük ortalama performans değeri 0,7752 olarak duyarlılık skorunda gerçekleşmiştir.



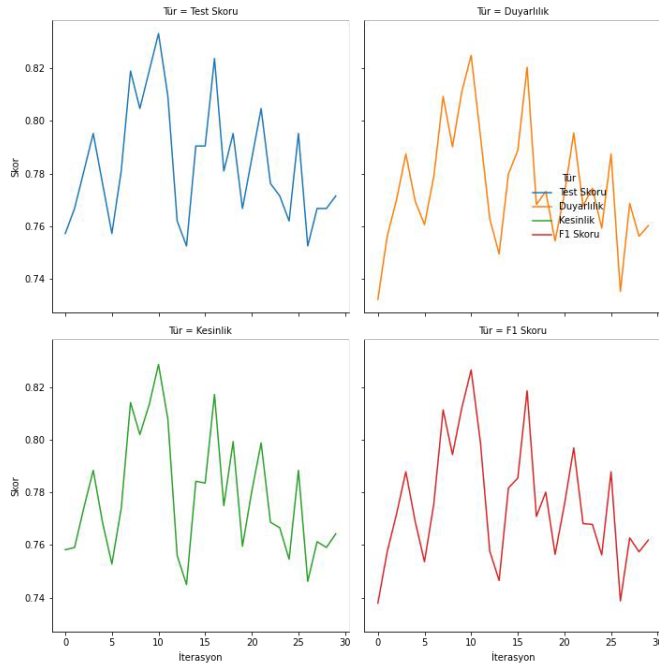
Şekil 3.6.4.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen BoW özellik çıkartımı ve LGR makine öğrenmesi modelinin 30 iterasyon sonuçlarının kutu grafiği.

Şekil 3.6.4.2. üzerinde eğitim ve test verisi Test/Eğitim ayrımı yöntemi ile elde edilmiş BoW Özellik çıkarımı ve LGR makine öğrenmesi algoritması ile oluşturulmuş olan modelin sonuçlarının histogram grafiği sunulmuştur. Test, kesinlik ve duyarlılık verileri Kolmogorov-Smirnov testine göre normal dağılmıştır.



Şekil 3.6.4.2. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş BoW Özellik çıkartımı ve LGR makine öğrenmesi yönteminin 30 iterasyonun sonuçlarının histogram grafiği.

Test/Eğitim ayrımı ile elde edilen eğitim ve test verileri ile eğitilen modelin 30 iterasyonda elde ettiği sonuçları Şekil 3.6.4.3. üzerinden incelendiğinde. 26. İterasyon da gerçekleşen eğitim test verisi kombinasyonu dışındaki dağılımlarda pozitif ve negatif olarak 0,08 miktarında dalgalanma göze çarpmaktadır. Kesinlik Skoru ortalama 0,77 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. Duyarlılık skoru ortalama 0,77 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. F1 Skoru ortalama 0,77 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. Modelin duyarlılık, kesinlik ve F1 skorunun eşit derecede gerçekleşmesi modelin Tip 1 ve Tip 2 hataya yatkınlığının aynı seviyede olduğunu göstermektedir.



Şekil 3.6.4.3. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş BoW Özellik çıkartımı ve LGR makine öğrenmesi modelinin 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.

3.6.5. TF-IDF ve MLP Modeli

TF-IDF özellik çıkartım yöntemi ile öznelik vektörü haline dönüştürülmüş olan sahte ve gerçek haber metinleri MLP makine öğrenmesi algoritması eğitilerek sınıflandırılmıştır. Çizelge 3.6.5.1. üzerinde verilmiş olan değerler dikkate alındığında Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilmiş model 30 iterasyonda ortalama 0,98, maksimum 1,00 eğitim skoru elde edilmiştir.

Çizelge 3.6.5.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile TF-IDF Özellik çıkartımı ve MLP makine öğrenmesi yönteminin 30 iterasyon sonuçları.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
count	30.000000	30.000000	30.000000	30.000000	30.000000
mean	0.984437	0.777143	0.764927	0.759740	0.758796
std	0.074316	0.042452	0.092562	0.054227	0.077022
min	0.591017	0.590476	0.295238	0.500000	0.371257
25%	0.996749	0.766667	0.758842	0.746648	0.751472
50%	0.998227	0.785714	0.784566	0.769083	0.772185
75%	0.998818	0.798810	0.800356	0.788916	0.791491
max	1.000000	0.823810	0.829010	0.811140	0.812235

Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test skoru Çizelge 3.6.5.1. incelendiğinde ortalama eğitim değerinden 0,21 azalarak ortalama 0,78, standart sapması 0,04 olarak gerçekleşmiştir. Maksimum 0,82 test performans değeri elde edilmiştir.

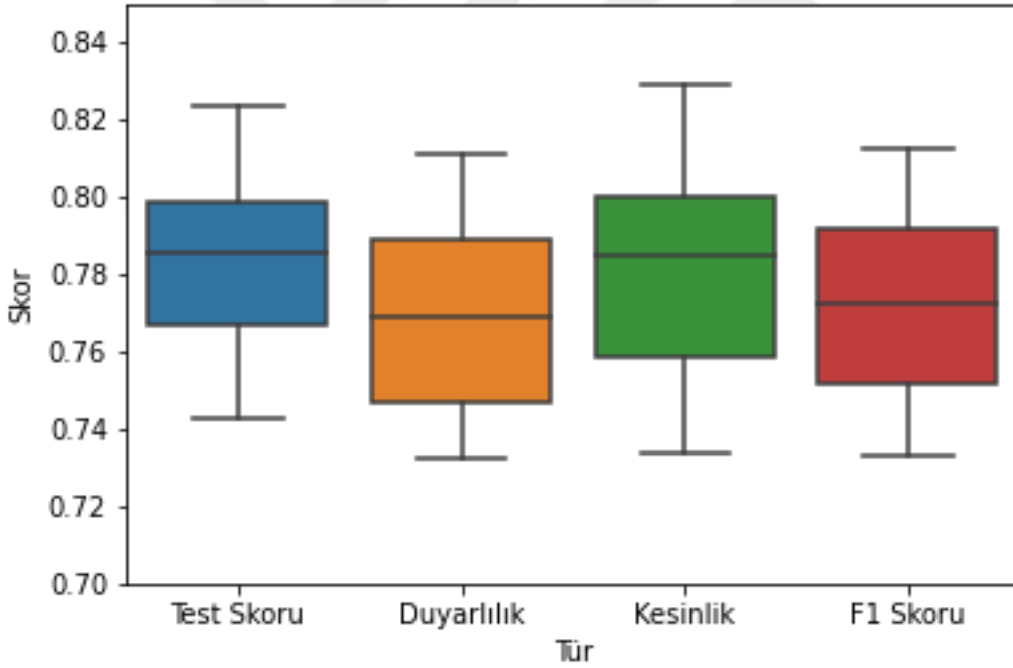
Model K-fold yöntemi ile TF-IDF özellik çıkarımı ve MLP makine öğrenmesi algoritması ile elde edilen modelin doğrulama gerçekleştirildiğinde elde edilen eğitim, test, kesinlik ve F1 skoru değerleri Çizelge 3.6.5.2. de gösterilmiştir. Test/Eğitim ayrımı yöntemi ile elde edilmiş TF-IDF MLP modelinin güvenilirliği K-fold yönteminin sonuçları ile desteklendiği

ve modelde örnekleme yanlılığının olmadığı görülmektedir.

Çizelge 3.6.5.2. TF-IDF Özellik çıkartımı ve MLP makine öğrenmesi yönteminin K-fold sonucu.

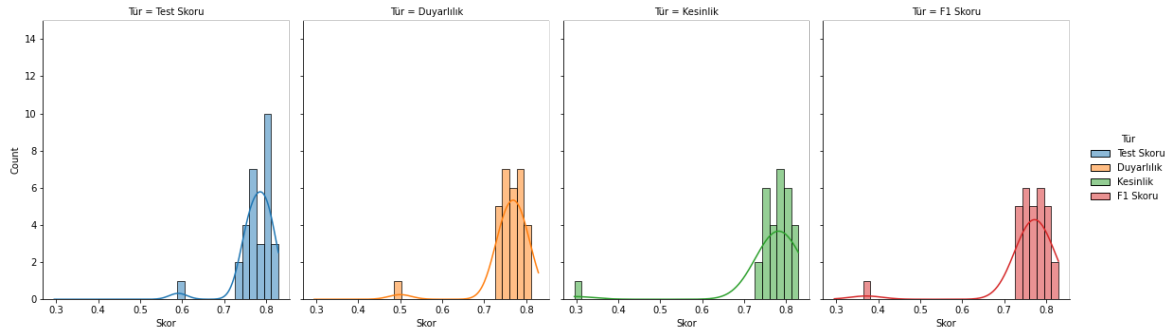
	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
K-Fold	0.998843	0.803955	0.800939	0.793866	0.795689

Şekil 3.6.5.1. incelendiğinde Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test, duyarlılık, kesinlik ve F1 skorunun minimum ve maksimum değerlerinin birbirine yakın seyrettiği görülmektedir. En düşük ortalama performans değeri 0,75 olarak F1 skorunda gerçekleşmiştir.



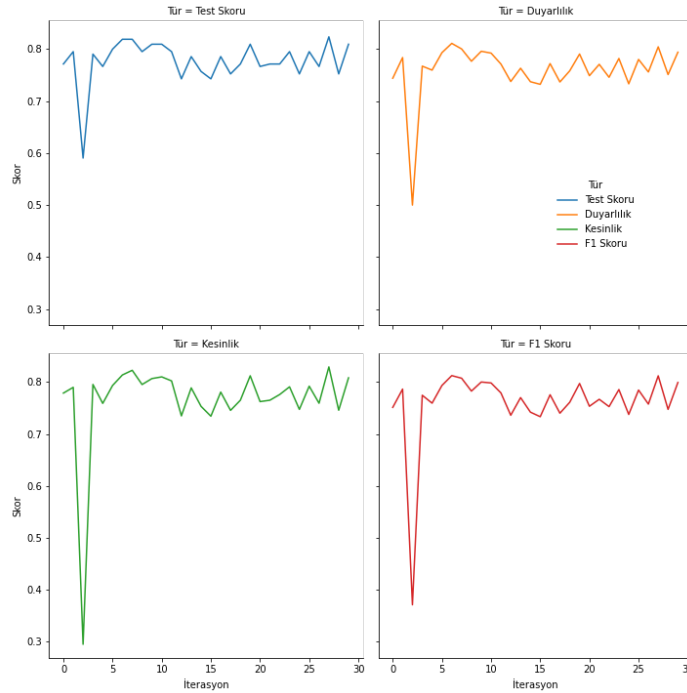
Şekil 3.6.5.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen TF-IDF Özellik çıkartımı ve MLP makine öğrenmesi modelinin 30 iterasyon sonuçlarının kutu grafiği.

Şekil 3.6.5.2. üzerinde eğitim ve test verisi Test/Eğitim ayrımı yöntemi ile elde edilmiş TF-IDF Özellik çıkarımı ve MLP makine öğrenmesi algoritması ile oluşturulmuş olan modelin sonuçlarının histogram grafiği sunulmuştur. Test, kesinlik ve duyarlılık verileri Kolmogorov-Smirnov testine göre normal dağılmıştır.



Şekil 3.6.5.2. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş TF-IDF özellik çıkartımı ve MLP makine öğrenmesi yönteminin 30 iterasyonun sonuçlarının histogram grafiği.

Test/Eğitim ayrımı ile elde edilen eğitim ve test verileri ile eğitilen modelin 30 iterasyonda elde ettiği sonuçları Şekil 3.6.5.3. üzerinden incelendiğinde. 3. İterasyon da gerçekleşen eğitim test verisi kombinasyonu dışındaki dağılımlarda pozitif ve negatif olarak 0,10 miktarında dalgalanma göze çarpmaktadır. Kesinlik Skoru ortalama 0,76 olarak gerçekleşmiş ve standart sapması 0,04 olarak gerçekleşmiştir. Duyarlılık skoru ortalama 0,75 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. F1 Skoru ortalama 0,77 olarak gerçekleşmiş ve standart sapması 0,05 olarak gerçekleşmiştir. Modelin duyarlılık, kesinlik ve F1 skoru arasında 0,01 sapma meydana gelmiştir.



Şekil 3.6.5.3. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş TF-IDF Özellik çıkartımı ve MLP makine öğrenmesi modelinin 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.

3.6.6. TF-IDF ve DT Modeli

TF-IDF özellik çıkartım yöntemi ile öznelik vektörü haline dönüştürülmüş olan sahte ve gerçek haber metinleri DT makine öğrenmesi algoritması eğitilerek sınıflandırılmıştır. Çizelge 3.6.6.1. üzerinde verilmiş olan değerler dikkate alındığında Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilmiş model 30 iterasyonda ortalama 0,99, maksimum 1,00 eğitim skoru elde edilmiştir.

Çizelge 3.6.6.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile TF-IDF Özellik çıkartımı ve DT makine öğrenmesi yönteminin 30 iterasyon sonuçları.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
count	30.000000	30.000000	30.000000	30.000000	30.000000
mean	0.999133	0.776508	0.774632	0.758311	0.762761
std	0.000532	0.028432	0.030505	0.030664	0.031052
min	0.998818	0.723810	0.714272	0.704239	0.708582
25%	0.998818	0.757143	0.751561	0.738888	0.742020
50%	0.998818	0.778571	0.773902	0.761487	0.765290
75%	0.999704	0.794048	0.790752	0.774827	0.780098
max	1.000000	0.842857	0.850289	0.824175	0.832167

Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test skoru Çizelge 3.6.6.1. incelendiğinde ortalama eğitim değerinden 0,22 azalarak ortalama 0,77, standart sapması 0,02 olarak gerçekleşmiştir. Maksimum 0,84 test performans değeri elde edilmiştir.

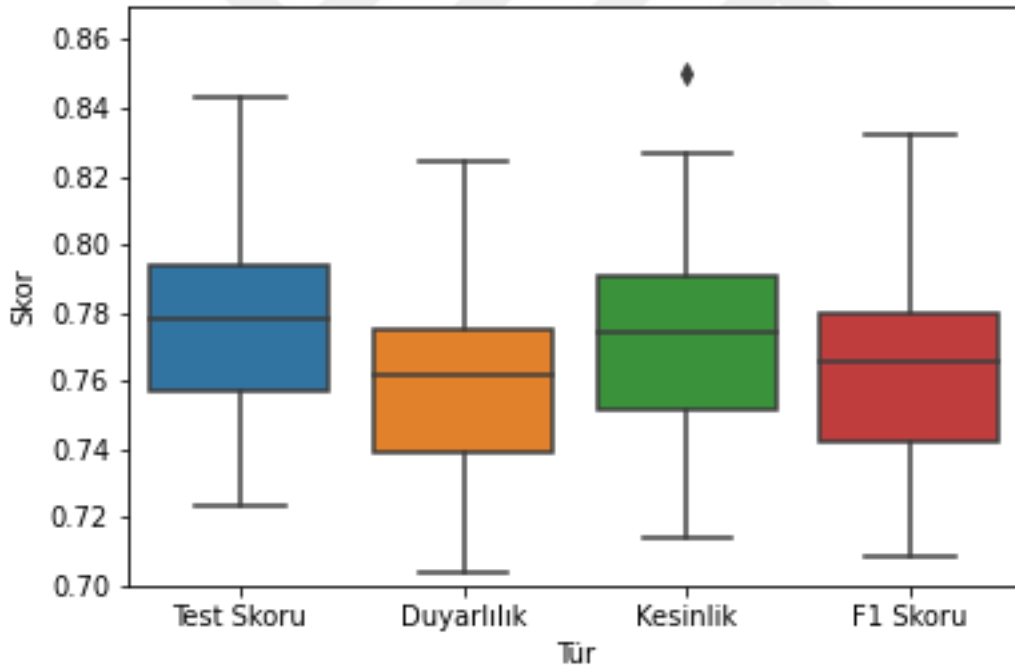
Model K-fold yöntemi ile TF-IDF özellik çıkarımı ve DT makine öğrenmesi algoritması ile elde edilen modelin doğrulama gerçekleştirildiğinde elde edilen eğitim, test, kesinlik ve F1 skoru değerleri Çizelge 3.6.6.2. de gösterilmiştir. Test/Eğitim ayrımı yöntemi ile elde edilmiş TF-IDF DT modelinin güvenilirliği K-fold yönteminin sonuçları ile desteklendiği

ve modelde örnekleme yanlılığının olmadığı görülmektedir.

Çizelge 3.6.6.2. TF-IDF Özellik çıkartımı ve DT makine öğrenmesi yönteminin K-fold sonucu.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
K-Fold	0.999158	0.776488	0.774412	0.759973	0.763610

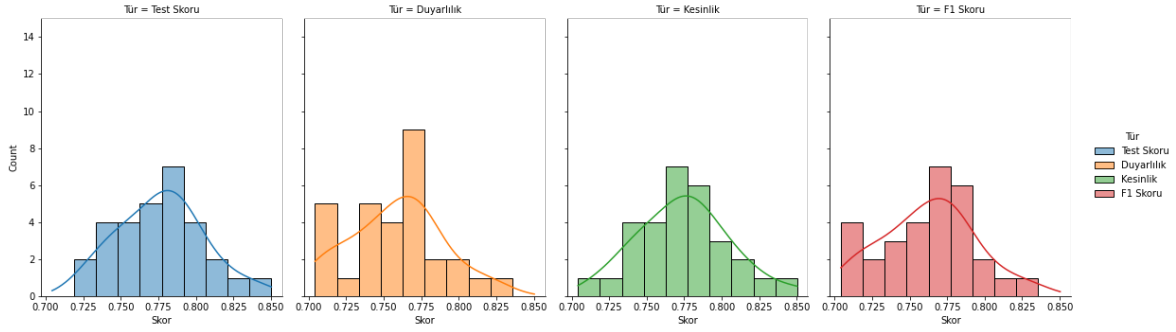
Şekil 3.6.6.1. incelendiğinde Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test, duyarlılık, kesinlik ve F1 skorunun minimum ve maksimum değerlerinin birbirine yakın seyrettiği görülmektedir. En düşük ortalama performans değeri 0,75 olarak duyarlılık skorunda gerçekleşmiştir.



Şekil 3.6.6.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen TF-IDF Özellik çıkartımı ve DT makine öğrenmesi modelinin 30 iterasyon sonuçlarının kutu Grafiği.

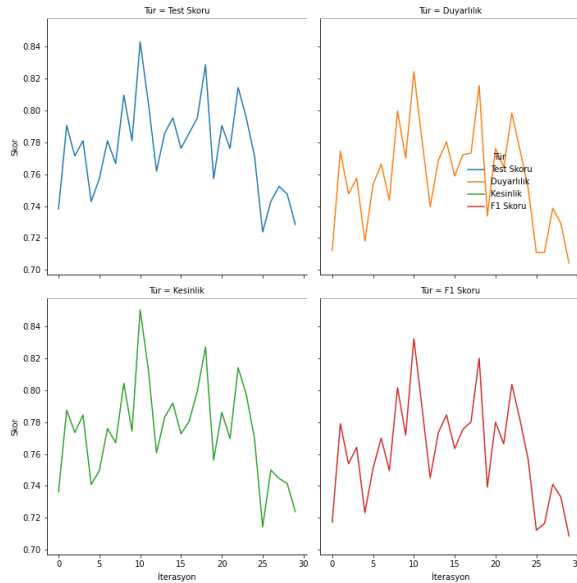
Şekil 3.6.6.2. üzerinde eğitim ve test verisi Test/Eğitim ayrımı yöntemi ile elde edilmiş TF-IDF Özellik çıkarımı ve DT makine öğrenmesi algoritması ile oluşturulmuş olan modelin

sonuçlarının histogram grafiği sunulmuştur. Test, kesinlik ve duyarlılık verileri Kolmogorov-Smirnov testine göre normal dağılmıştır.



Şekil 3.6.6.2. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş TF-IDF özellik çıkartımı ve DT makine öğrenmesi yönteminin 30 iterasyonunun sonuçlarının histogram grafiği.

Test/Eğitim ayrımı ile elde edilen eğitim ve test verileri ile eğitilen modelin 30 iterasyonda elde ettiği sonuçları Şekil 3.6.6.3. üzerinden incelendiğinde, 25. ve 4. İterasyon da gerçekleşen eğitim test verisi kombinasyonu dışındaki dağılımlarda pozitif ve negatif olarak 0,10 miktarında dalgalanma göze çarpmaktadır. Kesinlik Skoru ortalama 0,77 olarak gerçekleşmiş ve standart sapması 0,03 olarak gerçekleşmiştir. Duyarlılık skoru ortalama 0,75 olarak gerçekleşmiş ve standart sapması 0,03 olarak gerçekleşmiştir. F1 Skoru ortalama 0,76 olarak gerçekleşmiş ve standart sapması 0,05 olarak gerçekleşmiştir. Modelin duyarlılık, kesinlik ve F1 skoru arasında 0,03 sapma meydana gelmiştir.



Şekil 3.6.6.3. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş TF-IDF Özellik çıkartımı ve DT makine öğrenmesi modelinin 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.

3.6.7. TF-IDF ve SVM Modeli

TF-IDF özellik çıkartım yöntemi ile öznitelik vektörü haline dönüştürülmüş olan sahte ve gerçek haber metinleri SVM makine öğrenmesi algoritması eğitilerek sınıflandırılmıştır. Çizelge 3.6.7.1. üzerinde verilmiş olan değerler dikkate alındığında Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilmiş model 30 iterasyonda ortalama 0,97, maksimum 0,97 eğitim skoru elde edilmiştir.

Çizelge 3.6.7.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile TF-IDF Özellik çıkartımı ve SVM makine öğrenmesi yönteminin 30 iterasyon sonuçları.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
count	30.000000	30.000000	30.000000	30.000000	30.000000
mean	0.971907	0.782063	0.778522	0.781367	0.776326
std	0.002358	0.032650	0.030555	0.036483	0.036444
min	0.968085	0.714286	0.705556	0.675263	0.674122
25%	0.970449	0.754762	0.756953	0.760714	0.751985
50%	0.971631	0.788095	0.782224	0.789385	0.784116
75%	0.973700	0.809524	0.802778	0.806405	0.804236
max	0.977541	0.828571	0.822373	0.829426	0.823957

Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test skoru Çizelge 3.6.7.1. incelendiğinde ortalama eğitim değerinden 0,19 azalarak ortalama 0,78, standart sapması 0,03 olarak gerçekleşmiştir. Maksimum 0,82 test performans değeri elde edilmiştir.

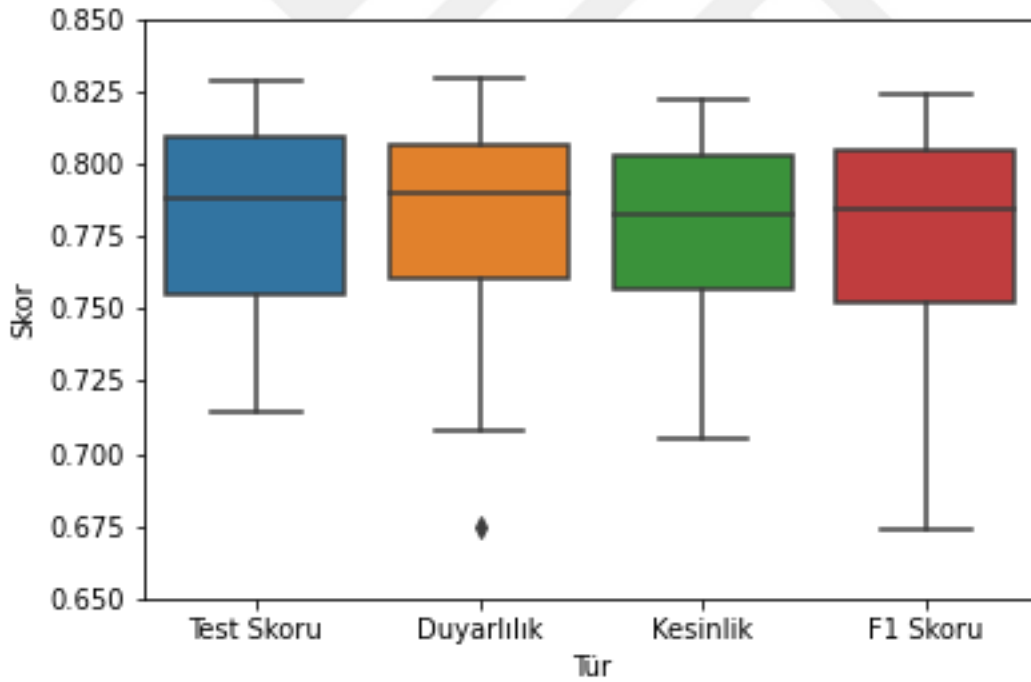
Model K-fold yöntemi ile TF-IDF özellik çıkarımı ve SVM makine öğrenmesi algoritması ile elde edilen modelin doğrulama gerçekleştirildiğinde elde edilen eğitim, test, kesinlik ve F1 skoru değerleri Çizelge 3.6.7.2. de gösterilmiştir. Test/Eğitim ayrımı yöntemi ile elde edilmiş TF-IDF SVM modelinin güvenilirliği K-fold yönteminin sonuçları ile desteklendiği

ve modelde örnekleme yanlılığının olmadığı görülmektedir.

Çizelge 3.6.7.2. TF-IDF Özellik çıkartımı ve SVM makine öğrenmesi yönteminin K-fold sonucu.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
K-Fold	0.973170	0.797458	0.799404	0.795252	0.790338

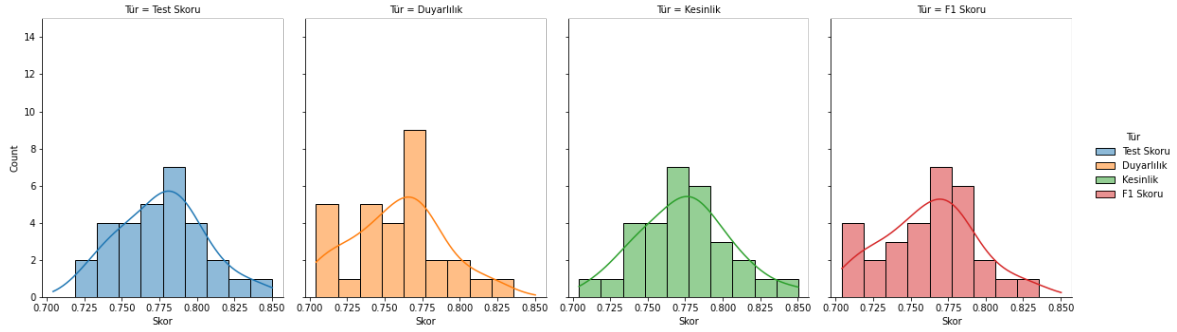
Şekil 3.6.7.1. incelendiğinde Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test, duyarlılık, kesinlik ve F1 skorunun minimum ve maksimum değerlerinin birbirine yakın seyrettiği görülmektedir. En düşük ortalama performans değeri 0,77 olarak duyarlılık skorunda gerçekleşmiştir. Duyarlılık skorunda aykırı bir değer olduğu görülmektedir.



Şekil 3.6.7.2. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen TF-IDF Özellik çıkartımı ve SVM makine öğrenmesi modelinin 30 iterasyon sonuçlarının kutu grafiği.

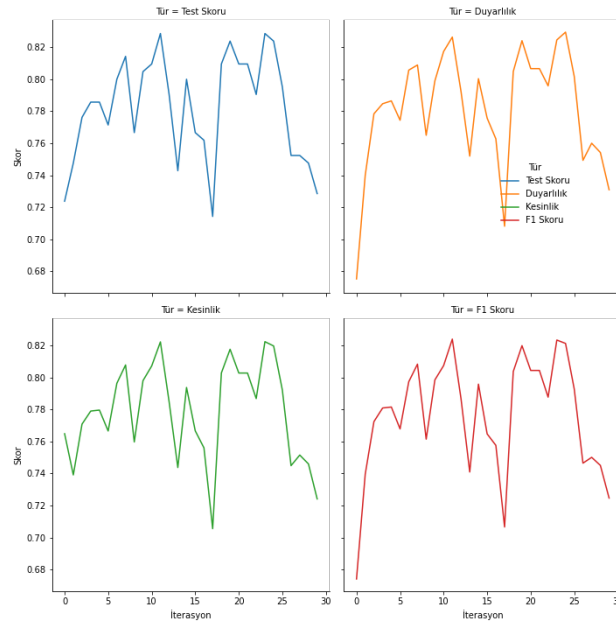
Şekil 3.6.7.2. üzerinde eğitim ve test verisi Test/Eğitim ayrımı yöntemi ile elde edilmiş TF-IDF Özellik çıkartımı ve SVM makine öğrenmesi algoritması ile oluşturulmuş olan modelin

sonuçlarının histogram grafiği sunulmuştur. Test, kesinlik ve duyarlılık verileri Kolmogorov-Smirnov testine göre normal dağılmıştır.



Şekil 3.6.7.3. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş TF-IDF özellik çıkartımı ve SVM makine öğrenmesi yönteminin 30 iterasyonun sonuçlarının histogram grafiği.

Test/Eğitim ayrımı ile elde edilen eğitim ve test verileri ile eğitilen modelin 30 iterasyonda elde ettiği sonuçları Şekil 3.6.7.3. üzerinden incelendiğinde. 17. İterasyon da gerçekleşen eğitim test verisi kombinasyonu dışındaki dağılımlarda pozitif ve negatif olarak 0,06 miktarında dalgalanma göze çarpmaktadır. Kesinlik Skoru ortalama 0,77 olarak gerçekleşmiş ve standart sapması 0,03 olarak gerçekleşmiştir. Duyarlılık skoru ortalama 0,78 olarak gerçekleşmiş ve standart sapması 0,03 olarak gerçekleşmiştir. F1 Skoru ortalama 0,77 olarak gerçekleşmiş ve standart sapması 0,03 olarak gerçekleşmiştir. Modelin duyarlılık, kesinlik ve F1 skoru arasında 0,03 sapma meydana gelmiştir.



Şekil 3.6.7.3. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş TF-IDF Özellik çıkartımı ve SVM makine öğrenmesi modelinin 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.

3.6.8. TF-IDF ve LGR Modeli

TF-IDF özellik çıkartım yöntemi ile öznitelik vektörü haline dönüştürülmüş olan sahte ve gerçek haber metinleri LGR makine öğrenmesi algoritması eğitilerek sınıflandırılmıştır. Çizelge 4.8.1. üzerinde verilmiş olan değerler dikkate alındığında Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilmiş model 30 iterasyonda ortalama 0,99, maksimum 1,00 eğitim skoru elde edilmiştir.

Çizelge 3.6.8.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile TF-IDF Özellik çıkartımı ve LGR makine öğrenmesi yönteminin 30 iterasyon sonuçları.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
count	30.000000	30.000000	30.000000	30.000000	30.000000
mean	0.999133	0.811905	0.826916	0.785850	0.794814
std	0.000532	0.023253	0.023797	0.026449	0.026897
min	0.998818	0.766667	0.780952	0.734715	0.741944
25%	0.998818	0.792857	0.808910	0.767137	0.775106
50%	0.998818	0.809524	0.827348	0.782586	0.791964
75%	0.999704	0.827381	0.842481	0.804389	0.813520
max	1.000000	0.866667	0.878571	0.847899	0.857226

Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test skoru Çizelge 3.6.8.1. incelendiğinde ortalama eğitim değerinden 0,18 azalarak ortalama 0,81, standart sapması 0,02 olarak gerçekleşmiştir. Maksimum 0,86 test performans değeri elde edilmiştir.

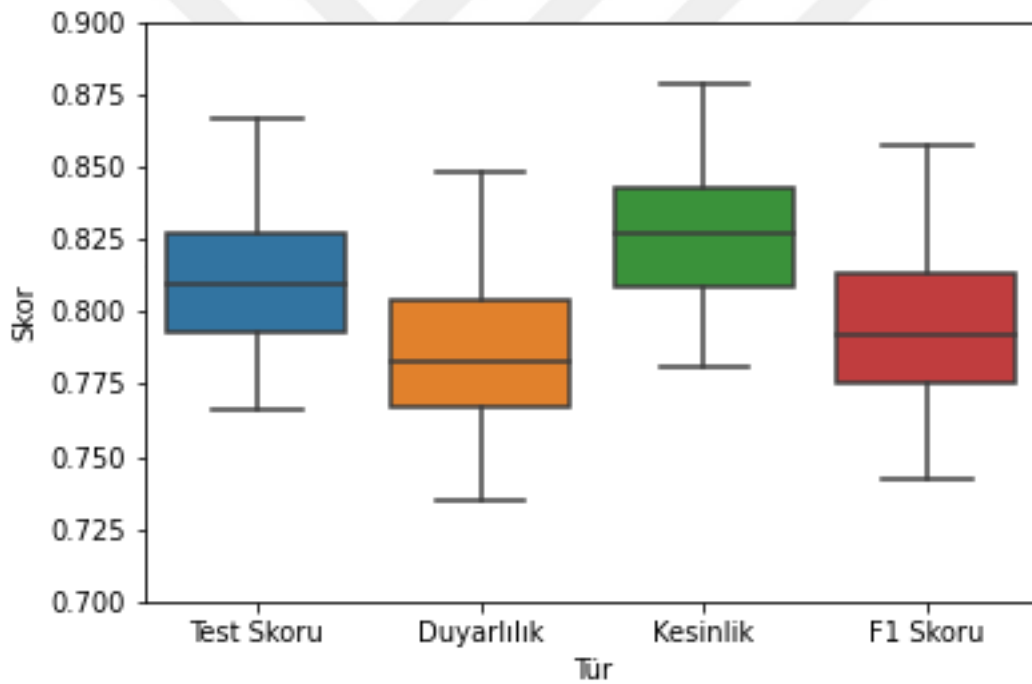
Model K-fold yöntemi ile TF-IDF özellik çıkarımı ve LGR makine öğrenmesi algoritması ile elde edilen modelin doğrulama gerçekleştirildiğinde elde edilen eğitim, test, kesinlik ve F1 skoru değerleri Çizelge 3.6.8.2. de gösterilmiştir. Test/Eğitim ayrımı yöntemi ile elde edilmiş TF-IDF LGR modelinin güvenilirliği K-fold yönteminin sonuçları ile desteklendiği

ve modelde örnekleme yanlılığının olmadığı görülmektedir.

Çizelge 3.6.8.2. TF-IDF Özellik çıkartımı ve LGR makine öğrenmesi yönteminin K-fold sonucu.

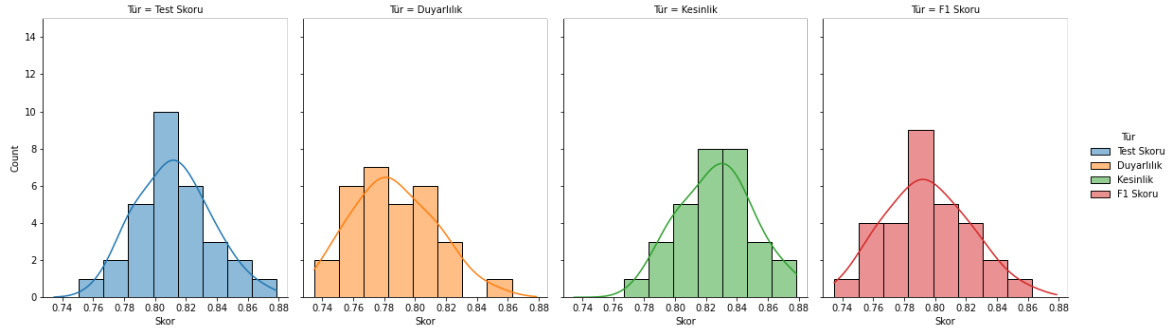
	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
K-Fold	0.999158	0.818089	0.833824	0.793076	0.801793

Şekil 3.6.8.1. incelendiğinde Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test, duyarlılık, kesinlik ve F1 skorunun ortalama değeri 0,77 üzerinde gerçekleşmiştir. En düşük ortalama performans değeri 0,78 olarak duyarlılık skorunda gerçekleşmiştir. Kesinlik performans değeri ortalama 0,82 olarak gerçekleşmiştir.



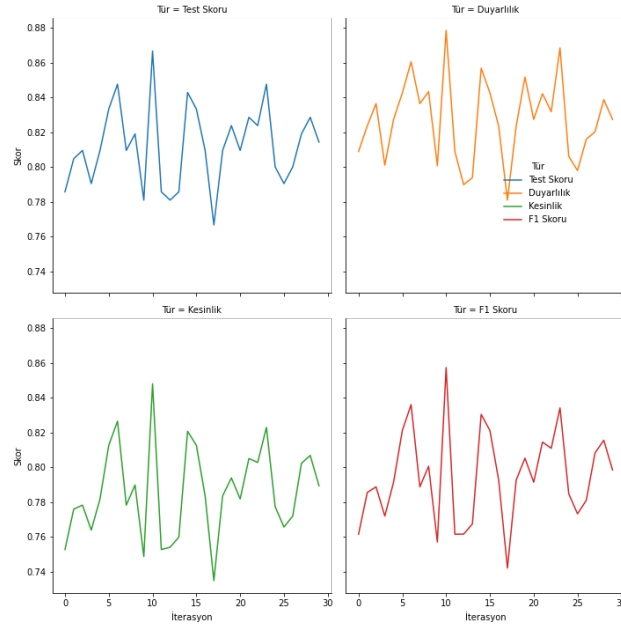
Şekil 3.6.8.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen TF-IDF Özellik çıkartımı ve LGR makine öğrenmesi modelinin 30 iterasyon sonuçlarının kutu grafiği.

Şekil 3.6.8.2. üzerinde eğitim ve test verisi Test/Eğitim ayrımı yöntemi ile elde edilmiş TF-IDF Özellik çıkarımı ve LGR makine öğrenmesi algoritması ile oluşturulmuş olan modelin sonuçlarının histogram grafiği sunulmuştur. Test, kesinlik ve duyarlılık verileri Kolmogorov-Smirnov testine göre normal dağılmıştır.



Şekil 3.6.8.2. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş TF-IDF özellik çıkartımı ve LGR makine öğrenmesi yönteminin 30 iterasyonun sonuçlarının histogram grafiği.

Test/Eğitim ayrımı ile elde edilen eğitim ve test verileri ile eğitilen modelin 30 iterasyonda elde ettiği sonuçları Şekil 3.6.8.3. üzerinden incelendiğinde, 17. İterasyon da gerçekleşen eğitim test verisi kombinasyonu dışındaki dağılımlarda pozitif ve negatif olarak 0,10 miktarında dalgalanma göze çarpmaktadır. Duyarlılık Skoru ortalama 0,82 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. Kesinlik skoru ortalama 0,78 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. F1 Skoru ortalama 0,79 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. Modelin ortalama duyarlılık skoru kesinlik skorundan daha başarılı sonuç elde etmiştir.



Şekil 3.6.8.3. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş TF-IDF Özellik çıkartımı ve LGR makine öğrenmesi modelinin 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.

3.6.9. Doc2Vec ve MLP Modeli

Doc2Vec özellik çıkartım yöntemi ile öznitelik vektörü haline dönüştürülmüş olan sahte ve gerçek haber metinleri MLP makine öğrenmesi algoritması eğitilerek sınıflandırılmıştır. Çizelge 3.6.9.1. üzerinde verilmiş olan değerler dikkate alındığında Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilmiş model 30 iterasyonda ortalama 0,80, maksimum 0,96 eğitim skoru elde edilmiştir.

Çizelge 3.6.9.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile Doc2Vec Özellik çıkartımı ve MLP makine öğrenmesi yönteminin 30 iterasyon sonuçları.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
count	30.000000	30.000000	30.000000	30.000000	30.000000
mean	0.803073	0.711270	0.611208	0.671474	0.630090
std	0.155168	0.089686	0.228467	0.125740	0.187725
min	0.591017	0.590476	0.295238	0.500000	0.371257
25%	0.591017	0.590476	0.295238	0.500000	0.371257
50%	0.889480	0.747619	0.744759	0.737294	0.738491
75%	0.910165	0.779762	0.773065	0.765402	0.768702
max	0.969267	0.828571	0.826923	0.815641	0.819931

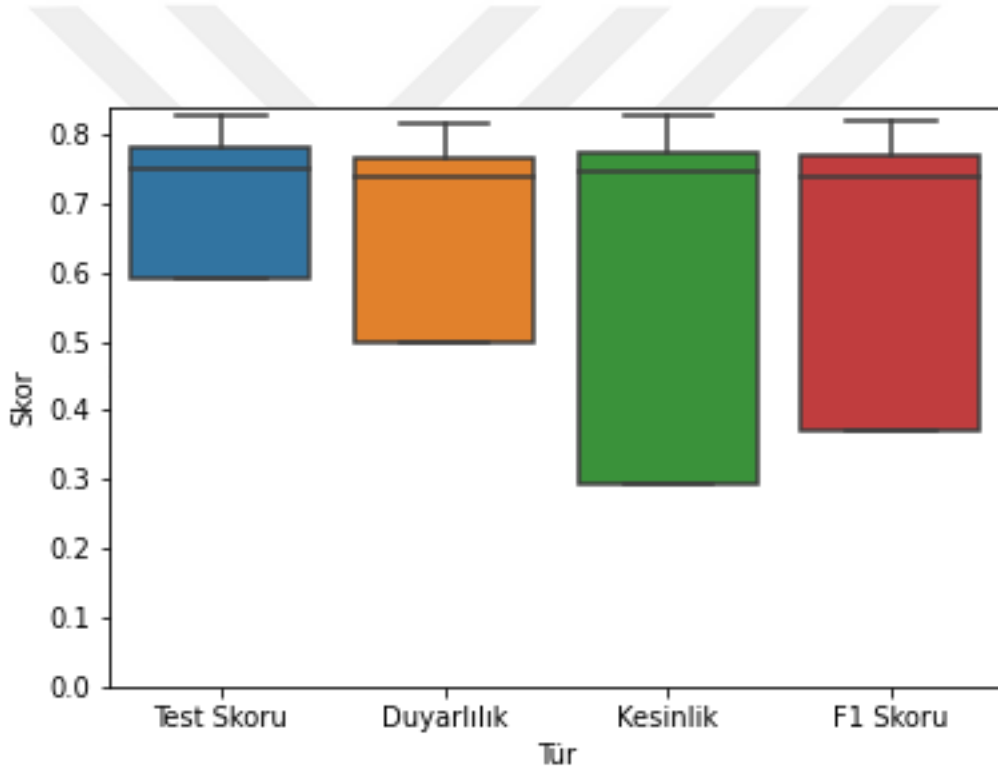
Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test skoru Çizelge 3.6.9.1. incelendiğinde ortalama eğitim değerinden 0,09 azalarak ortalama 0,71, standart sapması 0,08 olarak gerçekleşmiştir. Maksimum 0,82 test performans değeri elde edilmiştir.

Model K-fold yöntemi ile Doc2Vec özellik çıkarımı ve MLP makine öğrenmesi algoritması ile elde edilen modelin doğrulama gerçekleştirildiğinde elde edilen eğitim, test, kesinlik ve F1 skoru değerleri Çizelge 3.6.9.2. de gösterilmiştir. Test/Eğitim ayrımı yöntemi ile elde edilmiş Doc2Vec MLP modelinin güvenilirliği K-fold yönteminin sonuçları ile desteklendiği ve modelde örnekleme yanlılığının olmadığı görülmektedir.

Çizelge 3.6.9.2. Doc2Vec Özellik çıkartımı ve MLP makine öğrenmesi yönteminin K-fold sonucu.

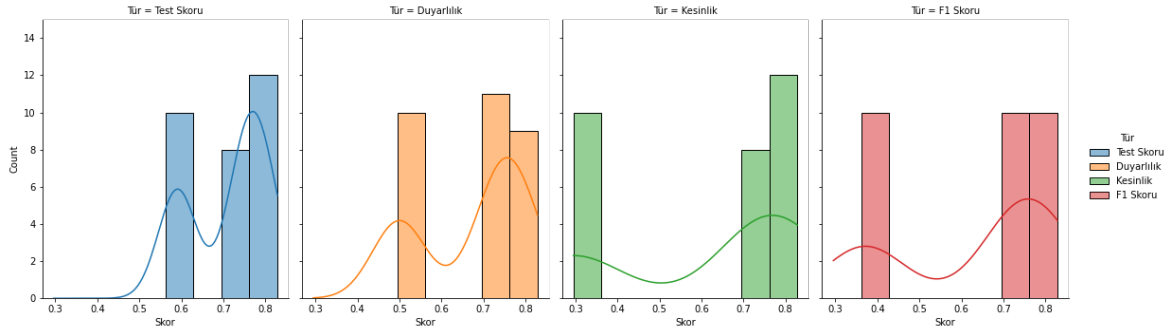
	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
K-Fold	0.732165	0.665008	0.485265	0.608681	0.530705

Şekil 3.6.9.1. incelendiğinde Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test, duyarlılık, kesinlik ve F1 skorunun ortalama değeri alt sınır ve aşırı sınır çizgilerinin kesiştiği ve anormal bir durum oluşturduğu görülmektedir. En düşük ortalama performans değeri 0,61 olarak kesinlik değerinde gerçekleşmiştir.



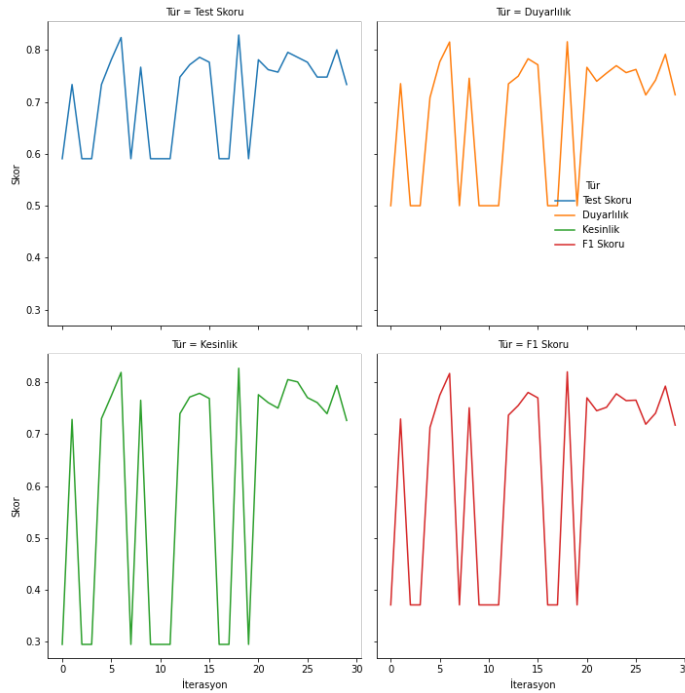
Şekil 3.6.9.1 Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen Doc2Vec Özellik çıkartımı ve MLP makine öğrenmesi modelinin 30 iterasyon sonuçlarının kutu grafiği.

Şekil 3.6.9.2. üzerinde eğitim ve test verisi Test/Eğitim ayrımı yöntemi ile elde edilmiş Doc2Vec Özellik çıkarımı ve MLP makine öğrenmesi algoritması ile oluşturulmuş olan modelin sonuçlarının histogram grafiği sunulmuştur. Test, kesinlik ve duyarlılık verileri Kolmogorov-Smirnov testine göre normal dağılmıştır.



Şekil 3.6.9.2. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş Doc2Vec özellik çıkartımı ve MLP makine öğrenmesi yönteminin 30 iterasyonun sonuçlarının histogram grafiği.

Test/Eğitim ayrımı ile elde edilen eğitim ve test verileri ile eğitilen modelin 30 iterasyonda elde ettiği sonuçları Şekil 3.6.9.3. üzerinden incelendiğinde, 17. İterasyon da gerçekleşen eğitim test verisi kombinasyonu dışındaki dağılımlarda pozitif ve negatif olarak 0,35 miktarında dalgalanma göze çarpmaktadır. Modelin birçok veri seti dağılımında öğrenme gerçekleştiremediği görülmektedir. Kesinlik Skoru ortalama 0,61 olarak gerçekleşmiş ve standart sapması 0,22 olarak gerçekleşmiştir. Duyarlılık skoru ortalama 0,67 olarak gerçekleşmiş ve standart sapması 0,12 olarak gerçekleşmiştir. F1 Skoru ortalama 0,63 olarak gerçekleşmiş ve standart sapması 0,18 olarak gerçekleşmiştir.



Şekil 3.6.9.3. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş Doc2Vec Özellik çıkartımı ve MLP makine öğrenmesi modelinin 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.

3.6.10. Doc2Vec ve DT Modeli

Doc2Vec özellik çıkartım yöntemi ile öznitelik vektörü haline dönüştürülmüş olan sahte ve gerçek haber metinleri DT makine öğrenmesi algoritması eğitilerek sınıflandırılmıştır. Çizelge 3.6.10.1. üzerinde verilmiş olan değerler dikkate alındığında Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilmiş model 30 iterasyonda ortalama 1,00 maksimum 1,00 eğitim skoru elde edilmiştir.

Çizelge 3.6.10.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile Doc2Vec Özellik çıkartımı ve DT makine öğrenmesi yönteminin 30 iterasyon sonuçları.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
count	30.0	30.000000	30.000000	30.000000	30.000000
mean	1.0	0.707460	0.698865	0.698478	0.697791
std	0.0	0.033711	0.034742	0.034137	0.034234
min	1.0	0.633333	0.624621	0.627157	0.625165
25%	1.0	0.683333	0.673577	0.677279	0.674286
50%	1.0	0.707143	0.700084	0.705458	0.700011
75%	1.0	0.732143	0.723033	0.721962	0.722439
max	1.0	0.771429	0.764278	0.769036	0.765995

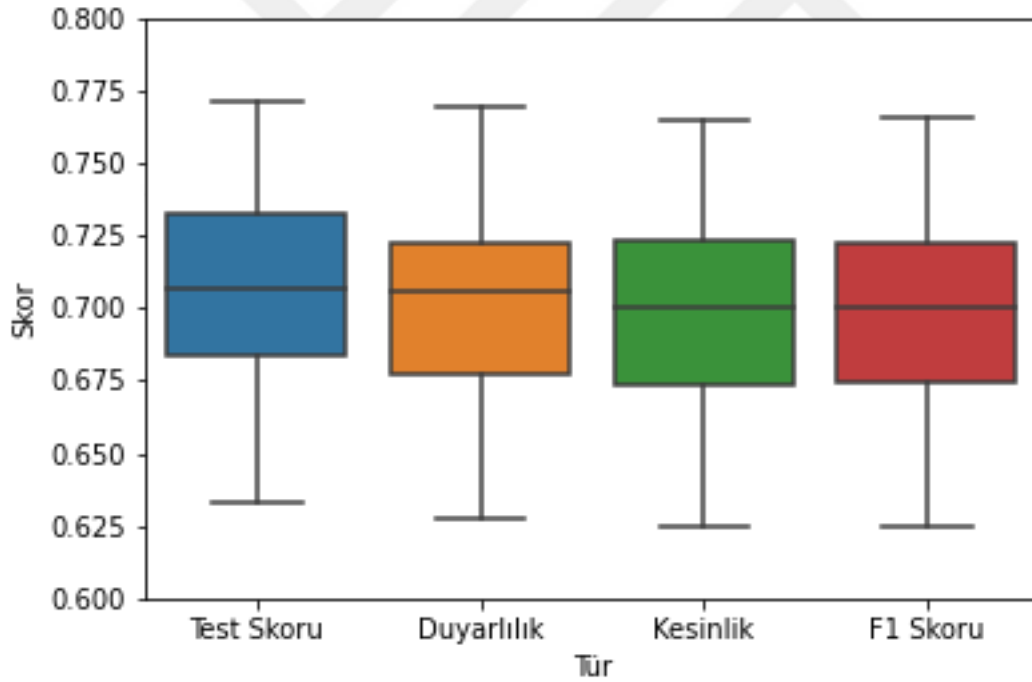
Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test skoru Çizelge 3.6.10.1. incelendiğinde ortalama eğitim değerinden 0,30 azalarak ortalama 0,70, standart sapması 0,03 olarak gerçekleşmiştir. Maksimum 0,77 test performans değeri elde edilmiştir.

Model K-fold yöntemi ile Doc2Vec özellik çıkarımı ve DT makine öğrenmesi algoritması ile elde edilen modelin doğrulama gerçekleştirildiğinde elde edilen eğitim, test, kesinlik ve F1 skoru değerleri Çizelge 3.6.10.2. de gösterilmiştir. Test/Eğitim ayrımı yöntemi ile elde edilmiş Doc2Vec DT modelinin güvenilirliği K-fold yönteminin sonuçları ile desteklendiği ve modelde örnekleme yanlılığının olmadığı görülmektedir.

Çizelge 3.6.10.2. Doc2Vec Özellik çıkartımı ve DT makine öğrenmesi yönteminin K-fold sonucu.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
K-Fold	1.000000	0.713979	0.705153	0.703189	0.701976

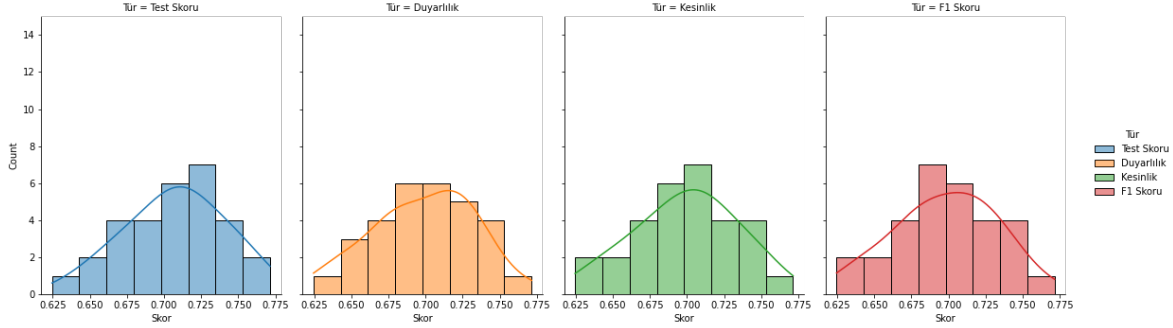
Şekil 3.6.10.1. incelendiğinde Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test, duyarlılık, kesinlik ve F1 skorunun minimum ve maksimum değerlerinin birbirine yakın seyrettiği görülmektedir. En düşük ortalama performans değeri 0,69 olarak duyarlılık skorunda gerçekleşmiştir.



Şekil 3.6.10.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen Doc2Vec Özellik çıkartımı ve DT makine öğrenmesi modelinin 30 iterasyon sonuçlarının kutu grafiği.

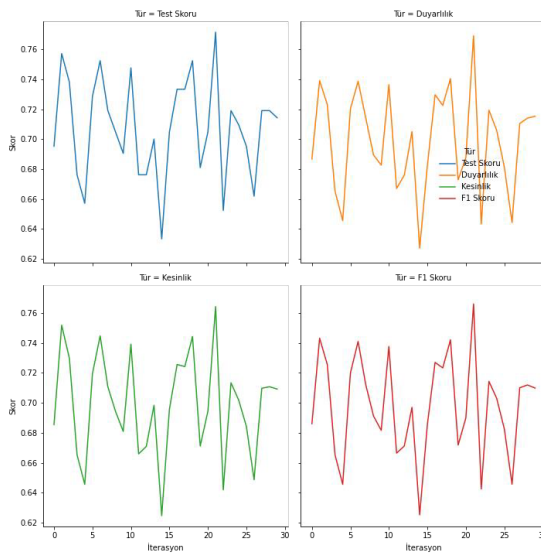
Şekil 3.6.10.2. üzerinde eğitim ve test verisi Test/Eğitim ayrımı yöntemi ile elde edilmiş Doc2Vec Özellik çıkarımı ve DT makine öğrenmesi algoritması ile oluşturulmuş olan

modelin sonuçlarının histogram grafiği sunulmuştur. Test, kesinlik ve duyarlılık verileri Kolmogorov-Smirnov testine göre normal dağılmıştır.



Şekil 3.6.10.2. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş Doc2Vec özellik çıkartımı ve DT makine öğrenmesi yönteminin 30 iterasyonun sonuçlarının histogram grafiği.

Test/Eğitim ayrımı ile elde edilen eğitim ve test verileri ile eğitilen modelin 30 iterasyonda elde ettiği sonuçları Şekil 3.6.10.3. üzerinden incelendiğinde. 14. İterasyon da gerçekleşen eğitim test verisi kombinasyonu dışındaki dağılımlarda pozitif ve negatif olarak 0,10 miktarında dalgalanma göze çarpmaktadır. Kesinlik Skoru ortalama 0,69 olarak gerçekleşmiş ve standart sapması 0,03 olarak gerçekleşmiştir. Duyarlılık skoru ortalama 0,69 olarak gerçekleşmiş ve standart sapması 0,03 olarak gerçekleşmiştir. F1 Skoru ortalama 0,69 olarak gerçekleşmiş ve standart sapması 0,03 olarak gerçekleşmiştir. Modelin ortalama duyarlılık, kesinlik ve F1 skoru eşit gerçekleşmiştir.



Şekil 3.6.10.3 Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş Doc2Vec Özellik çıkartımı ve MLP makine öğrenmesi modelinin 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.

3.6.11. Doc2Vec ve SVM Modeli

Doc2Vec özellik çıkartım yöntemi ile öznitelik vektörü haline dönüştürülmüş olan sahte ve gerçek haber metinleri SVM makine öğrenmesi algoritması eğitilerek sınıflandırılmıştır. Çizelge 3.6.11.1. üzerinde verilmiş olan değerler dikkate alındığında Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilmiş model 30 iterasyonda ortalama 0,89 maksimum 0,90 eğitim skoru elde edilmiştir.

Çizelge 3.6.11.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile Doc2Vec Özellik çıkartımı ve SVM makine öğrenmesi yönteminin 30 iterasyon sonuçları.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
count	30.000000	30.000000	30.000000	30.000000	30.000000
mean	0.890189	0.796667	0.809966	0.769917	0.778150
std	0.006014	0.023064	0.027671	0.024933	0.025828
min	0.875887	0.738095	0.738196	0.710521	0.715650
25%	0.886820	0.782143	0.794642	0.756728	0.763985
50%	0.891253	0.800000	0.810310	0.771380	0.779237
75%	0.894799	0.813095	0.833179	0.783032	0.792397
max	0.900709	0.833333	0.850733	0.814329	0.821995

Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test skoru Çizelge 3.6.11.1. incelendiğinde ortalama eğitim değerinden 0,10 azalarak ortalama 0,79, standart sapması 0,02 olarak gerçekleşmiştir. Maksimum 0,83 test performans değeri elde edilmiştir.

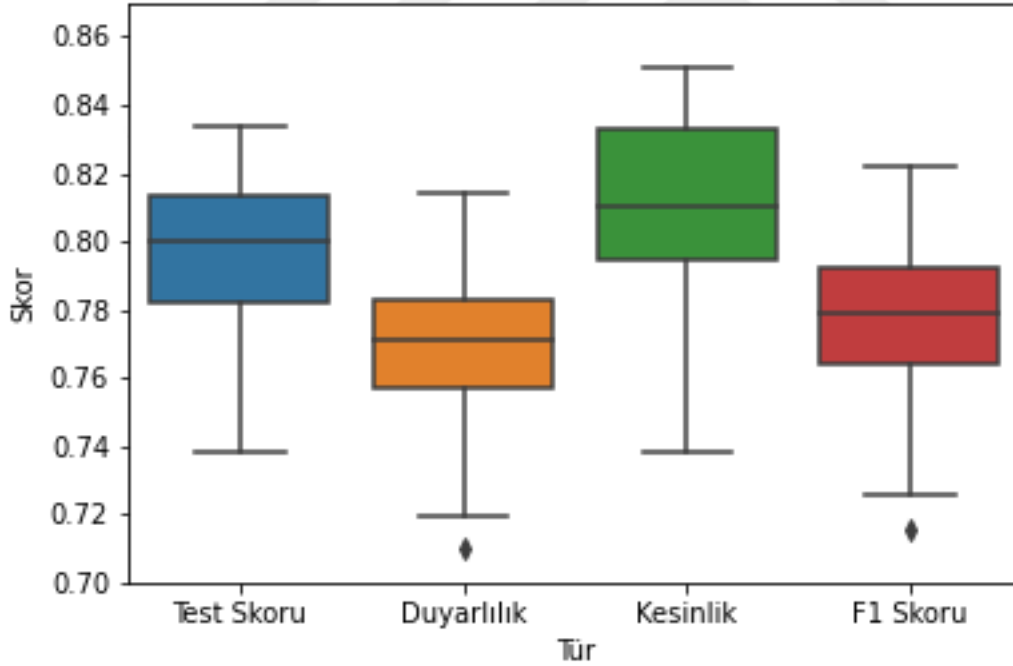
Model K-fold yöntemi ile Doc2Vec özellik çıkarımı ve SVM makine öğrenmesi algoritması ile elde edilen modelin doğrulama gerçekleştirildiğinde elde edilen eğitim, test, kesinlik ve F1 skoru değerleri Çizelge 3.6.11.2. de gösterilmiştir. Test/Eğitim ayrımı yöntemi ile elde edilmiş Doc2Vec SVM modelinin güvenilirliği K-fold yönteminin sonuçları ile

desteklendiği ve modelde örnekleme yanlılığının olmadığı görülmektedir.

Çizelge 3.6.11.2. Doc2Vec Özellik çıkartımı ve SVM makine öğrenmesi yönteminin K-fold sonucu.

	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
K-Fold	0.898464	0.801970	0.809675	0.779025	0.786527

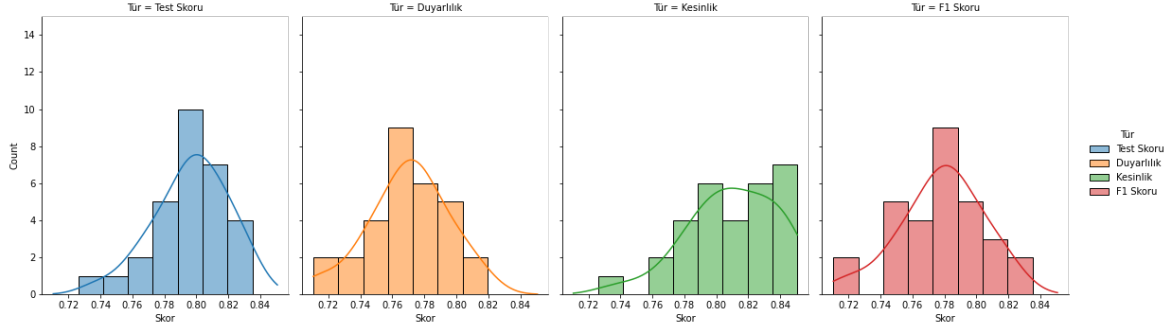
Şekil 3.6.11.1. incelendiğinde Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test, duyarlılık, kesinlik ve F1 skorunun minimum ve maksimum değerlerinin birbirine yakın seyrettiği görülmektedir. Modelin kesinlik değeri duyarlılık değerinden daha iyi sonuç vermiştir. En düşük ortalama performans değeri 0,76 olarak kesinlik skorunda gerçekleşmiştir.



Şekil 3.6.11.1. Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen Doc2Vec Özellik çıkartımı ve SVM makine öğrenmesi modelinin 30 iterasyon sonuçlarının kutu grafiği.

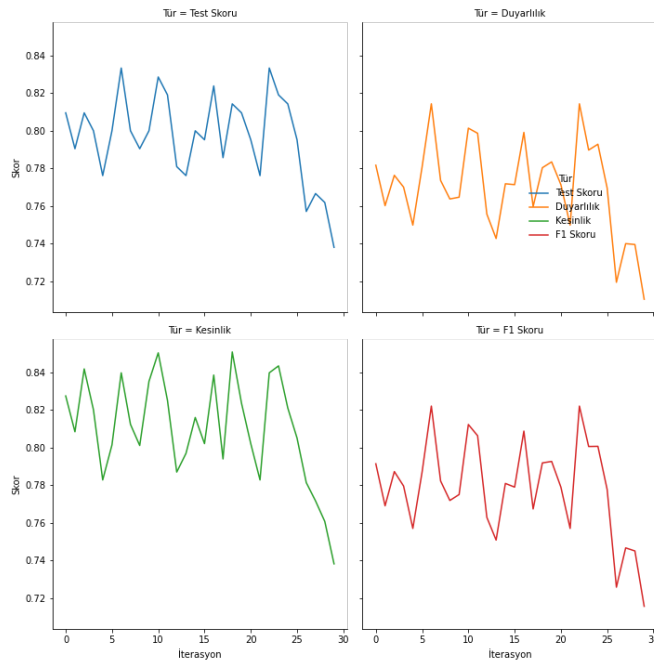
Şekil 3.6.11.2. üzerinde eğitim ve test verisi Test/Eğitim ayrımı yöntemi ile elde edilmiş Doc2Vec Özellik çıkarımı ve SVM makine öğrenmesi algoritması ile oluşturulmuş olan

modelin sonuçlarının histogram grafiği sunulmuştur. Test, kesinlik ve duyarlılık verileri Kolmogorov-Smirnov testine göre normal dağılmıştır.



Şekil 3.6.11.2. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş Doc2Vec özellik çıkartımı ve SVM makine öğrenmesi yönteminin 30 iterasyonun sonuçlarının histogram grafiği.

Test/Eğitim ayrımı ile elde edilen eğitim ve test verileri ile eğitilen modelin 30 iterasyonda elde ettiği sonuçları Şekil 3.6.11.3. üzerinden incelendiğinde. 26. İterasyon da gerçekleşen eğitim test verisi kombinasyonu dışındaki dağılımlarda pozitif ve negatif olarak 0,08 miktarda dalgalanma göze çarpmaktadır. Kesinlik Skoru ortalama 0,80 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. Duyarlılık skoru ortalama 0,76 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. F1 Skoru ortalama 0,77 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. Modelin ortalama kesinlik değeri, duyarlılık değerinden yüksek gerçekleşmiştir.



Test/Eđitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş Doc2Vec Özellik çıkartımı ve SVM makine öğrenmesi modelinin 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiđi.

3.6.12. Doc2Vec ve LGR Modeli

Doc2Vec özellik çıkartım yöntemi ile öznitelik vektörü haline dönüştürülmüş olan sahte ve gerçek haber metinleri LGR makine öğrenmesi algoritması eğitilerek sınıflandırılmıştır. Çizelge 3.6.12.1. üzerinde verilmiş olan değerler dikkate alındığında Test/Eđitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilmiş model 30 iterasyonda ortalama 0,83 maksimum 0,85 eğitim skoru elde edilmiştir.

Çizelge 3.6.12.1. Test/Eđitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile Doc2Vec Özellik çıkartımı ve LGR makine öğrenmesi yönteminin 30 iterasyon sonuçları.

	Eđitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
count	30.000000	30.000000	30.000000	30.000000	30.000000
mean	0.839125	0.797460	0.798510	0.780151	0.784900
std	0.010017	0.022845	0.025168	0.025862	0.025707
min	0.815603	0.738095	0.745594	0.705176	0.710345
25%	0.833333	0.786905	0.782115	0.770560	0.775748
50%	0.836879	0.800000	0.802024	0.785259	0.790820
75%	0.846040	0.809524	0.810763	0.793605	0.799462
max	0.856974	0.847619	0.851550	0.831770	0.838462

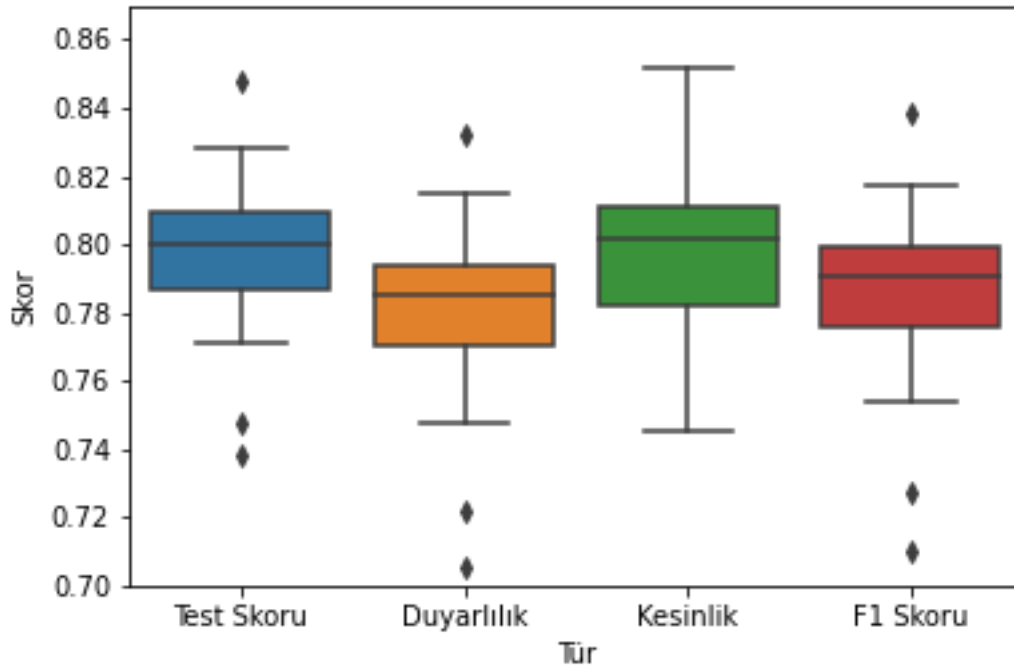
Test/Eđitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test skoru Çizelge 3.6.12.1. incelendiđinde ortalama eğitim değerinden 0,04 azalarak ortalama 0,79, standart sapması 0,02 olarak gerçekleşmiştir. Maksimum 0,84 test performans değeri elde edilmiştir.

Model K-fold yöntemi ile Doc2Vec özellik çıkarımı ve LGR makine öğrenmesi algoritması ile elde edilen modelin doğrulama gerçekleştirildiğinde elde edilen eğitim, test, kesinlik ve F1 skoru değerleri Çizelge 3.6.12.2. de gösterilmiştir. Test/Eğitim ayrımı yöntemi ile elde edilmiş Doc2Vec LGR modelinin güvenilirliği K-fold yönteminin sonuçları ile desteklendiği ve modelde örnekleme yanlılığının olmadığı görülmektedir.

Çizelge 3.6.12.2. Doc2Vec Özellik çıkartımı ve LGR makine öğrenmesi yönteminin K-fold sonucu.

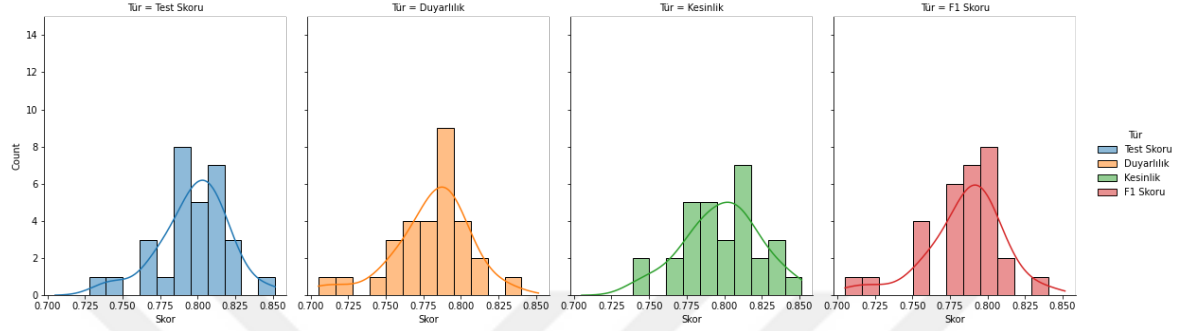
	Eğitim Skoru	Test Skoru	Kesinlik	Duyarlılık	F1 Skoru
K-Fold	0.838279	0.799184	0.799876	0.781315	0.786554

Şekil 3.6.12.1. incelendiğinde Test/Eğitim ayrımı yöntemi ile elde edilen eğitim ve test verisi ile eğitilen modelin test, duyarlılık, kesinlik ve F1 skorunun minimum ve maksimum değerlerinin birbirine yakın seyrettiği görülmektedir. Modelin kesinlik değeri duyarlılık değerinden daha iyi sonuç vermiştir. En düşük ortalama performans değeri 0,78 olarak duyarlılık skorunda gerçekleşmiştir.



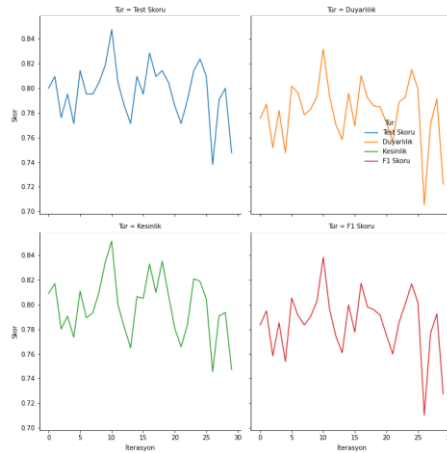
Şekil 3.6.12.1. Doc2Vec Özellik çıkartımı ve LGR makine öğrenmesi yöntemi 30 iterasyon sonuçlarının kutu grafiği.

Şekil 3.6.12.2. üzerinde eğitim ve test verisi Test/Eğitim ayrımı yöntemi ile elde edilmiş Doc2Vec Özellik çıkarımı ve LGR makine öğrenmesi algoritması ile oluşturulmuş olan modelin sonuçlarının histogram grafiği sunulmuştur. Test, kesinlik ve duyarlılık verileri Kolmogorov-Smirnov testine göre normal dağılmıştır.



Şekil 3.6.12.2. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş Doc2Vec özellik çıkartımı ve LGR makine öğrenmesi yönteminin 30 iterasyonun sonuçlarının histogram grafiği.

Test/Eğitim ayrımı ile elde edilen eğitim ve test verileri ile eğitilen modelin 30 iterasyonda elde ettiği sonuçları Şekil 3.6.12.3. üzerinden incelendiğinde. 26. İterasyon da gerçekleşen eğitim test verisi kombinasyonu dışındaki dağılımlarda pozitif ve negatif olarak 0,08 miktarında dalgalanma göze çarpmaktadır. Kesinlik Skoru ortalama 0,79 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. Duyarlılık skoru ortalama 0,78 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. F1 Skoru ortalama 0,78 olarak gerçekleşmiş ve standart sapması 0,02 olarak gerçekleşmiştir. Modelin ortalama kesinlik değeri, duyarlılık değerinden yüksek gerçekleşmiştir.

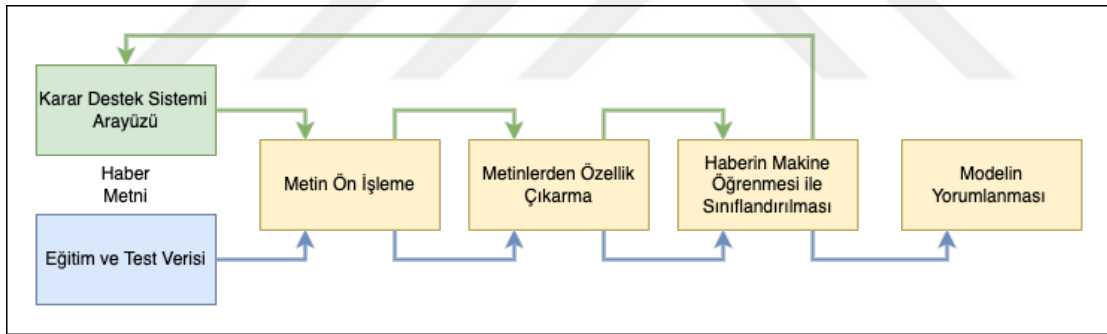


Şekil 3.6.12.3. Test/Eğitim ayrımı yöntemi ile eğitim ve test verileri elde edilmiş Doc2Vec Özellik çıkartımı ve LGR makine öğrenmesi modelinin 30 iterasyon sonuçlarının her iterasyonda gerçekleşen skor grafiği.

4. KARAR DESTEK SİSTEMİNİN TASARLANMASI

Dezenformasyonun zararlı etkilerinin ve dezenformasyondan etkilenen kişi sayısının azaltılmasında hızlı karar alabilmek için 3. Bölümünde sahte haber tespiti konusunda en iyi performans değerlerine sahip model ile karar destek sistemi tasarlanması amaçlanmıştır. Tasarlanan karar destek sisteminin farklı cihazlarda kullanılabilmesine imkân sağlamak için mobil, tablet ve masaüstü uyumlu bir web ara yüz tasarlanmıştır. Ayrıca uygulama işlem ara yüzü (API) geliştirilerek tasarlanan karar destek sisteminin program geliştiricilerin de kullanımına açılması hedeflenmiştir.

Karar destek sisteminde çalışmanın 3. Bölümünde en iyi F1, duyarlılık, keskinlik ve test skorunu elde eden TF-IDF ve LGR modeli sistemde sınıflandırıcı olarak kullanılmıştır. Ayrıca karar vericiye çalışmada elde edilen diğer modelleri de kullanma imkânı sunulmaktadır. Şekil 4.1. üzerinde karar destek sisteminin ara yüzü üzerinden makine öğrenimi algoritmasının kullanım adımları gösterilmiştir.



Şekil 4.1. Karar destek sisteminin makine öğrenmesi algoritması ile kullanımı.

Çalışmanın 3. Bölümünde elde edilen modelin karar destek sisteminde kullanılabilmesi için Python programa dilinin bir kütüphanesi olan Pickle kütüphanesinden yararlanılmıştır. Bu kütüphane geçici bellekte depolanan makine öğrenmesi modellerini ve bu modelin kullanılabilmesi için gerekli olan diğer sınıf ve değişkenleri kalıcı bellekte saklayıp, ihtiyaç halinde tekrar geçici bellek üzerine yüklenmesini ve kayıpsız şekilde kullanılmasını sağlamaktadır. Hem karar destek sisteminin Python dilinde hem de karar destek sisteminde kullanılacak olan makine öğrenmesi modelinin aynı dille gerçekleştirilmiş olması teknik karmaşıklığı azaltmıştır.

Karar destek sisteminin geliştirilmesinde Flask isimli web kütüphanesi kullanılmıştır. Geliştirilen karar destek sistemi www.dogrulukkontrolu.com.tr web adresi üzerinden açık erişime sunulmuştur. Ayrıca karar destek sisteminin kodları açık kaynak olarak GitHub üzerinden yayınlanmıştır.

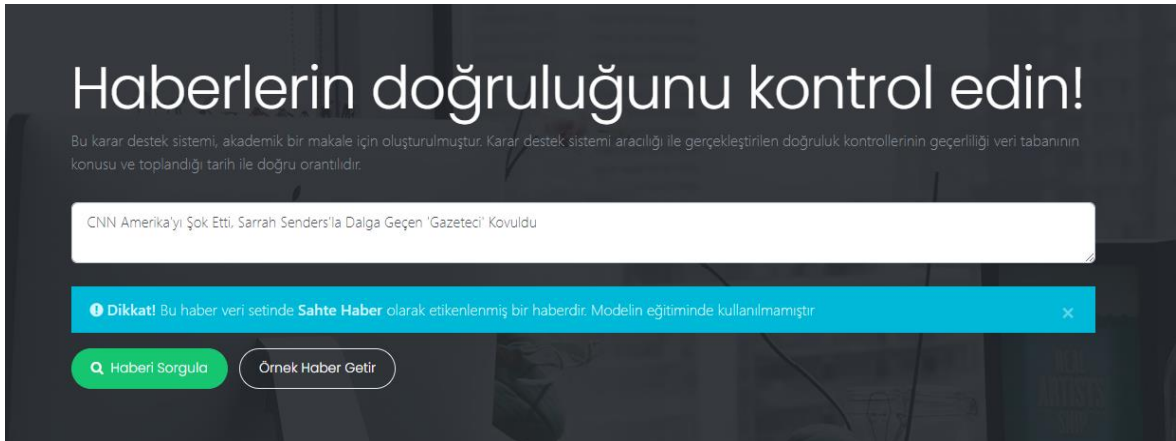
Karar verici karar destek sisteminde ilk olarak Resim 4.1. de gösterilen ara yüze ulaşmaktadır.



Resim 4.1. Karar destek sisteminin ilk erişim sayfası

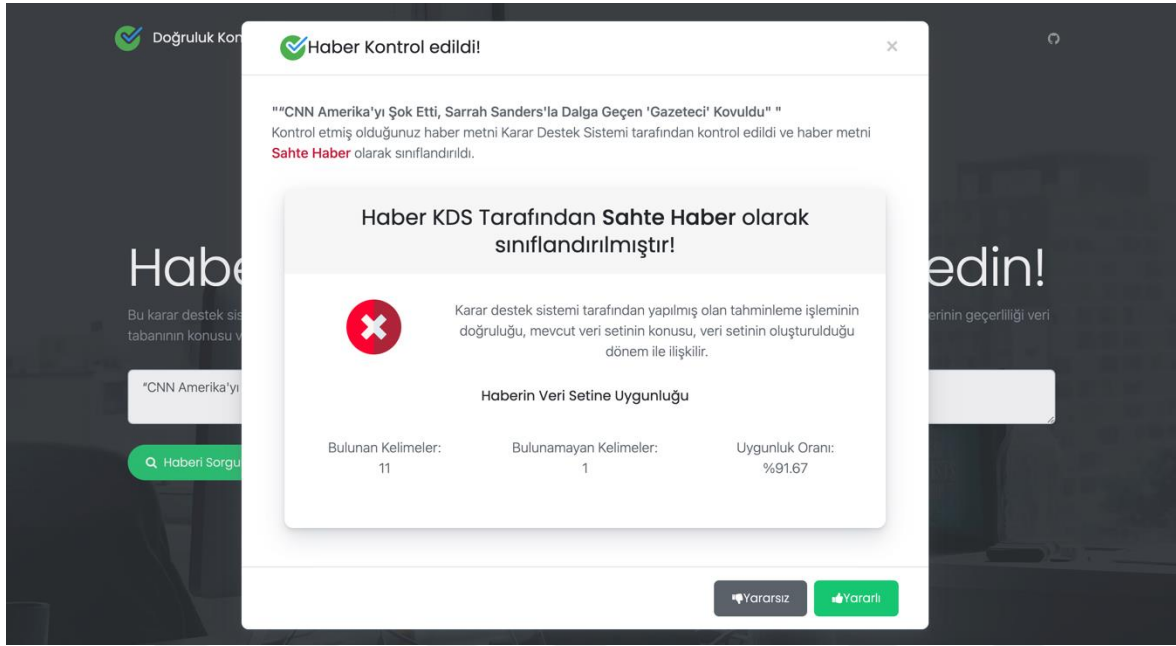
Resim 4.1. den de görüldüğü üzere karar verici karar destek sistemine ilk olarak eriştiğinde, karar problemi ile ilgili bilgilenmesini sağlayan bir bölüm yer almakta ve ek olarak doğrulanmak istenilen haber metninin girilebileceği ve bu haberin sınıflandırılmasını başlatacak olan buton yer aldığı görülmektedir.

Resim 4.2. üzerinde karar verici, karar destek sistemine ulaştığında geliştirilmiş olan sistemi test edebilmesi ve sistemin çalışmasını daha iyi anlaması için ara yüze “Örnek Haber Getir” butonu yerleştirilmiştir. Yerleştirilmiş olan buton karar vericiye sistemi test etmesi için etiketli bir şekilde örnek haber getirmektedir. Getirilen bu haber karar destek sisteminde kullanılacak olan makine öğrenmesi algoritmasının test verileri içinden rastgele seçilmektedir.



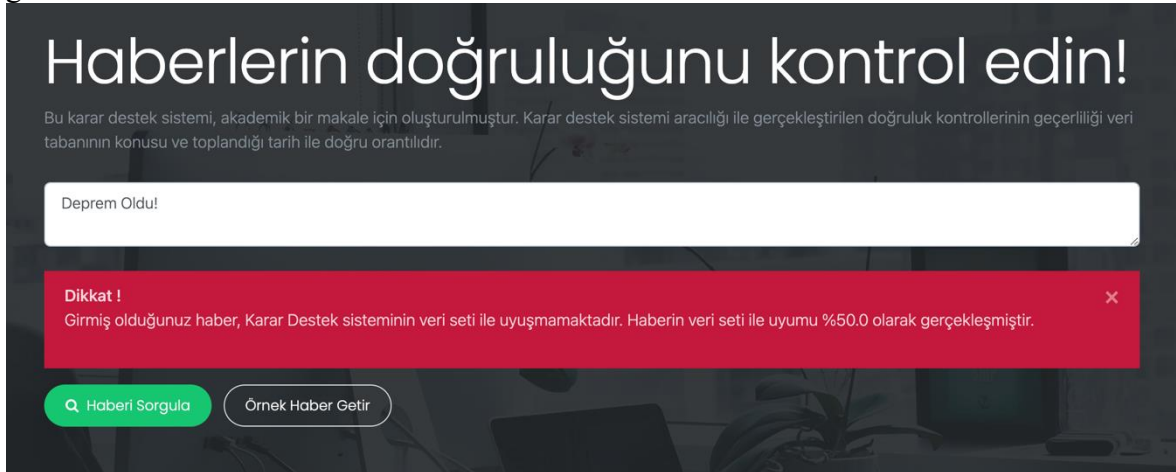
Resim 4.2. Etiketli bilinen bir haberin test amacı ile karar vericiye sunulması

Karar destek sistemi üzerinden sorgulanan haber metninin sınıflandırma sonucu, karar vericiye Resim 4.3. üzerinde gösterilen ara yüz aracılığı ile sunulmaktadır. Bu ara yüzde haberin gerçek ya da sahte olduğu karar vericiye iletilmektedir. Örneğin veri setinde sahte haber olarak etiketlenmiş ve karar vericiye örnek haber olarak sunulmuş olan “CNN Amerika'yı Şok Etti, Sarrah Sanders'la Dalga Geçen 'Gazeteci' Kovuldu” haberi, karar destek sistemi tarafından sahte haber olarak sınıflandırılmıştır.



Resim 4.3. Karar destek sisteminin bir haberi sahte haber olarak sınıflandırması

Karar destek sistemi makine öğrenmesi modelinin eğitildiği haberlere bağımlı olarak çıktı vermektedir. Eğer makine öğrenmesi modeli eğitilmediği kelimeler içeren bir haber ile karşılaşırsa hatalı sınıflandırma ihtimali bilinmeyen kelimeler oranında artacaktır. Eğer bir haber 50% oranında veri setinde geçmeyen ve makine öğrenmesinin hakkında öğrenme gerçekleştirmediği kelimeler içeriyorsa sınıflandırma işlemi gerçekleşmeyecektir. Sınıflandırma işlemi gerçekleşmediğinde karar vericiye Resim 4.4. de bulunan hata mesajı gösterilmektedir.



Resim 4.4. Karar destek sisteminin uyumsuz haber uyarı ekranı.

Resim 4.5.a. üzerinde gösterilen ara yüz ile karar vericilerin Çalışmanın 3. Bölümünde elde edilen 12 makine öğrenmesi modelinden ve daha sonra eklenebilecek olan veri setleri üzerinden sorulama yapılabilmesi amaçlanmıştır.

Resim 4.5.a. Karar destek sisteminin detaylı sahte haber sorgulama ekranı

Resim 4.5.b. Karar destek sisteminin detaylı sahte haber sorgulama ekranı

Resim 4.5.a. incelendiğinde çalışma kapsamında kullanılan 3 farklı özellik çıkarımı algoritmasından biri; 4.5.b. de ise kullanılan makine öğrenmesi algoritmaları tercih edilmek sureti ile çalışmada elde edilmiş olan farklı modellerin kullanılabilmesine imkân sağlanmıştır.

5. TARTIŞMA VE BULGULAR

Metin madenciliğinde her bir kelime farklı ayırıcı öznitelik değerine sahiptir. Bu durum örnekleme yanlılığına sebebiyet verebilmektedir. Örnekleme yanlılığının en aza indirilmesi için çalışmada elde edilen 12 makine öğrenmesi modelinin performans verileri Test/Eğitim yöntemi ile elde edilmiş ve elde edilen verilerin geçerliliği K-fold yöntemi ile doğrulanmıştır.

Test/Eğitim ayrımı yöntemi ile elde edilen sonuçların, veri setinin eğitim ve test olarak ayrılmasından en az şekilde etkilenmesi için veriler 30, 60, 90 defa rastgele karıştırılmıştır. Her bir iterasyonda elde edilen eğitim ve test verisi ile çalışmada elde edilen 12 makine öğrenmesi modeli eğitilerek, bütün modellerin eşit bir şekilde kıyaslanması sağlanmıştır.

Test/Eğitim ayrımı yöntemi için modellerin 30, 60 ve 90 iterasyon eğitilmeleri sonucunda elde edilen ortalama F1 skoru, kesinlik, duyarlılık ve test değeri Çizelge 5.1. de verilmektedir.

Çizelge 5.1. Test/Eğitim ayrımı yöntemi ile elde 12 modelin 30, 60 ve 90 iterasyonun ortalama F1 skoru.

İterasyon	Ort. F1 Skoru	Ort. Kesinlik	Ort. Duyarlılık	Ort. Test Değ.
30	<u>0.7441</u>	<u>0.7568</u>	<u>0.7456</u>	<u>0.7651</u>
60	0.7405	0.7520	0.7428	0.7623
90	0.7364	0.7464	0.7401	0.7597

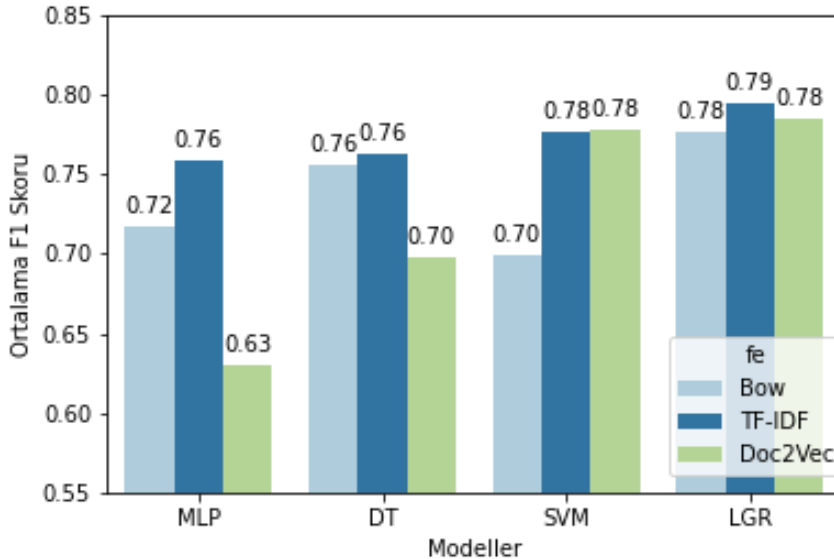
Çizelge 5.1. incelendiğinde en iyi ortalama değerlerine 30 iterasyonda ulaşıldığı görülmektedir.

Çizelge 5.2. Test/Eğitim ayrımı ile elde edilen eğitim ve test verisi ile eğitilen modellerin 30 iterasyon sonucunda elde ettiği F1 skorunun ortalama, standart sapma ve maksimum değerleri ve Şekil 5.1. de elde edilen ortalama F1 değerlerinin grafiği gösterilmektedir.

Çizelge 5.2 Modellerin 30 iterasyonda elde ettiği F1 Skorunun ortalama, standart sapma, maksimum ve k-fold doğrulama değerleri.

Özellik Çıkarımı	Makine Öğrenmesi	K-Fold	F1 Ortalama	F1 Std.	F1 Maks.
Bow	MLP	0.706102	0.716374	0.071052	0.777222
Bow	DT	0.747412	0.755693	0.030693	0.816434
Bow	SVM	0.695110	0.698903	0.024259	0.736979
Bow	LGR	0.789004	0.775615	0.022986	0.826729
TF-IDF	MLP	0.795689	0.758796	0.077022	0.812235
TF-IDF	DT	0.763610	0.762761	0.031052	0.832167
TF-IDF	SVM	0.790338	0.776326	0.036444	0.823957
TF-IDF	LGR	0.801793	0.794814	0.026897	0.857226
Doc2Vec	MLP	0.530705	0.63009	0.187725	0.819931
Doc2Vec	DT	0.701976	0.697791	0.034234	0.765995
Doc2Vec	SVM	0.786527	0.77815	0.025828	0.821995
Doc2Vec	LGR	0.786554	0.7849	0.025707	0.838462

Şekil 5.1. ve Çizelge 5.2. incelendiğinde kurulan Test/Eğitim ayrımı ile elde edilen eğitim ve test verisi ile eğitilen 12 modelden ortalama en iyi F1 Skorunun TF-IDF özellik çıkartma yöntemi ile elde edilmiş öznitelik vektörleri ile eğitilmiş Lojistik Regresyon makine öğrenmesi algoritması ile elde edildiği görülmektedir.



Şekil 5.1. Test/Eğitim ayrımı yöntemi ile kurulan modellerin ortalama F1 skoruna göre performanslarının değerlendirilmesi.

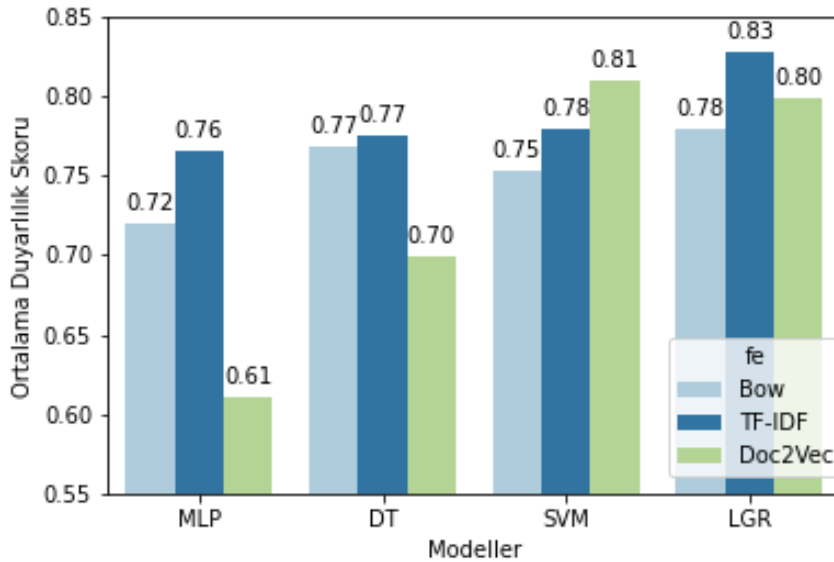
Kurulan modelin duyarlılık değeri gerçek haber olarak sınıflandırılması gereken verinin ne kadarının doğru bir şekilde sınıflandırıldığını ifade eder. Test/Eğitim ayrımı ile elde edilen eğitim ve test verisi ile eğitilen modellerin duyarlılık değerleri 30 iterasyon sonucunda elde

ettiği duyarlılık skorunun ortalama, standart sapma ve maksimum değerleri Çizelge 5.3. de ve elde edilen ortalama duyarlılık değerlerinin grafiği Şekil 5.2. de gösterilmektedir.

Çizelge 5.3. Modellerin 30 iterasyonda elde ettiği duyarlılık Skorunun ortalama, standart sapma, maksimum değerleri ve k-fold doğrulama değerleri.

Özellik Çıkarımı	Makine Öğrenmesi	K-fold	Duyarlılık Ortalama	Duyarlılık Std.	Duyarlılık Maks.
Bow	MLP	0.721200	0.720733	0.050409	0.781977
Bow	DT	0.745716	0.751475	0.028588	0.808515
Bow	SVM	0.692256	0.695402	0.021748	0.729839
Bow	LGR	0.787800	0.775244	0.022661	0.825019
TF-IDF	MLP	0.793866	0.75974	0.054227	0.81114
TF-IDF	DT	0.759973	0.758311	0.030664	0.824175
TF-IDF	SVM	0.795252	0.781367	0.036483	0.829426
TF-IDF	LGR	0.793076	0.78585	0.026449	0.847899
Doc2Vec	MLP	0.608681	0.671474	0.12574	0.815641
Doc2Vec	DT	0.703189	0.698478	0.034137	0.769036
Doc2Vec	SVM	0.779025	0.769917	0.024933	0.814329
Doc2Vec	LGR	0.781315	0.780151	0.025862	0.83177

Çizelge 5.3. ve Şekil 5.2. incelendiğinde, Test/Eğitim ayrımı ile elde edilen eğitim ve test verisi ile eğitilen modellerin en iyi duyarlılık performans değerinin TF-IDF özellik çıkarma ve Lojistik Regresyon makine öğrenmesi algoritmasıyla 0,78 performans değeri ile elde edildiği görülmektedir.



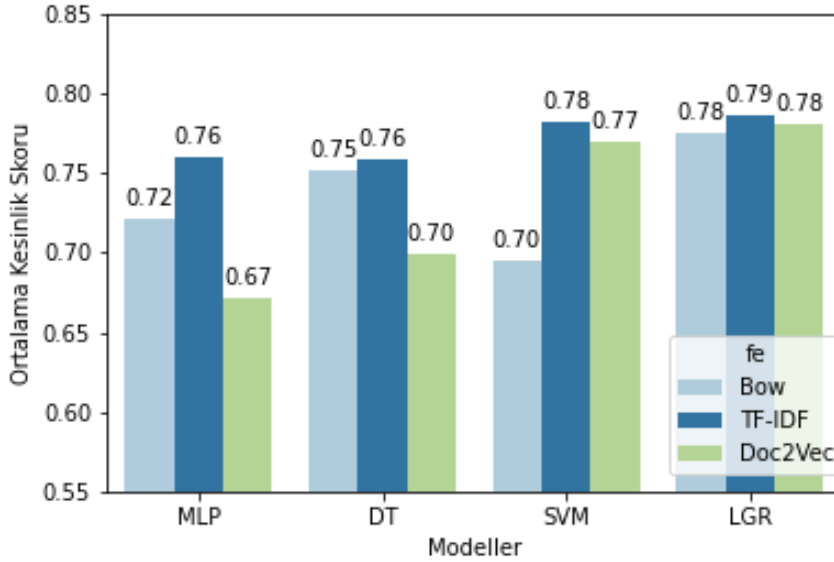
Şekil 5.2. Test Eğitim ayrımı yöntemi ile elde edilen modellerin ortalama duyarlılık performans değerine göre performanslarının değerlendirilmesi.

Test/Eğitim ayrımı yöntemi ile doğrulanan modellerin kesinlik değerleri Çizelge 5.4. ve ortalama kesinlik değerlerinin grafiği Şekil 5.3. de verilmektedir.

Çizelge 5.4. Test/Eğitim ayrımı ile elde edilen Modellerin 30 iterasyonda elde ettiği Kesinlik Skorunun ortalama, standart sapma, maksimum maksimum değerleri ve k-fold doğrulama değerleri.

Özellik Çıkarımı	Makine Öğrenmesi	K-Fold	Kesinlik Ortalama	Kesinlik Std.	Kesinlik Maks.
Bow	MLP	0.698694	0.719784	0.085502	0.788793
Bow	DT	0.758177	0.767653	0.035249	0.838462
Bow	SVM	0.736664	0.752822	0.025116	0.815201
Bow	LGR	0.793144	0.778368	0.023056	0.828811
TF-IDF	MLP	0.800939	0.764927	0.092562	0.82901
TF-IDF	DT	0.774412	0.774632	0.030505	0.850289
TF-IDF	SVM	0.799404	0.778522	0.030555	0.822373
TF-IDF	LGR	0.833824	0.826916	0.023797	0.878571
Doc2Vec	MLP	0.485265	0.611208	0.228467	0.826923
Doc2Vec	DT	0.705153	0.698865	0.034742	0.764278
Doc2Vec	SVM	0.809675	0.809966	0.027671	0.850733
Doc2Vec	LGR	0.799876	0.79851	0.025168	0.85155

Çizelge 5.4. ve Şekil 5.3. incelendiğinde, Test/Eğitim ayrımı ile elde edilen eğitim ve test verisi ile eğitilen modellerden en iyi kesinlik performans değeri olan 0,87 değerine TF-IDF ve LGR modeli ile ulaşıldığı görülmektedir.



Şekil 5.3. Test/Eğitim ayrımı ile kurulan modellerin ortalama kesinlik performans değerine göre performanslarının değerlendirilmesi

Test/Eğitim ayrımı ile elde edilen eğitim ve test verisi ile eğitilen modellerin, F1 skorlarının sunulduğu Çizelge 5.2., duyarlılık değerlerinin sunulduğu Çizelge 5.3. ve kesinlik değerlerinin sunulduğu Çizelge 5.4. incelendiğinde en iyi ortalama F1, duyarlılık ve kesinlik değerlerine TF-IDF özellik çıkarma ve Lojistik Regresyon makine öğrenimi algoritmasının birlikte kullanımı ile oluşturulan modelle erişildiği görülmektedir. Modelin ortalama F1 skoru 0,79, F1 skorunun standart sapması 0,02 ve maksimum F1 skoru 0,85 olarak elde edilmiştir. Modelin duyarlılık ve kesinlik değerleri arasında yalnızca 4%'lük bir fark oluşmuştur. Değerler modelin başarılı bir şekilde öğrenme gerçekleştirdiğini göstermektedir.

Test/Eğitim ayrımı yöntemi ile eğitilen 12 modelin 30 iterasyonda elde ettikleri test skorları üzerinden ortalama, standart sapma ve maksimum değerleri Çizelge 5.5. de verilmektedir.

Çizelge 5.5. Test/Eğitim yöntemi ile elde edilen 12 modelin 30 iterasyonda elde ettiği test skorunun ortalama, standart sapma, maksimum ve k-fold doğrulama değerleri..

Özellik Çıkarımı	Makine Öğrenmesi	K-Fold	Test Ortalama	Test Std.	Test Maks.
Bow	MLP	0.734103	0.734444	0.038517	0.785714
Bow	DT	0.760502	0.769206	0.031257	0.828571
Bow	SVM	0.728104	0.734127	0.018969	0.771429
Bow	LGR	0.797270	0.78381	0.022391	0.833333
TF-IDF	MLP	0.803955	0.777143	0.042452	0.82381
TF-IDF	DT	0.776488	0.776508	0.028432	0.842857
TF-IDF	SVM	0.797458	0.782063	0.03265	0.828571
TF-IDF	LGR	0.818089	0.811905	0.023253	0.866667
Doc2Vec	MLP	0.665008	0.71127	0.089686	0.828571
Doc2Vec	DT	0.713979	0.70746	0.033711	0.771429
Doc2Vec	SVM	0.801970	0.796667	0.023064	0.833333
Doc2Vec	LGR	0.799184	0.79746	0.022845	0.847619

Test/Eğitim ayrımı doğrulama yöntemi için Çizelge 5.5. incelendiğinde duyarlılık, kesinlik ve F1 değerleri açısından en başarılı model olan TF-IDF ve LGR modelinin 30 iterasyonda test ortalama değerinin 0,81, standart sapma değerinin 0,02 ve maksimum değerinin 0,86 olduğu görülmektedir.



6. SONUÇ VE ÖNERİLER

Tartışma ve bulgular bölümünde elde edilen 12 modelin Test/Eğitim yöntemine göre elde ettiği sonuçlar ve bu sonuçların K-Fold yöntemi ile doğrulanması gerçekleştirilmiştir. 12 Model incelendiğinde TF-IDF özellik çıkarımı yöntemi ile elde edilen özellik vektörleri ile beslenen LGR makine öğrenmesi algoritmasının en başarılı model olduğu görülmektedir. Elde edilen model çalışmanın veri setinden rastgele seçilecek bir eğitim ve test kümesi ile eğitilip karar destek sisteminde kullanılabilir.

Geliştirilen karar destek sistemi www.dogrulukkontrolu.com.tr adresinden kullanıma sunulmaktadır.

İlerleyen çalışmalarda tasarlanan karar destek sisteminde kullanılan siyasi veri setine ek olarak, sosyal hayatın farklı alanlarından haberler içeren veri setleri de dahil edilerek karar destek sisteminin sınıflandırma yapabileceği alan genişletilebilir.

Çalışma kapsamında haberlerin metinsel verileri ile sahte haber tespiti yapılmıştır. Çalışma için elde edilen veri seti haberleri yayın kullanıcıların metin dışı verilerini de içermektedir. Sosyal ağ analizleri ile durum bağlamı yöntemler çalışılabilir, diğer denetimli makine öğrenmesi algoritmaları, denetimsiz veya derin öğrenme algoritmaları kullanılabilir.



KAYNAKLAR

- Agarwal, V., Sultana, H. P., Malhotra, S., & Sarkar, A. (2019). Analysis of Classifiers for Fake News Detection. *Procedia Computer Science*, 165, 377-383. <https://doi.org/10.1016/j.procs.2020.01.035>
- Aksoy, A., & Türkölmez, O. (2020). Dijital çağa demokrasiyi çağırmak: cambridge analytica skandalı. *Journal of Political Administrative and Local Studies*, 41-59. <https://doi.org/10.37460/j.2636-7823/2020.3.1.03>
- al Asaad, B., & Erascu, M. (2018). A tool for fake news detection. *Proceedings - 2018 20th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC 2018*, 379-386. <https://doi.org/10.1109/SYNASC.2018.00064>
- Albahr, A., & Albahar, M. (2020). An Empirical Comparison of Fake News Detection using different Machine Learning Algorithms. *IJACSA) International Journal of Advanced Computer Science and Applications*, 11(9). www.ijacsa.thesai.org
- Ali, R., Qaiser, S., Utara, U., Sintok, M., Kedah, M., Ramsha, A., & Analytics, T. (2018). Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents Text Mining. *Article in International Journal of Computer Applications*, 181(1), 975-8887. <https://doi.org/10.5120/ijca2018917395>
- Amazeen, M. A. (2020). Journalistic interventions: The structural factors affecting the global emergence of fact-checking. *Journalism*, 21(1), 95-111. <https://doi.org/10.1177/1464884917730217>
- Başaran İnce Gökçen. (2010). *Medya ve Toplumsal Hafıza*.
- Bondielli, A., & Marcelloni, F. (2019). A survey on fake news and rumour detection techniques. *Information Sciences*, 497, 38-55. <https://doi.org/10.1016/j.ins.2019.05.035>
- Bradshaw, S., & Howard, P. N. (2019). *The Global Disinformation Order: 2019 Global Inventory of The Global Disinformation Order: 2019 Global Inventory of Organised Social Media Manipulation Organised Social Media Manipulation*. <https://digitalcommons.unl.edu/scholcom/207>
- Bulunmaz, B. (2011). *Yeni Medya Eski Medyaya Karşı: Savaşı Kim Kazandı Ya Da Kim Kazanacak? I New Media vs. Old Media: Who Won The War or Who Will Win?* <http://www.webhatti.com/genel-sohbet/40903-kisaca-internet-tarihi.html>,
- Caminhas, W. M., Vieira, D. A. G., & Vasconcelos, J. A. (2003). Parallel layer perceptron. *Neurocomputing*, 55(3-4), 771-778. [https://doi.org/10.1016/S0925-2312\(03\)00440-5](https://doi.org/10.1016/S0925-2312(03)00440-5)
- Chang, W., Xu, Z., Zhou, S., & Cao, W. (2018). Research on detection methods based on Doc2vec abnormal comments. *Future Generation Computer Systems*, 86, 656-662. <https://doi.org/10.1016/j.future.2018.04.059>

Chen, Q., & Sokolova, M. (2018). *Word2Vec and Doc2Vec in Unsupervised Sentiment Analysis of Clinical Discharge Summaries*. <https://doi.org/10.48550/arxiv.1805.00352>
 Cortes, C. (1995). *Support-Vector Networks* (C. 20).

Doğruluk Payı. (t.y.). Geliş tarihi 05 Aralık 2022, gönderen <https://www.dogrulukpayi.com/>

Fard, A. E., Mohammadi, M., Cunningham, S., & de Walle, B. van. (2019, Haziran 1). Rumour As an Anomaly: Rumour Detection with One-Class Classification. *Proceedings - 2019 IEEE International Conference on Engineering, Technology and Innovation, ICE/ITMC 2019*. <https://doi.org/10.1109/ICE.2019.8792644>

Gottlieb, M., & Dyer, S. (2020). Information and Disinformation: Social Media in the COVID-19 Crisis. *Academic Emergency Medicine*. <https://doi.org/10.1111/acem.14036>

Harris, Z. S. (1954). Distributional Structure. *WORD*, 10(2-3), 146-162. <https://doi.org/10.1080/00437956.1954.11659520>

He, S., Cao, J., Guo, R., & Wang, G. (2012). The capability analysis on the characteristic selection algorithm of text categorization based on F1 measure value. *Proceedings of the 2012 2nd International Conference on Instrumentation and Measurement, Computer, Communication and Control, IMCCC 2012*, 742-745. <https://doi.org/10.1109/IMCCC.2012.180>

Hofman, T. (1999). *Probabilistic latent semantic analysis*. Proceedings of Uncertainty in Artificial Intelligence. https://click.endnote.com/viewer?doi=10.48550%2Farxiv.1301.6705&token=WzExMTUzMDYsIjEwLjQ4NTUwL2FyeG12LjEzMDEuNjcwNSJd.onCNxmFWe6EgTgMbqToqbQi00_M

Holzinger, A. (2017). Introduction to Machine Learning & Knowledge Extraction (MAKE). *Machine Learning and Knowledge Extraction*, 1(1), 1-20. <https://doi.org/10.3390/MAKE1010001>

Horvat, T., Havaš, L., & Srpak, D. (2020). The impact of selecting a validation method in machine learning on predicting basketball game outcomes. *Symmetry*, 12(3). <https://doi.org/10.3390/SYM12030431>

Hosack, B., Hall, D., Paradise, D., & Courtney, J. F. (2012). A look toward the future: Decision support systems research is alive and well. *Journal of the Association for Information Systems*, 13(5), 315-340. <https://doi.org/10.17705/1JAIS.00297>

Indra, S. T., Wikarsa, L., & Turang, R. (2017). Using logistic regression method to classify tweets into the selected topics. *2016 International Conference on Advanced Computer Science and Information Systems, ICACSIS 2016*, 385-390. <https://doi.org/10.1109/ICACSIS.2016.7872727>

- Isaakidou, marianna, Zoulias, E., & Diomidous, marianna. (2021). *Machine Learning to Identify Fake News for Covid-19*. Public Health and Informatics. https://books.google.com.tr/books?hl=tr&lr=&id=81A2EAAAQBAJ&oi=fnd&pg=PA108&dq=fake+news+decision+support+system&ots=2k9bN9sS1g&sig=SZznllm7UxCWQYeKjEKSOSlkK_s&redir_esc=y#v=onepage&q=fake%20news%20decision%20support%20system&f=false
- Janze, C., & Risius, M. (2017). Association for Information Systems AIS Electronic Library (AISeL) Automatic Detection of Fake News on Social Media Platforms Recommended Citation. *Automatic Detection of Fake News on Social Media Platforms*. <http://aisel.aisnet.org/pacis2017http://aisel.aisnet.org/pacis2017/261>
- Kocabay Şener, N. (2018). “Doğruluk Kontrol Merkezi” ve “Yalan Haber” Kavramlarına İlişkin İçeriklerin Medyada Yansımalarının Araştırılması. *Akdeniz Üniversitesi İletişim Fakültesi Dergisi*, 351-369. <https://doi.org/10.31123/akil.454397>
- Lovins, J. B. (1968). Development of a Stemming Algorithm*. *Mechanical Translation and Computational Linguistics*, 11(2).
- Mahmud, Y., Shaeali, S., & Mutalib, S. (2021). Comparison of Machine Learning Algorithms for Sentiment Classification on Fake News Detection. *IJACSA International Journal of Advanced Computer Science and Applications*, 12(10), 2021. www.ijacsa.thesai.org
- Meel, P., & Vishwakarma, D. K. (2020). Fake news, rumor, information pollution in social media and web: A contemporary survey of state-of-the-arts, challenges and opportunities. İçinde *Expert Systems with Applications* (C. 153, s. 112986). Elsevier Ltd. <https://doi.org/10.1016/j.eswa.2019.112986>
- Newman, N., Fletcher, R., Kalogeropoulos, A., Levy, D. A. L., & Kleis Nielsen, R. (2018). *Reuters Institute for the Study of Journalism / Digital News Report 2018*.
- Newman, N., Richard Fletcher, with, Schulz, A., Andi, S., & Kleis Nielsen, R. (2020). *Reuters Institute Digital News Report 2020*.
- Okoro, E. M., Abara, B. A., Umagba, A. O., Ajonye, A. A., & Isa, Z. S. (2018). A hybrid approach to fake news detection on social media. *Nigerian Journal of Technology*, 37(2), 454. <https://doi.org/10.4314/NJT.V37I2.22>
- Poddar, K., Amali, G. B. D., & Umadevi, K. S. (2019). Comparison of Various Machine Learning Models for Accurate Detection of Fake News. *2019 Innovations in Power and Advanced Computing Technologies, i-PACT 2019*. <https://doi.org/10.1109/I-PACT44901.2019.8960044>
- Porter, M. F. (1980). An algorithm for suffix stripping. *Celebrating 40 Years Of Ict In Libraries, Museums And Archives* . <https://doi.org/10.1108/00330330610681286>

- Pranckevicius, T., & Marcinkevicius, V. (2017). Application of Logistic Regression with part-of-the-speech tagging for multi-class text classification. *2016 IEEE 4th Workshop on Advances in Information, Electronic and Electrical Engineering, AIEEE 2016 - Proceedings*. <https://doi.org/10.1109/AIEEE.2016.7821805>
- Quinlan, J. R. (1996). *Learning Decision Tree Classifiers*.
- Ramos, J. (2003). Using TF-IDF to Determine Word Relevance in Document Queries. *Proceedings of the First Instructional Conference on Machine Learning*, 242(1), 29-48.
- Reilly, I. (2018). F for Fake: Propaganda! Hoaxing! Hacking! Partisanship! and Activism! in the Fake News Ecology. *Journal of American Culture*, 41(2), 139-152. <https://doi.org/10.1111/jacc.12834>
- Ruchansky, N., Seo, S., & Liu, Y. (2017). CSI: A hybrid deep model for fake news detection. *International Conference on Information and Knowledge Management, Proceedings, Part F1318*, 797-806. <https://doi.org/10.1145/3132847.3132877>
- Sadiq, S., Wagner, N., Shyu, M. L., & Feaster, D. (2019). High Dimensional Latent Space Variational AutoEncoders for Fake News Detection. *Proceedings - 2nd International Conference on Multimedia Information Processing and Retrieval, MIPR 2019*, 437-442. <https://doi.org/10.1109/MIPR.2019.00088>
- Shabani, S., & Sokhn, M. (2018). Hybrid machine-crowd approach for fake news detection. *Proceedings - 4th IEEE International Conference on Collaboration and Internet Computing, CIC 2018*, 299-306. <https://doi.org/10.1109/CIC.2018.00048>
- Shu, K., Mahudeswaran, D., Wang, S., Lee, D., & Liu, H. (2018). *FakeNewsNet: A Data Repository with News Content, Social Context and Spatiotemporal Information for Studying Fake News on Social Media*. <http://arxiv.org/abs/1809.01286>
- Sprague, R. H. (1980). A framework for the development of decision support systems. *MIS Quarterly: Management Information Systems*, 4(4), 1-26. <https://doi.org/10.2307/248957>
- Taskin, S. G., Kucuksille, E. U., & Topal, K. (2022). Detection of Turkish Fake News in Twitter with Machine Learning Algorithms. *Arabian Journal for Science and Engineering*, 47(2), 2359-2379. <https://doi.org/10.1007/s13369-021-06223-0>
- Teyit - Şüpheli bilgileri inceleyen doğrulama organizasyonu*. (t.y.). Geliş tarihi 05 Aralık 2022, gönderen <https://teyit.org/>
- Tolles, J., & Meurer, W. J. (2016). Logistic regression: Relating patient characteristics to outcomes. İçinde *JAMA - Journal of the American Medical Association* (C. 316, Issue 5). <https://doi.org/10.1001/jama.2016.7653>
- Tong, S., & Koller, D. (2001). Support Vector Machine Active Learning with Applications to Text Classification. *Journal of Machine Learning Research*, 45-66.

- Webster, J. J., & Kit, C. (1992). Tokenization as the initial phase in nlp. *proc. of coling-92*.
- Wei, X., & Croft, W. B. (2006). LDA-based document models for ad-hoc retrieval. *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. <https://scihub.se/https://doi.org/10.1145/1148170.1148204>
- Ye, J., Li, G., Chen, D., Yang, H., Zhe, S., & Xu, Z. (2020). Block-term tensor neural networks. *Neural Networks*, 130, 11-21. <https://doi.org/10.1016/j.neunet.2020.05.034>
- Zhao, G., Wu, Y., Chen, F., Zhang, J., & Bai, J. (2015). Effective feature selection using feature vector graph for classification. *Neurocomputing*, 151(P1), 376-389. <https://doi.org/10.1016/J.NEUCOM.2014.09.027>
- Zhu, G., Zhou, Y., Zhou, F., Wu, M., Zhan, X., Si, Y., Wang, P., & Wang, J. (2021). Proactive Personality Measurement Using Item Response Theory and Social Media Text Mining. *Frontiers in Psychology*, 12. <https://doi.org/10.3389/fpsyg.2021.705005>
- Zhu, H., Tsang, E. C. C., & Zhu, J. (2018). Training an extreme learning machine by localized generalization error model. *Soft Computing*, 22(11), 3477-3485. <https://doi.org/10.1007/S00500-018-3012-5>



GAZİLİ OLMAK AYRICALIKTIR