



**4. DÖNEM 2. GRUP
VERİ ANALİTİĞİ
BİTİRME PROJESİ RAPORU**

Sosyal Medyada Dezenformasyon Yayılım Ağı Analizi
Rusya–Ukrayna Savaşı Dezenformasyon Tespiti

Ağustos, 2026

İÇİNDEKİLER

1. GİRİŞ

- 1.1. Projenin Amacı
- 1.2. Veri Seti Genel Bakış

2. MATERYAL VE YÖNTEM

- 2.1. Kullanılan Kütüphane ve Araçlar
- 2.2. Veri Ön İşleme Yöntemi
- 2.3. Özellik Çıkarımı Yöntemi (TF-IDF)
- 2.4. Sınıflandırma Algoritmaları

3. UYGULAMA

- 3.1. Veri Ön İşleme Uygulaması
 - 3.1.1. Mojibake (Bozuk Karakter) Düzeltmesi
 - 3.1.2. Yabancı Script / Dil Tespiti
 - 3.1.3. Konu Dışı İçeriğin Temizlenmesi
 - 3.1.4. Çok Kısa Metinlerin Temizlenmesi
- 3.2. Özellik Çıkarımı Uygulaması
- 3.3. Sınıflandırma Modelleri
- 3.4. Hiperparametre Optimizasyonu
 - 3.4.1. Linear SVM
 - 3.4.2. Random Forest
 - 3.4.3. XGBoost
- 3.5. Nihai Model Karşılaştırması
- 3.6. Model Hata Analizi
- 3.7. Özellik Önem Analizi
- 3.8. Kritik Bulgular ve Sınırlılıklar
- 3.9. Veri Hikâyeleştirme: Sekiz Temel Bulgu
 - 3.9.1. Dezenformasyon Hangi Konularda Yayılıyor?
 - 3.9.2. Metin Uzunluğu Karşılaştırması
 - 3.9.3. En Sık Kullanılan Kelimeler
 - 3.9.4. Bigram / Trigram Analizi
 - 3.9.5. Duygu Tonu Karşılaştırması
 - 3.9.6. Makine Öğrenmesi Modellerinin Genel Başarısı
 - 3.9.7. Kategori Bazında Model Zorlukları
 - 3.9.8. Etiket Güveni (confidence) ile Model Performansı

4. SONUÇ

5. KAYNAKÇA

1. GİRİŞ

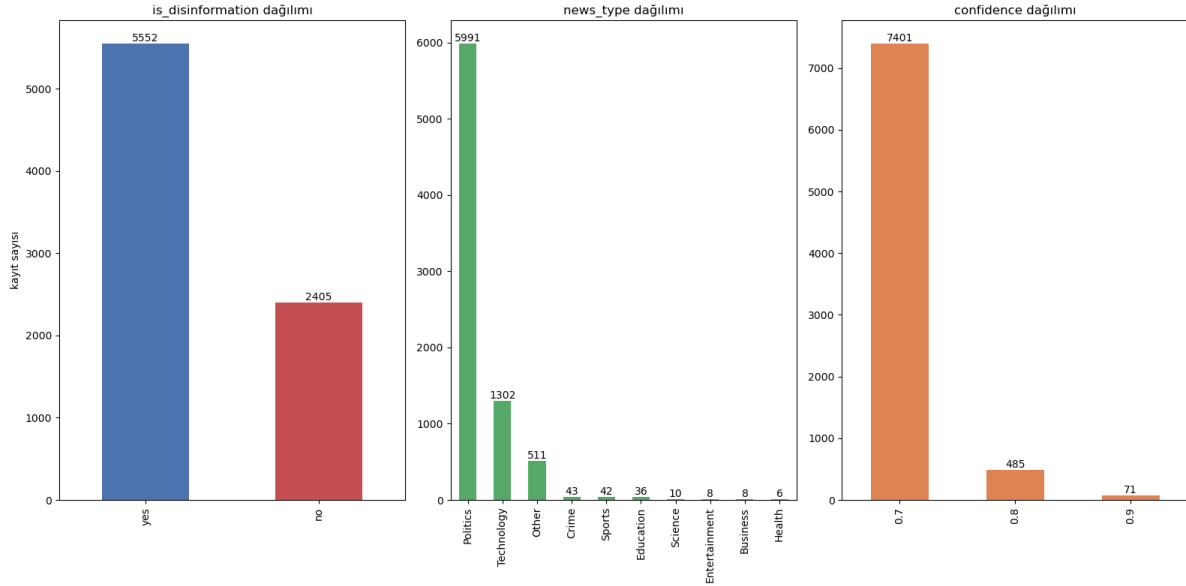
1.1. Projenin Amacı

Bu proje, Rusya–Ukrayna savaşı bağlamında sosyal medya ve haber kaynaklarından toplanan İngilizce metinleri, dezenformasyon içerip içermediğine göre (is_disinformation: yes/no) otomatik olarak sınıflandırmayı amaçlamaktadır. Çalışma; veri temizleme, özellik çıkarımı (TF-IDF), altı farklı makine öğrenmesi algoritmasının karşılaştırılması, üç modelin hiperparametre optimizasyonu, hata analizi, özellik önem analizi ve veriye dayalı sekiz ayrı bulgu (data storytelling) bölümünden oluşmaktadır.

Proje kapsamında ağ analizi (network analysis) yapılmamış, sadece temel/yaygın Python kütüphaneleri (pandas, numpy, scikit-learn, matplotlib) kullanılmıştır; XGBoost, nltk ve wordcloud gibi ek kütüphaneler yalnızca destekleyici/karşılaştırmalı analizlerde kullanılmıştır.

1.2. Veri Seti Genel Bakış

Özellik	Değer
Toplam kayıt sayısı	7.957
Sütunlar	post_id, text, is_disinformation, news_type, confidence
Eksik değer	Yok (0 satır)
Tekrarlı kayıt	Yok (0 satır)
Sınıf dağılımı (ham)	yes: 5.552 (%69,8) • no: 2.405 (%30,2)
Baskın kategori (news_type)	Politics (5.991 / %75,3)
confidence dağılımı	0,7: 7.401 • 0,8: 485 • 0,9: 71



Şekil 1.2.1. — Sınıf, kategori ve etiket güven (confidence) dağılımları (ham veri).

2. MATERYAL VE YÖNTEM

2.1. Kullanılan Kütüphane ve Araçlar

Proje Python (Jupyter Notebook) ortamında geliştirilmiştir. Veri işleme için pandas ve numpy, metin temizliği için re (regex), görselleştirme için matplotlib, makine öğrenmesi modelleri ve metrikleri için scikit-learn kullanılmıştır. Karşılaştırma amacıyla xgboost (boosting modeli) ve nltk (VADER duygu analizi) kütüphaneleri de kullanılmıştır.

2.2. Veri Ön İşleme Yöntemi

Ham veri; bozuk karakter kodlama (mojibake) düzeltilmesi, yabancı script/dil tespiti, konu dışı içerik filtrelemesi ve çok kısa metinlerin çıkarılması olmak üzere dört aşamalı bir temizlik sürecinden geçirilmiştir. Yöntemin ayrıntıları ve gerekçeleri Bölüm 3.1'de sunulmuştur.

2.3. Özellik Çıkarımı Yöntemi (TF-IDF)

Metin verisi, `TfidfVectorizer(stop_words="english", min_df=3, max_df=0.9, ngram_range=(1,2))` parametreleriyle sayısal vektörlere dönüştürülmüştür. Tek kelime (unigram) ve ikili kelime grupları (bigram) birlikte modellenmiş, ardından yalnızca sayısal karakterlerden oluşan anlamsız token'lar (örn. tarih kalıntıları) kelime dağarcığından çıkarılmıştır.

2.4. Sınıflandırma Algoritmaları

Aşağıdaki altı sınıflandırma algoritması karşılaştırmalı olarak değerlendirilmiştir: Logistic Regression, Linear SVM, Multinomial Naive Bayes, Complement Naive Bayes, Random Forest ve XGBoost. En güçlü üç aday (Linear SVM, Random Forest, XGBoost), GridSearchCV ile hiperparametre optimizasyonuna tabi tutulmuştur.

3. UYGULAMA

3.1. Veri Ön İşleme Uygulaması

3.1.1. Mojibake (Bozuk Karakter) Düzeltmesi

Ham text sütununda, bir kısım kayıta UTF-8 baytlarının yanlışlıkla cp1252 kod sayfasıyla çözülmesinden kaynaklanan karakter bozulmaları (mojibake) tespit edilmiştir — özellikle Kiril/Çince içerikli veya emoji içeren gönderilerde. Bu sorun, ham metni cp1252 ile yeniden kodlayıp UTF-8 olarak çözen (encode/decode) bir `fix_mojibake()` fonksiyonuyla otomatik olarak düzeltilmiştir. Bu adım, sonraki dil tespiti adımının doğru çalışabilmesi için kritik önem taşımaktadır: düzeltme yapılmadan, bozuk karakterler yanlışlıkla “Latin alfabesi” aralığına düşmekte ve gerçek yabancı-dil içeriğin tespitini engellemektedir.

3.1.2. Yabancı Script / Dil Tespiti

İlk denemede `langdetect` kütüphanesi kullanılmış, ancak kısa ve gazetecilik üslubuyla yazılmış İngilizce başlıklarda (örn. “CNN: UK delivers long-range missiles to Ukraine.”) sistematik olarak yanlış dil tahminleri (örn. Norveççe) ürettiği görülmüştür. Bunun yerine, metindeki alfabetik karakterlerin ne kadarının Latin-dışı bir script'e (Kiril, Çince vb.) ait olduğunu ölçen özel bir `non_latin_ratio()` fonksiyonu geliştirilmiştir. Test edilen eşik değerleri sonucunda %16 eşiği, gerçek yabancı-dil içerik (oran: 0,63–0,79) ile İngilizce metin içindeki tekil yabancı kelime/hashtag'leri (oran: ~0,07–0,15) net biçimde ayırdığı için seçilmiştir.

Aşama	Kayıt Sayısı	Açıklama
Filtre öncesi	7.957	—
Filtre sonrası	7.954	3 kayıt (BBC Çince haberleri) çıkarıldı

3.1.3. Konu Dışı İçeriğin Temizlenmesi

Veri seti Rusya–Ukrayna savaşına odaklandığından, İsrail–Gazze, İran–Suriye, Tayvan, Afganistan, COVID-19 gibi farklı gündemlere ait ancak anahtar kelime çakışması nedeniyle veri setine karışmış olabilecek kayıtlar iki ayrı grupta taranmıştır. Ukrayna/Rusya ile ilişkili terimler (ukraine, russia, putin, nato, kremlin vb.) içeren kayıtlar ‘ilgili’ kabul edilerek korunmuş, geri kalanlar çıkarılmıştır. İkinci grup ayrıca elle incelenmiş, 2 kayıt gerçekten ilgili bulunarak veri setinde tutulmuştur.

Grup	Aday Sayısı	Alakasız (Çıkarılan)
İsrail / Lübnan / Gazze	189	133
İran / Suriye / Tayvan / Afganistan / Libya / Covid / Kosova	270	268 (2 kayıt elle korundu)
TOPLAM (tekilleştirilmiş)	515 aday	356

3.1.4. Çok Kısa Metinlerin Temizlenmesi

Temizlik sonrası `clean_text` uzunluğu 25 karakterin altında kalan 12 kayıt (çoğunlukla kaynaktaki zaten kırpılmış gelen başlıklar) veri setinden çıkarılmıştır.

Aşama	Kayıt Sayısı
Alakasız içerik temizliği sonrası	7.598
Kısa metin temizliği sonrası (NİHAİ)	7.586

3.2. Özellik Çıkarımı Uygulaması

Temizlenmiş metin, %80 eğitim (6.068 kayıt) – %20 test (1.518 kayıt) olarak, sınıf oranı korunacak şekilde (stratify) bölünmüştür.

Ölçüt	Değer
Eğitim seti boyutu	6.068 kayıt

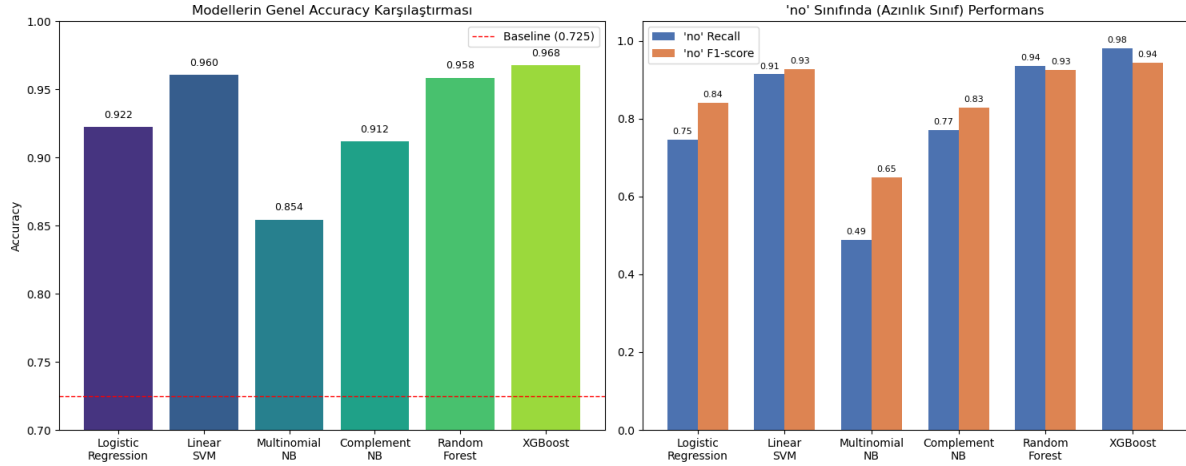
Ölçüt	Değer
Test seti boyutu	1.518 kayıt
Başlangıç özellik sayısı (TF-IDF)	11.143
Sayısal-sadece token temizliği sonrası	11.037

CountVectorizer ile aynı parametrelerle oluşturulan alternatif temsilin, TF-IDF ile birebir aynı kelime dağarcığını ürettiği doğrulanmıştır. Ayrıca yalnızca rakam/böşluktan oluşan 106 token kelime dağarcığından çıkarılmış; “000 soldiers”, “10 million” gibi anlamlı sayı+kelime ikilileri bilinçli olarak korunmuştur.

3.3. Sınıflandırma Modelleri

Çoğunluk sınıfı baseline'ı (her zaman “yes” tahmini), test setinde %72,46 accuracy vermektedir. Aşağıda, varsayılan ayarlarla eğitilen altı modelin sonuçları listelenmiştir.

Model	Accuracy	“no” Recall	“no” Precision	“no” F1
Baseline (çoğunluk sınıfı)	0,725	0,00	0,00	0,00
Multinomial Naive Bayes	0,854	0,49	0,97	0,65
Complement Naive Bayes	0,912	0,77	0,89	0,83
Logistic Regression	0,922	0,75	0,96	0,84
Random Forest (varsayılan)	0,958	0,94	0,92	0,93
Linear SVM (varsayılan)	0,960	0,91	0,94	0,93
XGBoost (varsayılan)	0,968	0,98	0,91	0,94



Şekil 3.3.1. — Varsayılan ayarlarla altı modelin accuracy ve “no” sınıfı performans karşılaştırması.

Veri seti dengesiz olduğundan (yes: %72, no: %28), yalnızca accuracy'ye bakmak yanıltıcıdır. Bu nedenle projede sistematik olarak “no” sınıfının recall değeri de takip edilmiştir. Multinomial NB, en yüksek “no” precision'a (0,97) sahip olsa da recall'ü yalnızca 0,49'dur — bu, Complement NB'nin neden tercih edildiğini somut biçimde göstermektedir.

3.4. Hiperparametre Optimizasyonu

En güçlü üç aday model, GridSearchCV (5 katlı çapraz doğrulama, scoring="f1_macro") ile optimize edilmiştir.

3.4.1. Linear SVM

Aranan parametreler: $C \in \{0,01; 0,1; 0,5; 1; 2; 5; 10\}$, $class_weight \in \{None, "balanced"\}$. En iyi kombinasyon: $C=2$, $class_weight="balanced"$ (CV Macro F1: 0,9474).

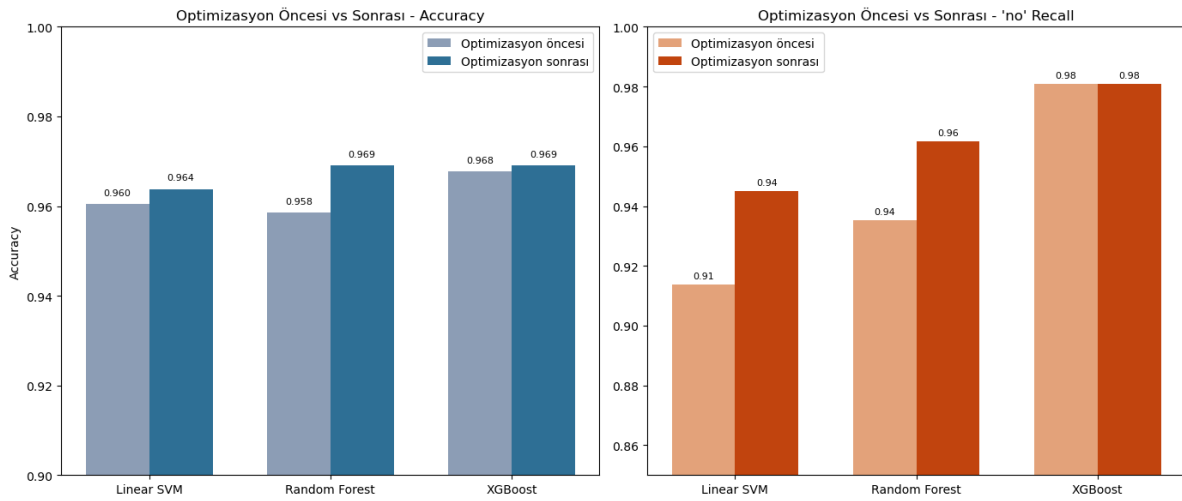
3.4.2. Random Forest

Aranan parametreler: $n_estimators \in \{200, 300\}$, $max_depth \in \{None, 30\}$, $min_samples_leaf \in \{1, 2\}$, $class_weight \in \{None, "balanced"\}$. En iyi kombinasyon: $n_estimators=300$, $max_depth=None$, $min_samples_leaf=1$, $class_weight="balanced"$ (CV Macro F1: 0,9583).

3.4.3. XGBoost

Aranan parametreler: $n_estimators \in \{200, 300\}$, $max_depth \in \{3, 6, 9\}$, $learning_rate \in \{0,05; 0,1; 0,2; 0,3\}$, $scale_pos_weight \in \{1, \text{sınıf oranı}\}$. En iyi kombinasyon: $learning_rate=0,3$, $max_depth=3$, $n_estimators=300$, $scale_pos_weight=1$ (CV Macro F1: 0,9569).

Model	Öncesi Accuracy	Sonrası Accuracy	Öncesi "no" Recall	Sonrası "no" Recall
Linear SVM	0,960	0,964	0,91	0,94
Random Forest	0,958	0,969	0,94	0,96
XGBoost	0,968	0,969	0,98	0,98

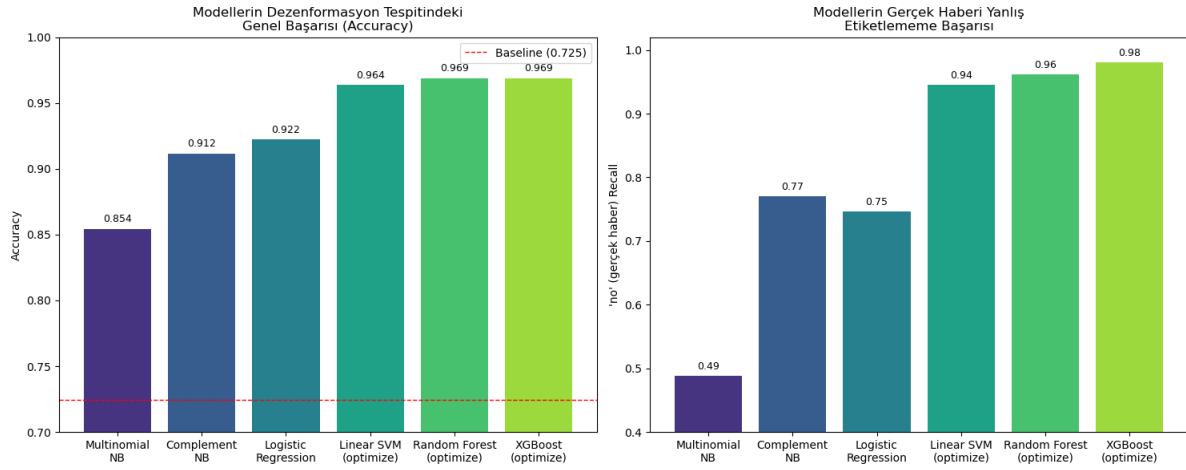


Şekil 3.4.3.1. — Optimizasyon öncesi ve sonrası accuracy ve "no" recall karşılaştırması.

$class_weight="balanced"$ parametresi, hem SVM hem Random Forest'ta belirgin bir iyileşme sağlamıştır. XGBoost'ta ise optimizasyon marjinal bir katkı sağlamıştır, çünkü model zaten en güçlü başlangıç noktasındaydı.

3.5. Nihai Model Karşılaştırması

Model	Accuracy	"no" Recall	"no" Precision	Not
Optimized XGBoost	0,969	0,98	0,91	En yüksek recall, en düşük "no" precision
Optimized Random Forest	0,969	0,96	0,93	Accuracy'de XGBoost'a eşit, daha dengeli
Optimized Linear SVM	0,964	0,94	0,93	En basit / yorumlanabilir, hâlâ güçlü
Logistic Regression	0,922	0,75	0,96	Baz referans
Complement NB	0,912	0,77	0,89	Dengesiz veri için tasarlanmış, orta performans
Multinomial NB	0,854	0,49	0,97	En zayıf, pedagojik referans noktası



Şekil 3.5.1. — Nihai modellerin (optimize edilmiş SVM/RF/XGBoost dahil) genel karşılaştırması.

Optimized XGBoost ve Optimized Random Forest, accuracy açısından birbirine eşit (0,969) en iyi sonucu vermektedir. Model seçimi, yanlış-pozitif ile yanlış-negatif maliyetlerinin göreceli önemine bağlıdır; dezenformasyon tespitinde yanlış-negatif azaltmak genellikle daha kritik kabul edildiğinden, XGBoost bu proje için önerilen nihai modeldir.

3.6. Model Hata Analizi

Model	TN	FP	FN	TP	Toplam Hata
Linear SVM (optimize)	395	23	32	1068	55
Random Forest (optimize)	402	16	33	1067	49
XGBoost (optimize)	410	8	39	1061	47

Modeller arasında ortak, tekrarlayan bir hata kalıbı tespit edilmiştir: doğru tahmin edilen kayıtların ortalama uzunluğu ~280–310 karakter iken, yanlış tahmin edilenlerin ortalaması ~140–150 karaktere düşmektedir. Bu örüntü algoritmadan bağımsızdır ve veri kalitesinin (kırpılmış/eksik metinler) doğrudan bir sonucudur.

False Positive örneklerinin incelenmesi, modelin “claim”, “denied”, “allegedly” gibi kelimeleri gördüğünde yanıltıldığını göstermektedir; bu kelimeler gerçek haberlerde de bir iddiayı aktarırken kullanılabilir.

3.7. Özellik Önem Analizi

Üç farklı algoritma ailesi (doğrusal — Linear SVM, ağaç-tabanlı — Random Forest, olasılıksal — Multinomial NB) birbirinden bağımsız olarak eğitilmiş ve en ayırt edici kelimeleri incelenmiştir.

Model	“yes” yönünde en güçlü kelimeler	“no” yönünde en güçlü kelimeler
Linear SVM (yönlü)	propaganda, fake, claims, allegedly, alleged, claim, fakes, rumors, western, west	university, statistics, scientific, confirmed, war briefing, published, drones
Random Forest (yönsüz)	propaganda, ukrainian, fake, ukraine, claims, west, allegedly, western, russia, russian	(yönsüz metrik — sadece genel önem)
Multinomial NB (yönlü)	propaganda, fake, west, western, regime, allegedly, claims, russian propaganda, kyiv regime	war briefing, university, statistics, bbc news, scientific

Üç modelin de bağımsız olarak aynı çekirdek kelime kümesine ulaşması, öğrenilen sinyalin rastgele bir yapay eser olmadığını göstermektedir. Ancak bu kelimelerin niteliği, dezenformasyonun kendi anlatısından çok, onu tarif eden/çürüten meta-dile işaret etmektedir.

3.8. Kritik Bulgular ve Sınırlılıklar

3.8.1. Metin Uzunluğu Dominansı

Dezenformasyon içerikli metinler (ortalama 356 karakter), gerçek haberlere (ortalama 82 karakter) kıyasla ortalama 4 kat daha uzundur (Mann-Whitney U, $p \approx 0$). Yalnızca metin uzunluğunu özellik olarak kullanan basit bir model bile %92,5 accuracy elde etmektedir — bu, tam TF-IDF tabanlı modelin performansına (%92,2) neredeyse eşittir.

3.8.2. Kaynak/Kanal Şablon Dili

Bigram/trigram analizinde öne çıkan bazı ifadeler, genel bir “dezenformasyon dili”nden çok, belirli bir fact-check kanalının şablon başlıklarına ait olabilir; bu durumda model kısmen belirli bir kaynağın yazım biçimini ezberlemiş olabilir.

3.8.3. Model Ne Öğrendi: İçerik mi, Meta-Dil mi?

Özellik önem analizinde öne çıkan kelimeler incelendiğinde, modelin dezenformasyonun kendi anlatı içeriğinden çok, onu tarif eden fact-check diline duyarlı olduğu görülmektedir.

3.8.4. Küçük Örneklemli Kategoriler

news_type dağılımı büyük ölçüde dengesizdir. Health, Entertainment, Science, Business gibi kategoriler test setinde 10'un altında kayıtlarla temsil edilmektedir; bu kategorilere ait bulgular istatistiksel olarak güvenilir değildir.

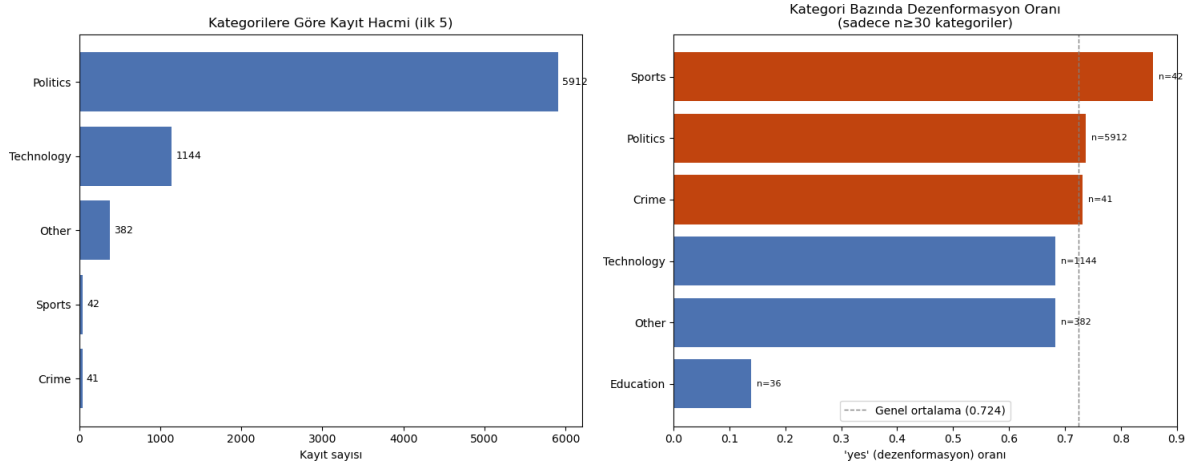
3.8.5. Etiket Güven (confidence) Sütununun Sınırlı Bilgi Değeri

confidence sütunu neredeyse hiç çeşitlilik göstermemektedir (kayıtların %93'ü 0,7 değerinde); bu nedenle etiketleme güveni ile model performansı arasında istatistiksel olarak güvenilir bir ilişki kurulamamıştır.

3.9. Veri Hikâyeleştirme: Sekiz Temel Bulgu

Sunum ve raporlama amacıyla, veri seti üzerinde sekiz ayrı araştırma sorusu incelenmiş ve görselleştirilmiştir.

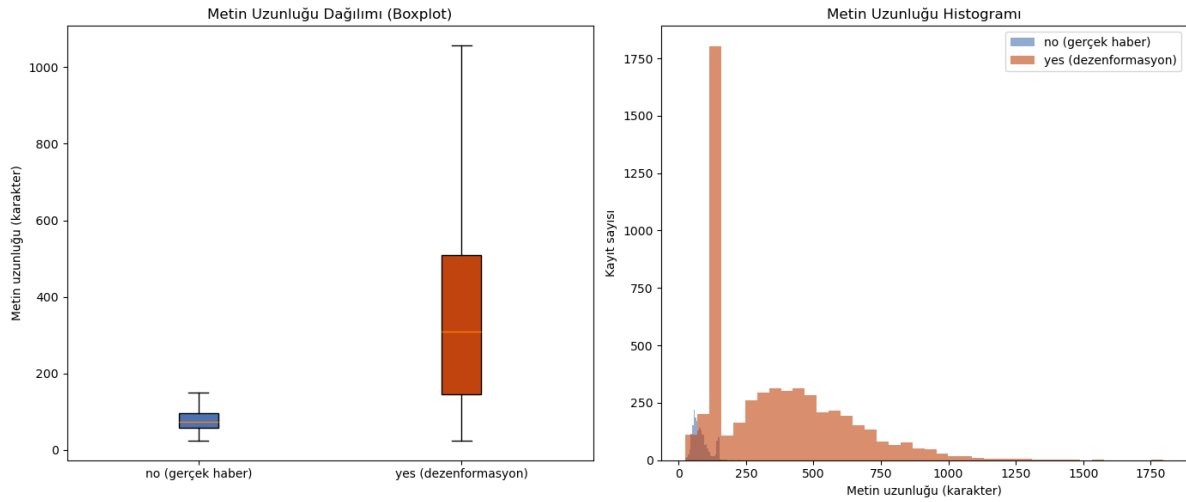
3.9.1. Dezenformasyon Hangi Konularda Yayılıyor?



Şekil 3.9.1.1. — Kategori bazında kayıt hacmi ve dezenformasyon oranı (yalnızca $n \geq 30$ kategoriler).

Politics, Technology ve Other kategorileri veri setinin %98'ini oluşturmakta ve dezenformasyon oranları genel ortalamaya yakındır (%68–74). Education kategorisi ($n=36$) belirgin şekilde daha düşük bir orana sahiptir (%14).

3.9.2. Metin Uzunluğu Karşılaştırması



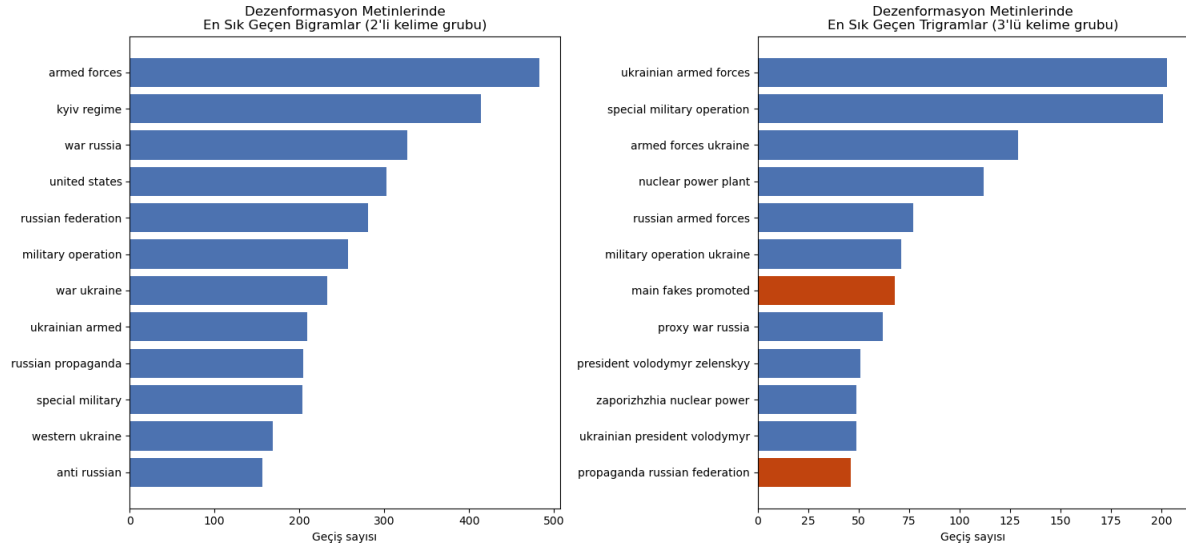
Şekil 3.9.2.1. — Metin uzunluğu dağılımı (boxplot ve histogram).

Bkz. Bölüm 3.8.1 — dezenformasyon metinleri ortalama 356 karakter, gerçek haberler ortalama 82 karakterdir ($p \approx 0$).

3.9.3. En Sık Kullanılan Kelimeler

Ham kelime sayımına göre her iki grup da aynı konu kelimelerini paylaşmaktadır (ukraine, russia, kyiv). Ancak yalnızca dezenformasyon metinlerinde geçen kelimeler arasında propaganda, regime, fake, nazi, claims, allegedly, provocation öne çıkmaktadır.

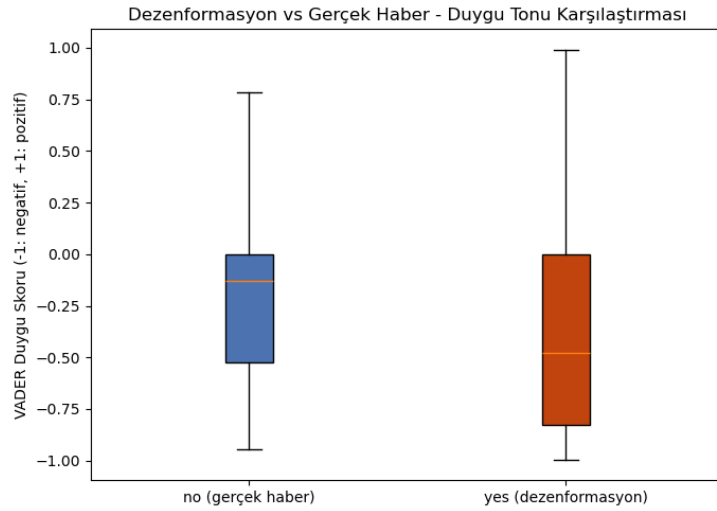
3.9.4. Bigram / Trigram Analizi



Şekil 3.9.4.1. — Dezenformasyon metinlerinde en sık geçen ikili ve üçlü kelime grupları.

“special military operation” ve “kyiv regime” gibi ifadeler, Rus devlet medyasının savaşı çerçevelemek için kullandığı bilinen propaganda terimleridir.

3.9.5. Duygu Tonu Karşılaştırması



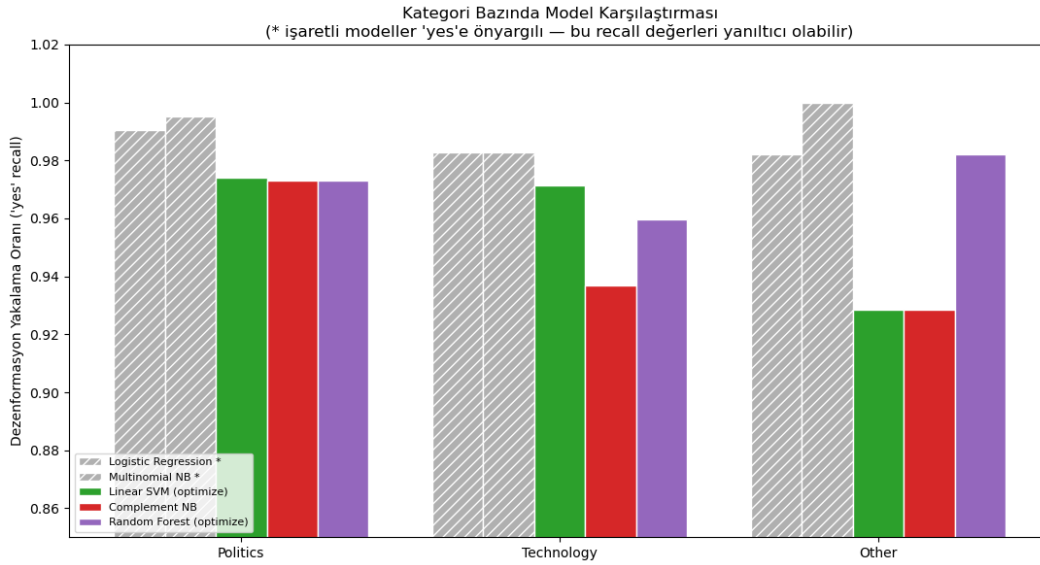
Şekil 3.9.5.1. — VADER duygu skoruna göre dezenformasyon vs. gerçek haber karşılaştırması.

Dezenformasyon metinleri (medyan: -0,477), gerçek haberlere (medyan: -0,128) kıyasla istatistiksel olarak anlamlı derecede daha negatif bir ton taşımaktadır ($p \approx 7,16 \times 10^{-54}$).

3.9.6. Makine Öğrenmesi Modellerinin Genel Başarısı

Bkz. Bölüm 3.5, Şekil 3.5.1 — tüm modeller baseline'ın (%72,5) belirgin şekilde üzerinde performans göstermektedir.

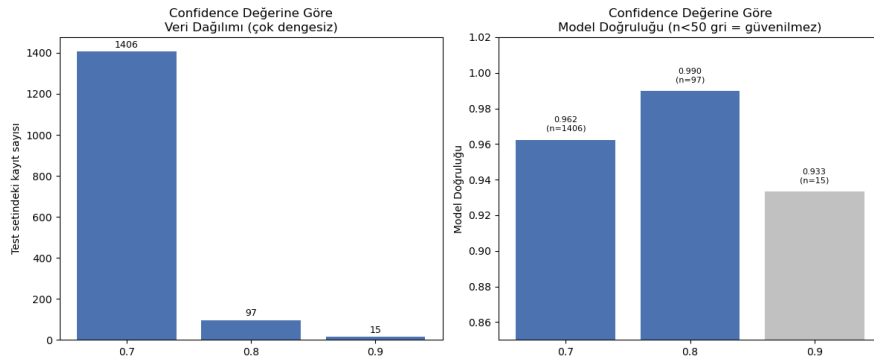
3.9.7. Kategori Bazında Model Zorlukları



Şekil 3.9.7.1. — Dengeli modellerin kategori bazında “yes” yakalama oranı.

Dengeli modeller, “Other” kategorisinde diğer kategorilere göre belirgin şekilde daha zayıf performans göstermektedir.

3.9.8. Etiket Güveni (confidence) ile Model Performansı



Şekil 3.9.8.1. — confidence değerine göre veri dağılımı ve model doğruluğu.

Bkz. Bölüm 3.8.5 — veri neredeyse hiç çeşitlilik göstermediğinden bu değişken ile model performansı arasında güvenilir bir ilişki kurulamamıştır.

4. SONUÇ

- Geliştirilen sınıflandırma sistemi, baseline'ı (%72,5) belirgin şekilde aşarak %96,4–%96,9 aralığında accuracy elde etmiştir; en iyi performans Optimized XGBoost ve Optimized Random Forest ile sağlanmıştır.
- `class_weight="balanced"` gibi dengesizlik-duyarlı hiperparametrelerin eklenmesi, özellikle doğrusal ve ağaç-bagging tabanlı modellerde “no” sınıfı recall'ünde somut iyileşmeler sağlamıştır.
- Üç bağımsız algoritma ailesi aynı çekirdek kelime kümesini (propaganda, fake, claims, allegedly, west/western) en ayırt edici özellik olarak belirlemiştir.
- Ancak modelin başarısının bir kısmı, gerçek dilsel/anlamsal anlayıştan çok yüzeysel kalıplardan (metin uzunluğu, meta-dil kelimeleri, olası kaynak-şablon dili) kaynaklanıyor olabilir; bu, sonuçların temkinli yorumlanmasını gerektirir.
- Gelecek çalışmalar için öneriler:
 - Uzunluk kontrollü (length-controlled) bir alt küme ile modelin gerçekten içerik mi yoksa biçim mi öğrendiğinin test edilmesi.
 - Kaynak/kanal bilgisini modelden çıkararak (leave-source-out çapraz doğrulama) genelleme kabiliyetinin ölçülmesi.
 - confidence sütununda daha dengeli bir dağılıma sahip ek veri toplanması.
 - Küçük örneklemli kategorilerde (Health, Science, Business vb.) daha fazla etiketli veri toplanması.

5. KAYNAKÇA

Pedregosa, F. ve ark. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Hutto, C.J. & Gilbert, E.E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. *Eighth International AAAI Conference on Weblogs and Social Media*.

Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5–32.

Rennie, J. D. M., Shih, L., Teevan, J., & Karger, D. R. (2003). Tackling the Poor Assumptions of Naive Bayes Text Classifiers. *Proceedings of the 20th International Conference on Machine Learning (ICML)*.

scikit-learn geliştirici dokümantasyonu — <https://scikit-learn.org/stable/documentation.html>

NLTK Project — <https://www.nltk.org/>

Proje veri seti: veri_seti_duzenlendi.xlsx (7.957 kayıt, Rusya–Ukrayna savaşı sosyal medya/haber dezenformasyon etiketli metin verisi).