# MAKİNE ÖĞRENMESİ İLE TWITTER' DA SAHTE HESAP TESPİTİ

## FAKE ACCOUNT DETECTION IN TWITTER WITH MACHINE LEARNING


### ÖZETÇE

*Sosyal ağlar günümüzde birçok alanda insan hayatının bir parçası olmuştur. Twitter ve Facebook gibi sosyal ağ siteleri tüm dünyada milyonlarca kullanıcıyı çekmekte ve sosyal ağlarla etkileşimleri insanların hayatlarını etkilemektedir. Arkadaş edinmek ve onlarla iletişim kurmak artık daha kolay hale gelmiştir. Ancak, tüm gelişmeler ve büyüme ile birlikte sahte profiller, çevrimiçi kimliğe bürünme gibi sorunlar da büyüdü. Bu durum, gerçek dünyada topluma büyük zarar verebilir. Sosyal medya gibi büyük veri içeren platformlarda veri boyutunun sürekli büyümesinden dolayı bu platformlardaki sahte hesapların tespiti zorlaşmaktadır. Sosyal medya, iletişim amacıyla çok tercih edilmesine rağmen spam göndericiler ve dolandırıcılar için gün geçtikçe daha cazip bir hedef haline gelmektedir. Sosyal ağlardaki bu popülerlik, sahte olaylar aracılığıyla kullanıcılara yanlış bilgi sunma ve kötü niyetli içeriğin yayılmasına neden olan farklı sorunlara yol açmaktadır. Bu durum, vatandaşlar, ticari kuruluşlar ve diğerleri de dahil olmak üzere genel olarak topluma gerçek dünyada büyük bir zarar verebilir. Bu durum sosyal ağlardaki temel problemlerden biridir. Twitter 'da bazı kötü niyetli hesaplar yanlış bilgi ve gündem oluşturma gibi amaçlar için kullanılabilmektedir. Bu nedenle kötü niyetli hesapların tespit edilmesi önemlidir. Bu çalışmada insanları yanlış yönlendirebilecek sahte hesapların tespiti için makine öğrenmesi tabanlı yöntemler kullanılmıştır. Bu amaçla oluşturulan veri kümesi ön işlemden geçirilmiş ve makine öğrenmesi algoritmaları tarafından sahte hesaplar tespit edilmiştir. Sahte hesapların tespiti için k-NN, Decision Tree, Random Forest, Naive Bayes ve Support Vector Machine algoritmaları kullanılacaktır.*



***Anahtar Kelimeler***- *Sosyal Aglar, Twitter, Sahte Hesap Tespiti, Makine Ogrenmesi, Sınıflandırma Algoritmaları, Twitter sahte hesap*



### ABSTRACT

*Social networks have become a part of human life in many areas today. Social networking sites like Twitter and Facebook attract millions of users all over the world and their interactions with social networks have affected their lives It is now easier to make friends and communicate with them. However, along with all developments and growth, problems such as fake profiles and online impersonation have grown. This can harm society in the real world. In the platforms containing big data such as social media, it is difficult to detect fake accounts on these platforms due to the constantly growing data size. Social media is becoming a more attractive target for spammers and scammers, although it is preferred for communication. This popularity in social networks has led to different problems causing false information to spread to users through spurious events and the spread of malicious content during life events. This can harm society in general in the real world,*

*including citizens, businesses and others. Some malicious accounts on Twitter are used for purposes such as misinformation and creating an agenda. This is one of the main problems in social networks. Therefore, it is important to detect malicious accounts. In this study, machine learning based methods were used to detect fake accounts that could mislead people. The dataset created for this purpose has been pre-processed and fake accounts have been detected by machine learning algorithms. k-NN, Decision Tree, Random Forest, Lojistik regression, Naive Bayes and Support Vector Machine algorithms will be used to detect fake accounts.*

***Keywords***- *Social Networks, Twitter, Fake Account Detection, Machine Learning, Classification Algorithms, Twitter fake account*

## 1. GİRİŞ

Sosyal ağ olgusu son yirmi yıl içinde son derece artmıştır. Bu artış sırasında, farklı sosyal ağ türleri, çok sayıda kullanıcı da gittikçe çoğalmıştır. Bu çalışmada ele alınan sosyal medya türü, Twitter' dır. Tweet'ler e-posta göndererek veya SMS metin mesajları göndererek yayınlanabilir. Twitter, kullanıcıların 140 karakterlik mesaj kapasitesini, çok çeşitli Web tabanlı hizmetleri kullanarak doğrudan akıllı telefonlardan yayınlamasına ve

değiştirmesine olanak tanır. Twitter, bilgileri gerçek zamanlı olarak etkin olan büyük bir kullanıcı grubuna yayar [1].

Sosyal medyadaki temel sorunlardan biri, hesaplar farklı hedefler için kullanabildikleri için spam göndereciledir. Bu hedeflerden biri, toplumu büyük ölçüde etkileyebilecek söylentiler yaymaktır. Oluşturulan sahte hesapların sayısı arttırmıştır. Sahte hesaplar, gerçek insanlara ait olmayan hesaplar anlamına gelir. Sahte hesaplar; sahte haberler, yanıltıcı web derecelendirmesi ve spam sunabilir. Gerçek hesaplar, Twitter kurallarını koruyan hesaplardır fakat sahte hesaplar Twitter kurallarını ihlal etmektedir. İnsanları aldatma veya yanıltma girişimleri olabilir, örneğin, zararlı bağlantılar gönderme, toplu takip etme gibi olumsuz takip davranışları, birden fazla hesap oluşturma, ilgili olmayan güncellemelerle bağlantı gönderme, ve cevaplama işlevlerinin kötüye kullanılmasına neden olabilmektedir.

Bu çalışmada, sosyal medyanın topluma etkisinin önemine göre, sahte haberlerin, reklamların ve sahte takipçilerin yayılmasını önlemek için Twitter çevrimiçi sosyal ağından sahte profil hesaplarını tespit etme amaçlanmaktadır. Konu ile ilgili önceden yapılan çalışmalardan ardından ilgili çalışma için kullanılan veri setinden bahsedilecektir ve farklı makine öğrenmesi algoritmalarını uygulayıp, sonuçların analizi sunularak, çalışma sonuçlandırılacaktır.

## 2. LİTERATÜR TARAMASI

Sahte hesap tespiti için farklı araştırmalar sunulmuştur. Sahte profilleri saptamak çevrimiçi araçlar tarafından da sunulmuştur, bu araçlardan biri sekiz özelliği kontrol eden “FakeFollowerCheck” [3] 'dır, ancak bu özelliklerin sınıflandırma için nasıl kullanıldığını veya kullanılan tekniğin ne olduğunu açıklayan herhangi bir ayrıntı yoktur. Ayrıca, diğer araştırmacılar sahte twitter hesabını tespit etmek için başka bir kriter seti tanımlamışlardı. Referans [4], ABD Başkanı seçimlerinde Obama, Romney ve diğer politikacıların hesabına sahte takipçileri tespit etmek için bir çalışma sunmuştur. Sahte takipçilerin tespiti için yirmi iki kriter kullanılmış; algoritması, her bir takipçi hesabı için belirlenen ölçütlere göre puanlarını hesaplamaya dayanmakta ve hesap, her kategoride kazanılan puanlara göre insan veya sahte olarak kategorize edilmektedir.

Çevrimiçi Sosyal Ağların (OSN'lerin) yaygınlaşmasıyla, bu tür hizmetlere katılan kullanıcıların gizliliği büyük bir endişe kaynağı haline gelmektedir. Birçok araştırmacı, OSN' lerde profilleri olan kişiler için gizlilik tehdidini azaltmak için çözümler önermiştri. Bu tür gizlilik tehdidi azaltma çözümlerinin bir örneği olarak, [5] 'de Sanal Özel Sosyal Ağ (VPSN) kavramını önermişlerdir. VPSN temel olarak OSN' lerdeki Sanal Özel Ağ kavramını yansıtmakta: yalnızca VPSN içindeki arkadaşlar bir kişinin gerçek bilgilerini görebilir. OSN' deki diğer kişilerin yanı sıra OSN yöneticisi de aynı bilgilere erişme araçlarına sahip değildir. Literatürde OSN' nin farkında olduğu bilginin korunmasına yönelik çalışmaların çoğu, yalnızca yetkili kişilerce yetkili bir şekilde erişilebilirdir. Örneğin, [6] 'da, bir düşmanın, mağdurun profilinde paylaştığı bilgilere, mağdurun isteklerine karşı nasıl erişebileceğini göstermektedir. Gerçek bir kişinin OSN' ye koyduğu bilginin gizliliğini koruma sorunu en azından dikkate alınmış olsa da, belirli bir şekilde kullanamayan insanların gizliliğinin korunmasına çok az dikkat edildiğine inanılmaktadır. Aslında, bir mağdurun özel bilgilerini almak isteyen bir düşman, “sosyal mühendislik” tipi bir saldırı düzenleyebilir. Örnek olarak, mağdurun profilini oluşturabilir ve kurbanın sahte profile bağlı gerçek arkadaşlarıyla etkileşim kurarken kurbanın özel bilgilerini almaya çalışabilir. Bu kötü niyetli davranışı Sahte Profil Saldırısı (FPA) olarak adlandırıyoruz. [7] 'te ilk önce iki varyantla OSN' ye kimlik tehdidi saldırıları monte etmenin uygulanabilirliğini gösterdiler: tek OSN ve siteler arası OSN'ler. İlk durumda, mağdurun zaten OSN'de, rakibin klon profilini oluşturacağı bir profili var. Siteler arası OSN' lerde, mağdurun saldırının yapıldığı aynı OSN' de bir profili yoktur, ancak mağdurun profili diğer OSN' lerde mevcuttur. [8] 'te, profil benzerliklerine dayanan bir tespit çerçevesi sunulmaktadır: özellik ve arkadaş ağı benzerliği. [8] 'te Facebook, düşmanın saldırıyı yürüttüğü OSN olarak kabul edilir. Twitter' ı hedef OSN olarak kabul eden benzer bir yaklaşım [9] 'de kullanılmıştır. Tespit için mevcut çözümlerin, bazı OSN' de mağdur profilinin mevcut olduğu varsayımından yararlanmakta ki bu her zaman böyle olmayabilir. Ayrıca, saldırının yapıldığı OSN ve benzerlik için bir referans olarak kabul edilen OSN aynı tipte değilse, tespit çözümlerinin performanslarının, örneğin, referans profili, klonlanan profille aynı OSN' dir. [8] 'te, yaklaşımlarının (benzerliklerin ölçülmesi) kurbanın zaten var olduğu orijinal bir profilin varlığına çok spesifik olduğunu belirtmişlerdir. Aslında, önceden var olan bir profil olmadığından, benzerlik ölçümleri ve saptanması için şimdiye kadar önerilen diğer teknikler FPA' ya uygulanamaz. Son zamanlarda başka bir sorun ortaya çıkmıştır, OSN' lerde birden fazla kimlikle ilgilidir [10]. Aynı kişiyi ifade eden benzer kimlikleri gruplandırmak için bir çerçeve önerilmektedir. Örneğin, [12] 'de, bir OSN' deki düğümlerin mikroskobik davranışını analiz ederek büyük ölçekli bir ağın küresel yapısını değerlendirmenin bir yolunu önermektedir. Onların modeli, maksimum olabilirlik ilkesini benimseyerek grafiğin uçtan uca gelişimini gözlemlemektedir. Grafiğe yeni bir düğüm gelişi göz önüne alındığında, yeni kenarların oluşturulması düğümün derecesinden ve yaşından etkilenir. [11] 'de OSN' lerin grafik evriminin

mikroskobik bir görünümünü elde ederler. Üç tip grubun evrimini değerlendirerek OSN büyümesini inceler: singletonlar, dev bileşenler ve orta bölge. Diğer araştırmacılar [12] OSN' lerin iki önemli özelliğini düşünmüş ve birleştirmişlerdir: hizmet önerisi ve arkadaşlık tahmini. Çalışma, kullanıcı-servis etkileşimleri ile kullanıcılar (arkadaşlık ağı) arasındaki bağlantıyı göstermektedir. Bununla birlikte, bu çalışmaların hiçbiri gizlilik amacıyla OSN' lerin dinamik özelliklerinden yararlanmamıştır.

## 3. YAPILACAK ÇALIŞMA

Sosyal ağdaki her profil (veya hesap) ad, cinsiyet, arkadaş sayısı, takipçi sayısı, beğeni sayısı, konum vb. gibi birçok bilgi içerir. Bu bilgilerin bazıları özeldir ve bazıları herkese açıktır. Özel bilgilere erişilemediğinden sosyal ağdaki sahte profilleri belirlemek için herkese açık bilgiler kullanılacaktır.

Sahte profillerin tespiti için izlenecek adımlar aşağıdaki gibidir:

1. Çalışma için gerekli olan veri seti araştırılıp, bulunmuştur. Sahte veya gerçek olarak sınıflandırılmış profillerin veri kümesine ihtiyaç vardır. Ad, durum sayısı, arkadaş sayısı, takipçi sayısı, sık kullanılanlar, bilinen diller vb. Gibi çeşitli özelliklerden oluşan 1337 sahte kullanıcı ve 1481 gerçek kullanıcının herkese açık bir veri kümesini kullanılmıştır.

2. Veri seti önce önişlemden geçilmiştir. Öznitelikler, diğer özniteliklere bağlı değilse ve sınıflandırma verimliliğini artırıyorsa, özellik olarak seçilir ve veri seti hazır hale getirilmiştir.

3. Hazır hale gelen bu veri kümesinden her iki profilin % 80'i (gerçek ve sahte) bir eğitim veri kümesi hazırlamak için kullanılır ve her iki profilin% 20'si bir test veri kümesi hazırlamak için kullanılır.

4. Eğitim veri kümesi daha sonra sınıflandırma algoritmasına uygulanır. Eğitim veri kümesinden öğrenir ve test veri kümesi için doğru sınıf etiketleri vermesi beklenir.

5. Test veri kümesindeki etiketler çıkarılır ve eğitimli sınıflandırıcı model tarafından belirlemeye bırakılır.

6. Sınıflandırma algoritmalarının sonucu gösterilmiştir.

### 3.1. Veri seti ve Öznitelikler

Sahte ve gerçek profillerin veri kümesine ihtiyaç vardı. Veri kümesine dahil edilen çeşitli özellikler, arkadaş sayısı, takipçi sayısı, durum sayısıdır. Veri kümesi eğitim ve test verilerine ayrılmıştır. Sınıflandırma algoritmaları eğitim veri seti kullanılarak eğitilir ve algoritmanın etkinliğini belirlemek için test veri seti kullanılır. Kullanılan veri kümesinden her iki profilin % 80'i (gerçek ve sahte) bir eğitim veri kümesi hazırlamak için kullanılır ve her iki profilin % 20'si bir test veri kümesi hazırlamak için kullanılır. Orijinal kullanıcılardan 11459 ve sahte kullanıcılardan 5789 tweet kullanılmıştır.

```
Index([u'id', u'name', u'screen_name', u'statuses_count', u'followers_count',
       u'friends_count', u'favourites_count', u'listed_count', u'created_at',
       u'url', u'lang', u'time_zone', u'location', u'default_profile',
       u'default_profile_image', u'geo_enabled', u'profile_image_url',
       u'profile_banner_url', u'profile_use_background_image',
       u'profile_background_image_url_https', u'profile_text_color',
       u'profile_image_url_https', u'profile_sidebar_border_color',
       u'profile_background_tile', u'profile_sidebar_fill_color',
       u'profile_background_image_url', u'profile_background_color',
       u'profile_link_color', u'utc_offset', u'protected', u'verified',
       u'description', u'updated', u'dataset'],
      dtype='object')
```

Şekil 3.1 Ön işlemden geçilmemiş veri setindeki öznitelikler

### 3.2. Özellik Çıkarımı

Özellik çıkarımı işlemi bir boyut işlemidir. Buna göre gereksiz veya fazla olan bir özelliği veri kümesinden çıkararak daha basit bir problem haline indirgenir. Doğru yapılmış bir özellik çıkarımı işlemi sayesinde daha kesin sonuçlara daha hızlı ulaşılmaktadır.

Veri kümesinde bulunan öznitelikler Tablo 1'de verilmiştir [8]. Bu öznitelikler bir hesabın insana veya bota ait olduğunu belirlemede önemli birer ipuçlarıdır. Örneğin; 15 yıllık bir Twitter hesabının bot olması düşük bir ihtimaldir. Çünkü biz biliyoruz ki bot hesapları anlık veya günlük yanlış bilgi sızdırma veya bilgi kötü niyetli kullanma gibi amaçları vardır. Bu nedenle bu hesaplar genellikle gündeme bağlı olarak açıldığından 15 yıllık bir botun olması düşük bir ihtimal barındırmaktadır.

```
Index([u'statuses_count', u'followers_count', u'friends_count',
       u'favourites_count', u'listed_count', u'sex_code', u'lang_code'],
      dtype='object')
       statuses_count  followers_count  friends_count  favourites_count  \
count     2818.000000      2818.000000    2818.000000       2818.000000
mean      1672.198368       371.105039     395.363023        234.541164
std       4884.669157      8022.631339     465.694322       1445.847248
min          0.000000         0.000000       0.000000          0.000000
25%         35.000000        17.000000     168.000000          0.000000
50%         77.000000        26.000000     306.000000          0.000000
75%       1087.750000       111.000000     519.000000         37.000000
max      79876.000000    408372.000000   12773.000000      44349.000000

       listed_count     sex_code    lang_code
count   2818.000000  2818.000000  2818.000000
mean       2.818666    -0.180270     2.851313
std       23.480430     1.679125     1.992950
min        0.000000    -2.000000     0.000000
25%        0.000000    -2.000000     1.000000
50%        0.000000     0.000000     1.000000
75%        1.000000     2.000000     5.000000
max      744.000000     2.000000     7.000000
```

Şekil 3.2 Veriseti özellik çıkarımı işleminden sonraki kullanılacak olan öznitelikler

### 3.3. Kullanılacak Araç ve Yöntemler

Sınıflandırma, bir nesneyi sınıflandırıcıyı eğitmek için kullanılan eğitim veri setine dayalı olarak belirli bir sınıfa kategorize etme tekniğidir. Sınıflandırıcı veri kümesiyle beslenir, böylece ilgili nesneleri mümkün olan en iyi doğrulukla tanımlamak için eğitilebilir. Sınıflandırıcı, sınıflandırma için kullanılan bir algoritmadır. Bu projede Sahte hesapların tespiti için k-NN, Decision Tree, Random Forest, Naive Bayes ve Support Vector Machine algoritmaları kullanılacaktır. Sınıflandırıcıların çalışması ile ilgili detaylar aşağıda verilmiştir.

#### *3.3.1. K-NN*

KNN tembel bir öğrenme algoritmasıdır. KNN algoritması benzer şeylerin yakınlarda var olduğunu varsayar. Başka bir deyişle, benzer şeyler birbirine yakındır. Tembel algoritma, model üretimi için herhangi bir eğitim veri noktasına ihtiyaç duymadığı anlamına gelir. Tüm eğitim verileri test aşamasında kullanılır. Bu durum, eğitimi daha hızlı ve test aşamasını daha yavaş ve daha maliyetli hale getirir. Maliyetli test aşaması zaman ve bellek anlamına gelir [13].

Bu algoritma beş adımdan oluşur.

1. Öncelikle K değeri belirlenir.
2. Diğer nesnelerden hedef nesneye olan öklit uzaklıkları hesaplanır.
3. Uzaklıklar sıralanır ve en minimum uzaklığa bağlı olarak en yakın komşular bulunur.
4. En yakın komşu kategorileri toplanır.
5. En uygun komşu kategorisi seçilir.

#### *3.3.2. Decision Tree*

Karar ağaçları, özellik ve hedefe göre karar düğümleri ve yaprak düğümlerinden oluşan ağaç yapısı formunda bir model oluşturan bir sınıflandırma yöntemidir. Karar ağacı algoritması, veri setini küçük ve hatta daha küçük parçalara bölerek geliştirilir. Bir karar düğümü bir veya birden fazla dallanma içerebilir [14].

#### *3.3.3. Random Forest*

Rastgele Orman denetimli bir öğrenme algoritmasıdır. Hem sınıflandırma hem de regresyon için kullanılabilir. Aynı zamanda esnek ve kullanımı kolay bir algoritmadır. Zaten adından da anlaşılacağı gibi, bir orman oluşturur ve bunu bir şekilde rastgele yapar. Ne kadar çok ağacı varsa, o kadar sağlam bir orman olduğu söylenir. Rastgele ormanlar rastgele seçilen veri örneklerinde karar ağaçları oluşturur, her ağaçtan bir tahmin alır ve oylama yoluyla en iyi çözümü seçer. Basit kelimelerle söylemek gerekirse: Rastgele orman, birden fazla karar ağacını oluşturur ve daha doğru ve istikrarlı bir tahmin elde etmek için onları birleştirir. Ayrıca özellik öneminin oldukça iyi bir göstergesidir. Rasgele ormanlar, öneri motorları, görüntü sınıflandırma ve özellik seçimi gibi çeşitli uygulamalara sahiptir. Kredi adaylarını sınıflandırmak, hileli faaliyeti belirlemek ve hastalıkları tahmin etmek için kullanılabilir [15].

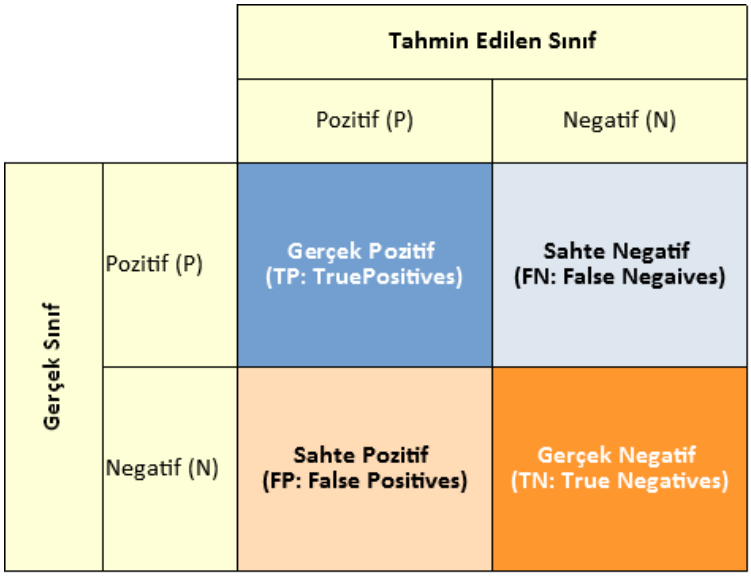
#### *3.3.4. Naivce Bayes*

Naive Bayes, Bayes Teoremine dayanan istatistiksel bir sınıflandırma tekniğidir. En basit denetimli öğrenme algoritmalarından biridir. Naive Bayes sınıflandırıcı hızlı, doğru ve güvenilir bir algoritmadır. Naive Bayes sınıflandırıcıları büyük veri kümelerinde yüksek doğruluk ve hıza sahiptir. Naive Bayes sınıflandırıcısı, bir sınıftaki belirli bir özelliğin etkisinin diğer özelliklerden bağımsız olduğunu varsayar. Bu özellikler birbirine bağlı olsa bile, bu özellikler hala bağımsız olarak kabul edilir. Bu varsayım hesaplamayı basitleştirir ve bu yüzden saf olarak kabul edilir. Bu varsayım, sınıf şartlı bağımsızlığı olarak adlandırılır [16].

#### *3.3.5. Support Vector Machine*

"Destek Vektör Makinesi (SVM), sınıflandırma veya regresyon problemleri için kullanılabilen denetimli bir makine öğrenmesi algoritmasıdır. Bununla birlikte, çoğunlukla sınıflandırma problemlerinde kullanılır. Bu algoritmada, her bir veri maddesini belirli bir koordinatın değeri olan her özelliğin değeri ile birlikte n-boyutlu boşluğa(n özellik sayısı) bir nokta olarak çizilir. Ardından, iki sınıftan oldukça iyi ayrım yapan hiper-düzlemi bularak sınıflandırma gerçekleştirilir. Destek vektörleri, sadece gözlemin koordinatlarıdır. Destek Vektör Makinesi, iki sınıftan en iyi ayıran bir sınırdır. [17]

### 3.4. Algoritma Başarım Değerlendirme Ölçütleri




| | |
|---|---|
| **Gerçek Pozitif (TP: TruePositives)** | Gerçekte pozitif olan ve pozitif olarak sınıflandırılan örnekleri belirtmektedir |
| **Sahte Negatif (FN: False Negaives)** | Gerçekte pozitif olan ve negatif olarak sınıflandırılan örnekleri belirtmektedir |
| **Sahte Pozitif (FP: False Positives)** | Gerçekte negatif olan ve pozitif olarak sınıflandırılan örnekleri belirtmektedir |
| **Gerçek Negatif (TN: True Negatives)** | Gerçekte negatif olan ve negatif olarak sınıflandırılan örnekleri belirtmektedir |

Şekil 3.3 Confusion Matris ( Karışıklık Matrisi)

Sınıflandırma algoritmalarının başarımlarının değerlendirilmesinde karışıklık matrisi(confisuon matrix) ve ROC(reciever operator characteristics curve) eğrisi gibi kriterler kullanmaktadır. Karışıklık matrisi, makine öğrenme algoritmalarında kullanılan sınıflandırma performansını değerlendirmek için hedefe ait tahminlerin ve gerçek değerlerin karşılaştırıldığı bir matristir. Sınıflandırma tahminleri dört adet değerlendirmeden birine sahip olacaktır: gerçek pozitifler, gerçek negatifler, yanlış pozitifler ve yanlış negatifler.

ROC eğrisinde ise gerçek pozitif oranı(hassasiyet), farklı kesme noktaları için yanlış pozitif oranın(100-Özgüllük fonksiyonunda çizilir. ROC eğrisindeki her nokta belirli bir karar eşiğine karşılık gelen bir duyarlılık ve özgüllük çiftini temsil eder. Mükemmel ayrımcılıkla(iki dağılımda çakışma olmaz) yapılan bir test, sol üst köşeden geçen bir ROC eğrisine sahiptir(% 100 hassasiyet,% 100 özgüllük). Bu nedenle, ROC eğrisinin sol üst köşeye yaklaştıkça, testin genel doğruluğu artar [18]. ROC eğrisi altındaki alan(AUC), bir parametrenin iki grubun ne kadar iyi ayırt edilebildiğinin bir ölçüsüdür. Bu ölçü 1'e ne kadar yakınsa o kadar mükemmeldir.

Sınıflandırma sonuçlarını değerlendirilmesinde karışıklık matrisi dışında kullanılan ölçütler aşağıdaki gibidir;

•Accuracy(isabet oranı): Çalışmadaki temel değerlendirme unsurudur. Toplam doğru tahmin sayının test edilen veriye oranıdır [19].

•Precision(kesinlik): Gerçek değeri pozitif olup pozitif değere sınıflandırılan sayısının, pozitif değere sınıflandırılanların toplamına oranıdır [19].

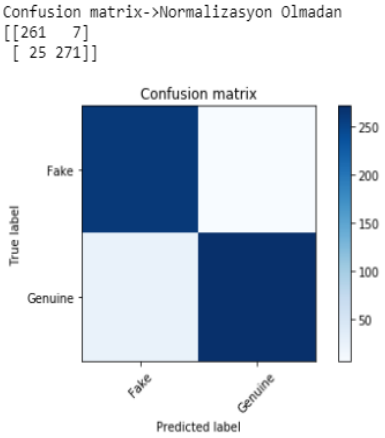
•Recall(hassasiyet): Gerçek değeri pozitif olup pozitif değere sınıflandırılan sayısının, gerçek değeri pozitif olanların tümüne oranıdır [19].

•F Score: Kesinlik ve hassasiyetin harmonik ortalamasıdır [19].

•ROC eğrisi altındaki alan(AUC), bir parametrenin iki grubun ne kadar iyi ayırt edilebildiğinin bir ölçüsüdür. Bu değer 1'e ne kadar yakınsa o kadar mükemmeldir [20].

## 4. UYGULAMA SONUÇLARI

Bu çalışmada Twitter hesaplarını gerçek veya sahte olarak sınıflandırmak amacıyla makine öğrenmesi tabanlı sınıflandırma yöntemleri kullanılmıştır. Çalışmada sınıflandırma işlemi için K-NN, Decision Tree, Random Forest, Naive Bayes, Support Vector Machine algoritmaları seçilmiştir. Sınıflandırma sonuçlarını değerlendirirken karışıklık matrisi, kesinlik, f l-ölçütü, hassasiyet, ROC ve isabet oranı gibi kriterler dikkate alınmaktadır. Karışıklık matrisi, sınıflandırma algoritmalarının öğrenme düzeyinin ne kadar verimli olduğunun gösteren matrisidir. Karışıklık matrisinden: gerçek pozitif, gerçek negatif, yanlış pozitif, yanlış negatif değerleri öğrenilebilir. İsabet oranı, çalışmanın doğruluğunu gösteren temel kriterdir. Gerçek pozitif ile gerçek negatif sayılarının toplamının toplam tahmine oranıdır [21]. Kesinlik, gerçek pozitif sayısının gerçek pozitif ve yanlış pozitif toplamına oranıdır. Olabildiğince yüksek olmalıdır [21]. Hassasiyet, gerçek pozitiflerin yanlış negatif ve gerçek pozitif toplamına oranıdır. Geliştirilen modelin gerçekleri bilme oranı da denilebilir [21]. F-ölçütü, kesinlik ve hassasiyet değerleriyle hesaplanmaktadır. Bu iki değerin harmonik ortalamasıdır [21]. ROC eğrisi altındaki alan, geliştirilen modelin her iki sınıfı ne kadar iyi ayırt edebildiğini gösteren ölçüttür.

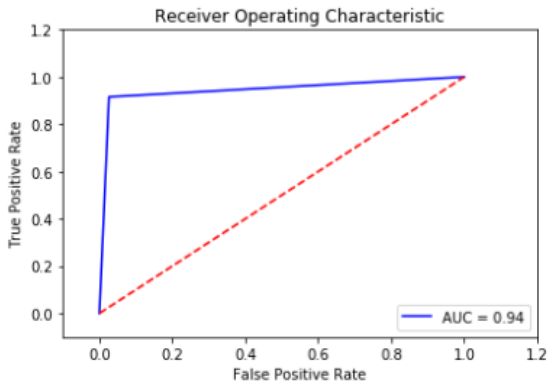



Şekil 4.1 K-NN algoritması sonucunda Confusion Matrix

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Fake | 0.91 | 0.97 | 0.94 | 268 |
| Genuine | 0.97 | 0.92 | 0.94 | 296 |
| micro avg | 0.94 | 0.94 | 0.94 | 564 |
| macro avg | 0.94 | 0.94 | 0.94 | 564 |
| weighted avg | 0.95 | 0.94 | 0.94 | 564 |

Şekil 4.2 K-NN algoritması sonucunda Ölçüm Kriterleri

```
('False Positive rate: ', array([0.       , 0.0261194, 1.       ]))
('True Positive rate: ', array([0.       , 0.91554054, 1.       ]))
```

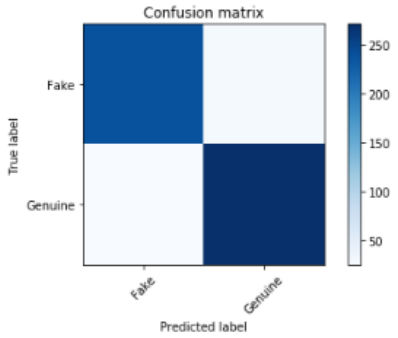



Şekil 4.3 K-NN algoritması sonucunda ROC eğrisi

Veri kümelerimizde K-NN algoritması çalıştırıldığında, %0.9432624113475178 başarı oranı elde edilmiştir. (hassasiyet) Şekil 4.1-4.3' te K-NN algoritması için confusion matrisi (karışıklık matrisi) ve diğer ölçütler olarak kesinlik, hassasiyet, f ölçütü, ROC altındaki alan ve isabet oranı) görülmektedir.

564 test verisinden 261 tanesini gerçek olarak sınıflandırmış ve 7 tanesini yanlış bir şekilde sahte hesap olarak sınıflandırmıştır. 271 tane sahte hesabı doğru şekilde etiketlemiş ve 25 tanesini yanlış bir şekilde gerçek hesap olarak etiketlenmiştir. Yukarıdaki şekil 4.3' te görüldüğü gibi ROC eğrisi altındaki alan(AUC), 0.94 olarak sonuçlanmıştır.

```
Confusion matrix->Normalizasyon Olmadan
[[240  28]
 [ 25 271]]
```

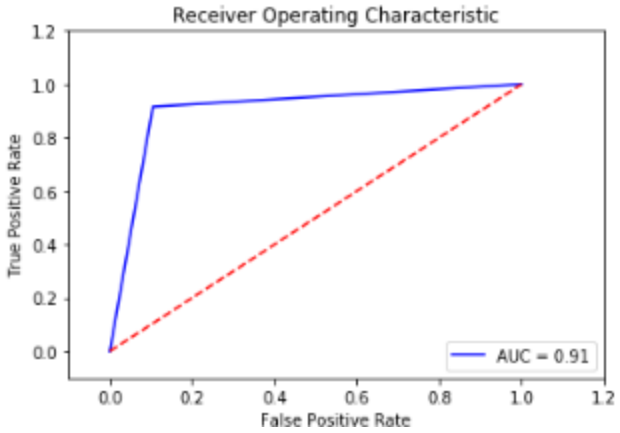



Şekil 4.4 Decision Tree algoritması sonucunda Confusion Matrix

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Fake | 0.91 | 0.90 | 0.90 | 268 |
| Genuine | 0.91 | 0.92 | 0.91 | 296 |
| micro avg | 0.91 | 0.91 | 0.91 | 564 |
| macro avg | 0.91 | 0.91 | 0.91 | 564 |
| weighted avg | 0.91 | 0.91 | 0.91 | 564 |

Şekil 4.5 Decision Tree algoritması sonucunda Ölçüm Kriterleri

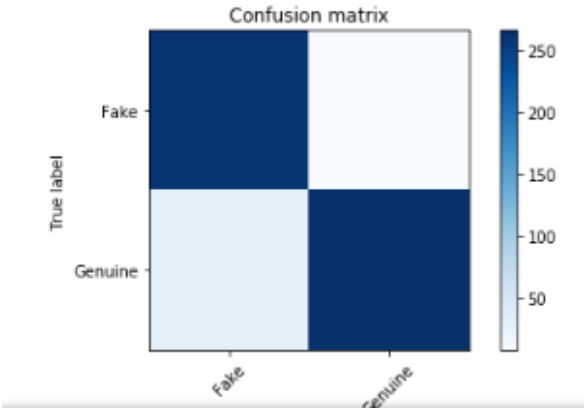
```
('False Positive rate: ', array([0.        , 0.10447761, 1.        ]))
('True Positive rate: ', array([0.        , 0.91554054, 1.        ]))
```




Şekil 4.6 Decision Tree algoritması sonucunda ROC eğrisi

Veri kümelerimizde Decision Tree algoritması çalıştırıldığında, %0. 9060283687943262 başarı oranı elde edilmiştir. (hassasiyet) Şekil 4.4 - 4.6' te Decision Tree algoritması için confusion matrisi (karışıklık matrisi) ve diğer ölçütler olara kesinlik, hassaiyet, f ölçütü, roc altındaki alan ve isabet oranı) görülmektedir. 564 test verisinden 240 tanesini gerçek olarak sınıflandırmış ve 28 tanesini yanlış bir şekilde sahte hesap olarak sınıflandırmıştır. 271 tane sahte hesabı doğru şekilde etiketlemiş ve 25 tanesini yanlış bir şekilde gerçek hesap olarak etiketlenmiştir. Yukarıdaki şekil 4.6' da görüldüğü gibi ROC eğrisi altındaki alan(AUC), 0.91 olarak sonuçlanmıştır.

```
Confusion matrix->Normalizasyon Olmadan
[[260   8]
 [ 30 266]]
```

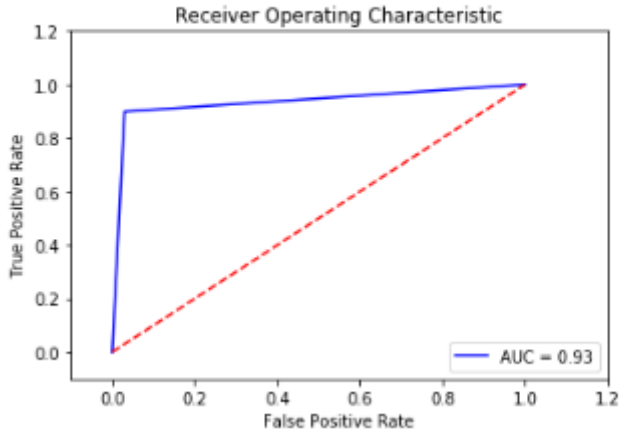



Şekil 4.7 Random Forest algoritması sonucunda Confusion Matrix

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Fake | 0.90 | 0.97 | 0.93 | 268 |
| Genuine | 0.97 | 0.90 | 0.93 | 296 |
| micro avg | 0.93 | 0.93 | 0.93 | 564 |
| macro avg | 0.93 | 0.93 | 0.93 | 564 |
| weighted avg | 0.94 | 0.93 | 0.93 | 564 |

Şekil 4.8 Random Forest algoritması sonucunda Ölçüm Kriterler

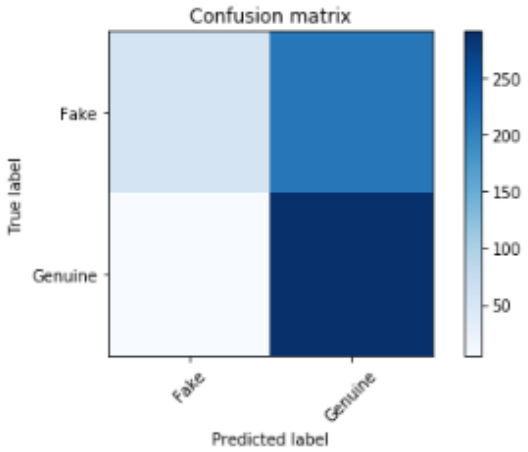
```
False Positive rate:  [0.         0.02985075 1.        ]
True Positive rate:  [0.         0.89864865 1.        ]
```




Şekil 4.9 Random Forest algoritması sonucunda ROC eğrisi

Veri kümelerimizde Random Forest algoritması çalıştırıldığında, %0.9326241134751773 başarı oranı elde edilmiştir. (hassasiyet) Şekil 4.7-4.9 ' da Random Forest algoritması için confusion matrisi (karışıklık matrisi) ve diğer ölçütler olara kesinlik, hassasiyet, f ölçütü, roc altındak alan ve isabet oranı) görülmektedir. 564 test verisinden 260 tanesini gerçek olarak sınıflandırmış ve 8 tanesini yanlış bir şekilde sahte hesap olarak sınıflandırmıştır. 266 tane sahte hesabı doğru şekilde etiketlemiş ve 30 tanesini yanlış bir şekilde gerçek hesap olarak etiketlenmiştir. Yukarıdaki şekil 4.9' da görüldüğü gibi ROC eğrisi altındaki alan(AUC), 0.93 olarak sonuçlanmıştır.

```
Confusion matrix->Normalizasyon Olmadan
[[ 56 212]
 [  5 291]]
```

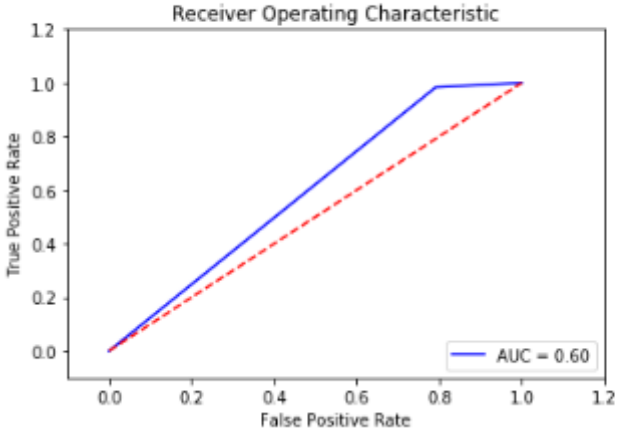



Şekil 4.10 Naive Bayes algoritması sonucunda Confusion Matrix

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Fake | 0.92 | 0.21 | 0.34 | 268 |
| Genuine | 0.58 | 0.98 | 0.73 | 296 |
| micro avg | 0.62 | 0.62 | 0.62 | 564 |
| macro avg | 0.75 | 0.60 | 0.53 | 564 |
| weighted avg | 0.74 | 0.62 | 0.54 | 564 |

Şekil 4.11 Naive Bayes algoritması sonucunda Ölçüm Kriterler

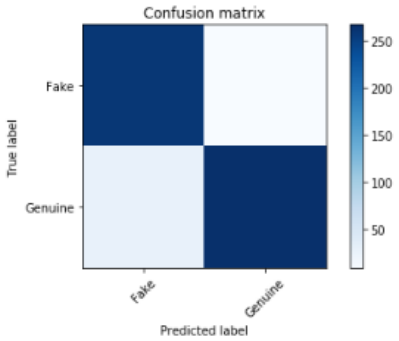
```
('False Positive rate: ', array([0.        , 0.79104478, 1.        ]))
('True Positive rate: ', array([0.        , 0.98310811, 1.        ]))
```




Şekil 4.12 Naive Bayes algoritması sonucunda ROC eğrisi

Veri kümelerimizde Naive Bayes algoritması çalıştırıldığında, %0.6152482269503546 başarı oranı elde edilmiştir. (hassasiyet) Şekil 4.10 – 4.12 'de Naive Bayes algoritması için confusion matrisi (karışıklık matrisi) ve diğer ölçütler görülmektedir. 564 test verisinden 56 tanesini gerçek olarak sınıflandırmış ve 212 tanesini yanlış bir şekilde sahte hesap olarak sınıflandırmıştır. 291 tane sahte hesabı doğru şekilde etiketlemiş ve 5 tanesini yanlış bir şekilde gerçek hesap olarak etiketlenmiştir. Yukarıdaki şekil 4.12' de görüldüğü gibi ROC eğrisi altındaki alan(AUC), 0.60 olarak sonuçlanmıştır.

```
Confusion matrix->Normalizasyon Olmadan
[[259   9]
 [ 29 267]]
```

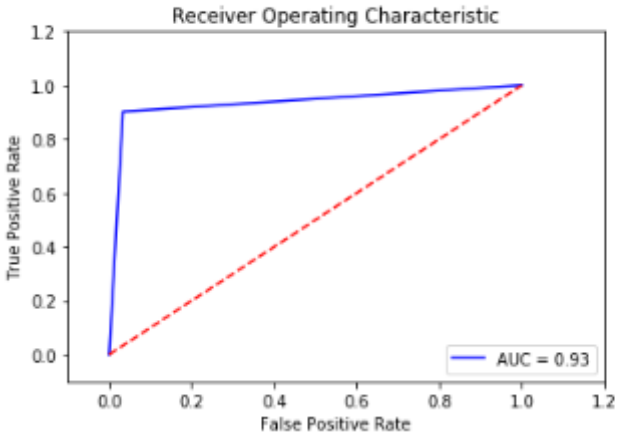



Şekil 4.13 Support Vector Machine algoritması sonucunda Confusion Matrix

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Fake | 0.90 | 0.97 | 0.93 | 268 |
| Genuine | 0.97 | 0.90 | 0.93 | 296 |
| micro avg | 0.93 | 0.93 | 0.93 | 564 |
| macro avg | 0.93 | 0.93 | 0.93 | 564 |
| weighted avg | 0.94 | 0.93 | 0.93 | 564 |

Şekil 4.14 Support Vector Machine algoritması sonucunda Ölçüm Kriterler

```
('False Positive rate: ', array([0.        , 0.03358209, 1.        ]))
('True Positive rate: ', array([0.        , 0.90202703, 1.        ]))
```




Şekil 4.15 Support Vector Machine algoritması sonucunda ROC eğrisi

Veri kümelerimizde Support Vector Machine algoritması çalıştırıldığında, %0. 9326241134751773 başarı oranı elde edilmiştir. (hassasiyet) Şekil 4.13 – 4.15' te Support Vector Machine algoritması için confusion matrisi (karışıklık matrisi) ve diğer ölçütler görülmektedir. 564 test verisinden 259 tanesini gerçek olarak sınıflandırmış ve 9 tanesini yanlış bir şekilde sahte hesap olarak sınıflandırmıştır. 267 tane sahte hesabı doğru şekilde etiketlemiş ve 29 tanesini yanlış bir şekilde gerçek hesap olarak etiketlenmiştir. Yukarıdaki şekil 4.15' te görüldüğü gibi ROC eğrisi altındaki alan(AUC), 0.93 olarak sonuçlanmıştır.

## 5. SONUÇ ve TARTIŞMA

Bu çalışmada Twitter' ın sosyal medyadaki, sahte ve gerçek hesapların sosyal ağlardaki öneminden bahsedilmiştir. Gelişen teknoloji ile beraber sosyal ağlarda giderek popülaritesini arttırmaktadır. Sahte hesapların önemi de bu doğrultuda artacaktır. Çalışmada Twitter hesaplarını sınıflandırmak için K-NN, Decision Tree, Random Forest, Naive Bayes, Support Vector Machine algoritmaları kullanılmıştır.

| Sınıflandırıcı | Accuracy |
|---|---|
| K-NN | 0.9432624113475178 |
| Decision Tree | 0.9060283687943262 |
| Random Forest | 0.9326241134751773 |
| Naive Bayes | 0.6152482269503546 |
| Support Vector Machines | 0.9326241134751773 |

Tablo 5.1 Algoritma Sonuçları

Çalışmada kullanılan beş makine öğrenmesi algoritmasının sınıflandırma sonucundaki başarım ölçütleri Şekil 5.1'de verilmiştir. Yöntemlerin AUC' lerine bakıldığında iki sınıfı en iyi ayırt eden yöntemin K-NN olduğu görülmektedir. Kesinlik ölçütü bakımından K-NN, Random Forest, Support Vector Macihne algoitmaları en iyi sonucu vermektedir. Onu Decision Tree ve Naive Bayes takip etmektedir. Kesinlik ölçütünü tek başına yorumlamak yanlış olabilir. Duyarlılık ölçütüne baktığımızda ise yine aynı sıralama görülmektedir. Kesinlik ve duyarlılık ölçütlerini beraber değerlendirmek için, her iki değerin harmonik ortalaması olan F1 ölçütüne baktığımızda yine K-NN en iyi sonuca sahip olduğu görülmektedir. Hassasiyet ölçütüne baktığımızda en yüksek oran K-NN algoritmasına aittir. Ayrıca en yüksek gerçek negatif rakamı da (FN) 212 ile Naive Bayes algoritmasındadır. Bu da veri kümesindeki sahte hesapların tespitinin daha zor olduğunu göstermektedir. Accuracy değerine bakınca K-NN algoritması en iyi sonuç vermiştir ve Random Forest ve SVM algoritmaları da aynı sonucu vermiştir.

## 6. KAYNAKÇA


[1] Global social media research summary 2019.[URL: httpd://www.smartinsights.com/social-media-marketing-social-mediastrategy/new-global-social-media-reserach (Erişim Tarihi : 27.03.2020)

[2] Davis, C. A., Varol, 0. , Ferrara, E., Flammini, A. and Menczer , F., 2016 . Botornot: A system to evaluate social bots, In Proceedings of the 25th Internati onal Conference Companion on World Wide Web, Montreal, Quebec, Canada, April 2016, 273-274

[3] SocialBakers. (Internet) http://www.socialbakers.com/products/analytics?ref=fakefollowers-top-bar, Erişim Tarihi: 26-03-2020

[4] M. Camisani-Calzolari. (2012, August ) Analysis of Twitter followers of the US Presidential Election candidates: Barack Obama and Mitt Romney. (Online). http://digitalevaluations.com/

[5] Lei Jin, Hassan Takabi, and James B.D. Joshi, "Towards active detection of identity clone attacks on online social networks", in Proceedings of the first ACM CODASPY, 2011

[6] Shah Mahmood and Yvo Desmedt, "Your Facebook deactivated friend or a cloaked spy", in Proceedings of the 4th IEEE International Workshop on SESOC '12, 2012

[7] Leyla Bilge, Thorsten Strufe, Davide Balzarotti, and Engin Kirda, "All your contacts belong to us: automated identity theft attacks on social networks", in Proceedings of the 18th WWW, 2009, pp. 551–560.

[8] Lei Jin, Hassan Takabi, and James B.D. Joshi, "Towards active detection of identity clone attacks on online social networks", in Proceedings of the first ACM CODASPY, 2011, pp. 27–38.

[9] Georgios Kontaxis, Iasonas Polakis, Sotiris Ioannidis, and Evangelos P. Markatos, "Detecting social network profile cloning.", in Proceedings of PerCom Workshop, 2011, pp. 295–300.

[10] Kahina Gani, Hakim Hacid, and Ryan Skraba, "Towards multiple identity detection in social networks", in Proceedings of the 21st international conference companion on World Wide Web, 2012, pp. 503–504.

[11] Ravi Kumar, Jasmine Novak, and Andrew Tomkins, "Structure and evolution of online social networks", in Proceedings of the 12th ACM SIGKDD, 2006, pp. 611–617.

[12] Shuang-Hong Yang, Bo Long, Alex Smola, Narayanan Sadagopan, Zhaohui Zheng, and Hongyuan Zha, "Like like alike: joint friendship and interest propagation in social networks", in Proceedings of the 20th WWW, 2011, pp. 537–546.

[13] Towardsdatascience. (Internet) https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761, Erişim Tarihi: 26-03-2020

[14] Medium. (Internet) https://medium.com/deep-math-machine-learning-ai/chapter-4-decision_trees-algorithms-b93975f7a1f1, Erişim Tarihi: 26-03-2020

[15] Medium. (Internet) https://medium.com/@Synced/how-random-forest-algorithm-works-in-machine-learning-3c0fe15b6674, Erişim Tarihi: 26-03-2020

[16] BilgisayarKavramları. (Internet) http://bilgisayarkavramlari.sadievrenseker.com/2013/02/08/naif-bayes-siniflandiricisi-naive-bayes/, Erişim Tarihi: 26-03-2020

[17] Analyticsvşdhya. (Internet https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/, Erişim Tarihi: 26-03-2020

[18] M. H. Zweig and G. Campbell, "Receiver-operating characteristic (ROC) plots: A fundamental evaluation tool in clinical medicine," *Clin. Chem.*, vol. 39, no. 4, pp. 561–577, 1993.

[19] E. Van Der Walt and J. Eloff, "Using Machine Learning to Detect Fake Identities: Bots vs Humans," *IEEE Access*, vol. 6, pp. 6540– 6549, 2018.

[20] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi, and M. Tesconi, "DNA-Inspired Online Behavioral Modeling and Its Application to Spambot Detection," *IEEE Intell. Syst.*, vol. 31, no. 5, pp. 58–64, 2016.

[21] E. Van Der Walt and J. Eloff, "Using Machine Learning to Detect Fake Identities: Bats vs Humans," IEEE Access, vol. 6, pp. 6540- 6549, 20 18.