NAİVE BAYES KULLANARAK BBC HABERLERİNİN SINIFLANDIRMASI

Mustafa DURAN

Bilgisayar Mühendisliği Bölümü, Yıldız Teknik Üniversitesi Makine Öğrenmesi Dersi
Öğrenci Numarası: 12501205

1. GİRİŞ

Bu ödevde www.bbc.co.uk/news web adresindeki haberler "Technology", "Science / Environment", "Sport" ve "Business" olmak üzere dört sınıfa ayrılacaktır. Uygulama bölümünde deneysel sonuçlar ve sistemin başarısı değerleri verilecektir. Sonuç bölümünde ise Naive Bayes algoritmasının bu problem için başarısı değerlendirilecektir.

1.1. Naive Bayes Sınıflandırıcısı

Naive Bayes algoritması koşullu olasılığa dayalı bir algoritmadır. Bu algoritma Bayes Teoremi'ni kullanır. Formüle göre daha önce elde edilen veri ile değerlerin sıklık değerleri kullanılarak elde edilen olasılıkların bileşimini kullanır.

Bayes Teorem'i daha önceden olmuş olan bir olaya dayanarak yeni bir olayın olma olasılılığını bulur. Eğer B olayı bağımlı bir olayı temsil ediyorsa ve A olayı daha önce olmuş bir olaysa, Bayes Teorem'i aşağıdaki gibi ifade edilebilir.

Bayes Teorem:

Prob(B given A) = Prob(A and B)/Prob(A)

A olayı bilindiğinde B olayının olma olasılığını hesaplamak için, algoritma A ve B olaylarının birlikte gerçekleştiği olayları sayar ve bunu A olayının tek başına gerçekleştiği olay sayısına böler.

Naive Bayes yönteminde olayların birbirinden bağımsız olduğu varsayılarak hesaplama yapılır. Aslında bu gerçekte doğru bir yaklaşım olmasa da sistemin doğruluk başarısını kayda değer biçimde etkilememektedir. Hatta Naive Bayes'in başarısı şaşırtıcı derecede iyidir. Diğer yandan bu durum algoritmayı basit ve hızlı yapmaktadır.

1.2. Naive Bayes Yönteminin Avantajları

Naive Bayes algoritması hızlı çalışır ve çok büyük veri ile de başa çıkabilir. Tahmincilerle satırları lineer olarak ölçeklendirir.

Naive Bayes, hem ikili sınıflandırma hem de çoklu sınıflandırma problemleri için kullanılabilir. (Oracle 2012)

2. UYGULAMA

Yapmış olduğum bu çalışmada www.bbc.co/news web adresinden almış olduğum haberleri aşağıda belirlediğim sınıflara ayırmak için haberler kullandım. Bu işlemi gerçekleştirmek için her bir sınıf için 10 adet haber kullanıldı. Bu 10 adet haberin alt sınıfları için dağılımı genelde 3 adet, 3 adet, 2 adet, 2 adet şeklinde oldu. Ayrıca test için benzer şekilde her bir sınıf için 10 adet ve alt sınıfları için de benzer dağılımda haber kullanıldı.

Sınıflar ve alt sınıflar adetleri ile dağılımları:

- **Technology (Toplam 10 adet)**
  - Biology (2 adet)
  - Computer / Internet (3 adet)
  - Electronics (3 adet)
  - Machine (2 adet)
- **Science / Environment (Toplam 10 adet)**
  - Space (3 adet)
  - Medicine (3 adet
  - Animals (1 adet)
  - Food (3 adet)
- **Sport (Toplam 10 adet)**
  - Motorsports (2 adet)
  - Football (3 adet)
  - Golf (2 adet)
  - Cycling (3 adet)
- **Business (Toplam 10 adet)**
  - Asia Business (2 adet)
  - Economy (4 adet)
  - Companies (4 adet)

Yukarıda belirtilen sınıfların her biri için test gerçekleştirildi. Ancak alt sınıflar için yalnızca "Sport" sınıfının alt sınıfları için test yapıldı.

2.1. Verinin Hazırlanması

Bu çalışmada kaynak olarak kullanılan bbc.co.uk/news web adresinden alınan haberler öncelikle bir txt dosyasına kaydedildi. Kaydedilen verinin matlab ile kullanılabilmesi için txt dosyaları tek tek parse işlemine tabi tutuldu. Parse işleminde boşluk karakteri, nokta, noktalı virgül, virgül ayraç olarak kullanıldı. Bu işleme ilave olarak tek tırnak ve çift tırnak işaretleri sonucu kötü yönde etkilememesi için çıkarıldı. Bu karakterlerin çıkarılmaması durumunda aynı kelime geçse bile tek tırnak ya da çift tırnak işaretinden dolayı farklı bir kelime gibi algılanacaktı.

2.2. Sözlük Oluşturulması

Veri setimizden sözlük oluşturmak için öncelikle bütün sınıflardaki kelimeler bir dosyada toplandı. Sonrasında tekrar edilen kelimeler çıkarılarak her bir sınıfta geçen kelimeden 1 adet alınmış olması sağlandı.

Sözlük oluşturma aşamasında her veri kümesinde ortak olan bazı kelimeler çıkarıldı. Bu kelimeler her sınıfta çok defa geçebilecek olan ve herhangi bir sınıfı tanımlamayan kelimeler olduğundan sistemin başarısını negatif etkileyecekti. Bu yüzden bu kelimeler sözlüğe eklenmedi. Eklenmeyen bu kelimelerin listesi aşağıdaki gibidir.

- The the on in of
- or and a is A
- to for

Çalışmada kullanılan veri ile ilgili sayısal bilgiler aşağıdaki tabloda belirtilmiştir.

Eğitim Verisi Sayısal Değerler	
Toplam Kelime Sayısı	**6913**
Sözlükteki Kelime Sayısı	**2655**
Business Sınıfı Kelime Sayısı	**1519**
Business Eşsiz Kelime Sayısı	**784**
Sport Sınıfı Kelime Sayısı	**1952**
Sport Eşsiz Kelime Sayısı	**1012**
Science Sınıfı Kelime Sayısı	**1555**
Science Eşsiz Kelime Sayısı	**777**
Technology Sınıfı Kelime Sayısı	**1887**
Technology Eşsiz Kelime Sayısı	**885**

Tablo 1

2.3. Eğitim Aşaması

Sözlüğün oluşturulmasından sonra sistemin eğitilmesi için gerekli olan diğer parametreleri elde etmek için her bir sınıfta geçen eşsiz kelimelerin sayısı bulundu. Ardından her bir sınıf ayrı ayrı parse edildi ve her bir sınıfın toplam kelime sayısı hesaplandı. Bu işlemlerin ardından sistem eğitilmiş ve test için hazır hale gelmiş oldu.

2.4. Test Aşaması

Çalışmanın test aşaması 2 kısımdan oluşmaktadır. Birincisi ana sınıflandırma, ikincisi de "Sport" sınıfına ait alt sınıflandırmaların yapılması.

2.4.1. Ana Sınıflandırma

Bu aşamada her bir sınıfta geçen kelimenin sözlükte olup olmadığı ve o sınıfta o kelimenin kaç defa geçtiği bilgisi de bulunarak aşağıdaki formüle göre bir hesaplama yapıldı. (Alpaydın 2010) Bu hesaplama her bir sınıf için ayrı ayrı yapıldı.

$$P(a_i|V_j) = \frac{n_i + 1}{n_j + |Vocabulary|}$$

Sınıflandırmayı yapmak için her bir sınıf için elde edilen sonuçlar karşılaştırıldı. Test edilen haber en yüksek değeri olan sınıfa atandı. (Bishop 2006)

Test edilen verinin sayısal değerleri aşağıdaki tabloda verilmiştir.

Test Verisi Sayısal Değerler	
Toplam Kelime Sayısı	**7372**
Business Sınıfı Örnek Haber Sayısı	**10**
Business Sınıfı Kelime Sayısı	**1648**
Sport Sınıfı Örnek Haber Sayısı	**10**
Sport Sınıfı Kelime Sayısı	**1838**
Science Sınıfı Haber Sayısı	**10**
Science Sınıfı Kelime Sayısı	**1920**
Technology Sınıfı Haber Sayısı	**10**
Technology Kelime Sayısı	**1966**

Tablo 2

Yapılan test sonucunda elde edilen sınıflar ve gerçek sınıflar aşağıdaki tablolarda verilmiştir.

Ana Sınıflar için Test Sonuçları		
Business Sınıfı için Sınıflama Sonuçları		
Test Verisi	**Test sonucu**	**Doğru / Yanlış**
Business	Business	Doğru
Business	Business	Doğru
Business	Business	Doğru
Business	Business	Doğru
Business	Business	Doğru
Business	Business	Doğru
Business	Business	Doğru
Business	Business	Doğru
Business	Business	Doğru
Business	Business	Doğru

Tablo 3

Ana Sınıflar için Test Sonuçları		
Science Sınıfı için Sınıflama Sonuçları		
Test Verisi	**Test sonucu**	**Doğru / Yanlış**
Science	Technology	Yanlış
Science	Science	Doğru
Science	Science	Doğru
Science	Science	Doğru
Science	Science	Doğru
Science	Science	Doğru
Science	Technology	Yanlış
Science	Technology	Yanlış
Science	Technology	Yanlış
Science	Science	Doğru

Tablo 4

Ana Sınıflar için Test Sonuçları		
Sport Sınıfı için Sınıflama Sonuçları		
Test Verisi	**Test sonucu**	**Doğru / Yanlış**
Sport	Sport	Doğru
Sport	Sport	Doğru
Sport	Sport	Doğru
Sport	Sport	Doğru
Sport	Sport	Doğru
Sport	Sport	Doğru
Sport	Sport	Doğru
Sport	Sport	Doğru
Sport	Sport	Doğru
Sport	Sport	Doğru

Tablo 5

Ana Sınıflar için Test Sonuçları		
Technology Sınıfı için Sınıflama Sonuçları		
Test Verisi	**Test sonucu**	**Doğru / Yanlış**
Technology	Technology	Doğru
Technology	Technology	Doğru
Technology	Science	Yanlış
Technology	Technology	Doğru
Technology	Technology	Doğru
Technology	Technology	Doğru
Technology	Technology	Doğru
Technology	Technology	Doğru
Technology	Technology	Doğru
Technology	Technology	Doğru

Tablo 6

Test sonuçlarımızı aşağıdaki formüle göre hesaplayabiliriz.

$$\text{Başarı} = \frac{Doğru\ Sınıflama\ Sayısı}{Toplam\ Veri}$$

$$\text{Business için sınıflama başarısı} = \frac{10}{10} = \%100$$

$$\text{Science için sınıflama başarısı} = \frac{6}{10} = \%60$$

$$\text{Sport için sınıflama başarısı} = \frac{10}{10} = \%100$$

$$\text{Technology için sınıflama başarısı} = \frac{9}{10} = \%90$$

$$\text{Genel Başarı} = \frac{35}{40} = \%87{,}5$$

Görüldüğü gibi "Business" ve "Sport" sınıfları için %100 başarı ile sınıflandırma gerçekleşmiştir. "Science" sınıfı için en düşük sınıflandırma başarısı elde edilmiştir. Yanlış sınıflama sonuçları incelendiğinde hepsinin de technology olarak sınıflandığı görülmektedir. Bunun sebebi bbc.co.uk/news adresinde yayınlanan science haberleri ile technology haberlerinin birbirlerine çok yakın bazen de aynı haberler olmasıdır. Örneğin yeni çıkan bir işletim sistemi hem science haberi olarak hem de technology haberi olarak görülebilmektedir. Diğer yandan her iki haber grubunda geçen kelimeler pek çok zaman aynıdır.

2.4.2.Alt Sınıflandırma

Çalışmanın devamında "Sport" sınıfına ait alt sınıfların testi gerçekleştirildi. Sistem önce eğitim için ayrılmış farklı alt sınıflardan oluşan toplamda 10 adet haber verisi ile eğitildi. Eğitim aşaması ana sınıflandırma ile birebir aynı olarak yapıldı. Eğitim aşamasında kullanılan veri ile ilgili sayısal veriler aşağıdaki tabloda verilmiştir.

Eğitim Verisi Sayısal Değerler	
Toplam Kelime Sayısı	**2448**
Sözlükteki Kelime Sayısı	**1057**
Golf Sınıfı Kelime Sayısı	**218**
Football Sınıfı Kelime Sayısı	**761**
Motorsport Sınıfı Kelime Sayısı	**834**
Cycling Sınıfı Kelime Sayısı	**678**

Tablo 7

Sistemin eğitilmesinin ardından test aşamasına geçilmiştir. Yapılan teste ait sonuçlar aşağıdaki tabloda verilmiştir.

Alt Sınıflar için Test Sonuçları		
Sport Sınıfı için Alt Sınıflama Sonuçları		
Test Verisi	**Test sonucu**	**Doğru / Yanlış**
Golf	Golf	Doğru
Football	Motorsport	Yanlış
Football	Football	Doğru
Motorsport	Motorsport	Doğru
Motorsport	Motorsport	Doğru
Golf	Cycling	Yanlış
Cycling	Cycling	Doğru
Cycling	Cycling	Doğru
Foottball	Motorsport	Yanlış
Cycling	Cycling	Doğru

Tablo 8

Ana sınıflar için kullanılan formül ile sistemimizin başarısını hesaplayabiliriz.

Golf için sınıflama başarısı = $\frac{1}{2}$ = %50

Football için sınıflama başarısı = $\frac{1}{3}$= %33

Cycling için sınıflama başarısı = $\frac{3}{3}$= %100

Motorsport için sınıflama başarısı = $\frac{2}{2}$= %100

Genel Başarı = $\frac{7}{10}$= %70

Alt sınıfların başarısı ana sınıflara göre daha düşük çıktı. Bunun sebeplerinden birisi daha az veri ile eğitilmiş olmasıdır. Diğer önemli sebebi de sınıfların birbirine yakın özellikte olmasıdır. Ancak bu testte "Football" sınıfını "Motorsport" olarak sınıflandırması bu duruma biraz ters düşmektedir. Diğer yandan her iki sınıfta geçen kelimeler incelendiğinde sözlükte geçen kelimeler açısından benzerlik gösterdiği fark edilmektedir.

Naive Bayes algoritmasının daha verimli çalışması için yeterli sayıda veriye sahip olmalıyız. Bununla birlikte sınıflarımız da iyi belirlenmiş ve birbirinden yeterince ayrılmış olmalıdır.

3. SONUÇ

Naive Bayes sınıflandırıcısı hızlı çalışması, küçük ya da çok büyük verilerle çalışabilmesi ve basit algoritmasıyla başarılı bir sınıflandırıcı. Olayları birbirinden bağımsız yapması hem anlaşılmasını hem de uygulanmasını son derece kolay kılmaktadır.

Sınıflar birbirinden yeterince ayrılıyorsa %90 larda bir başarı veren Naive Bayes sınıflandırıcısı sınıfların birbirine yakın olduğu durumlarda da çok kötü sonuçlar vermemektedir.

Özellikle doküman sınıflandırmaya çok elverişli bir araç olan Naive Bayes sınıflandırıcısı olayların birbirinden bağımsız olarak değerlendirilebileceği diğer problemlerde de rahatlıkla kullanılabilir.

KAYNAKÇA

Alpaydın, Ethem. *Introduction to Machine Learning.* Cambridge, Massachusetts: The MIT Press, 2010.

Bishop, Christopher M. *Pattern Recognition and Machine Learning.* Cambridge U.K.: Springer, 2006.

Oracle. *Oracle* . 2012. http://docs.oracle.com/cd/B28359_01/datamine.111/b28129/algo_nb.htm (Kasım 2012 tarihinde erişilmiştir).

EK -1

ÖDEVDE KULLANILAN KODLAR

```
function
[Vocabulary,business_class,science_class,technology_class,sport_clas
s] = train_Naive_Bayes
(train_data_set,business,science,technology,sport)

%Sözlük oluşturma.

for i=1:43
    x=1;
    for j=2:295

        Kelime = train_data_set{i,j};
        Kelime1 = strcmp(Kelime,'the');
        Kelime2 = strcmp(Kelime,'on');
        Kelime3 = strcmp(Kelime,'in');
        Kelime4 = strcmp(Kelime,'of');
        Kelime5 = strcmp(Kelime,'or');
        Kelime6 = strcmp(Kelime,'and');
        Kelime7 = strcmp(Kelime,'a');
        Kelime8 = strcmp(Kelime,'is');
        Kelime9 = strcmp(Kelime,'The');
        Kelime10 = strcmp(Kelime,'A');
        Kelime11 = strcmp(Kelime,'to');
        Kelime12 = strcmp(Kelime,'for');

            if ~( Kelime1==1 || Kelime2==1  || Kelime3==1  ||
Kelime4==1 || Kelime5==1  || Kelime6==1  || Kelime7==1 ||
Kelime8==1|| Kelime9==1 || Kelime10==1 || Kelime11==1 || Kelime12==1
)
                Vocabulary{i,x}= train_data_set{i,j};

            end

        x=x+1;

    end
end

emptyCells = cellfun('isempty', Vocabulary);
Vocabulary(emptyCells) = [];

%Sınıflarda geçen kelime sayısını bulma

for i=1:10

    for j=1:294

        Kelime = business{i,j};
        Kelime1 = strcmp(Kelime,'the');
        Kelime2 = strcmp(Kelime,'on');
```

```
        Kelime3 = strcmp(Kelime,'in');
        Kelime4 = strcmp(Kelime,'of');
        Kelime5 = strcmp(Kelime,'or');
        Kelime6 = strcmp(Kelime,'and');
        Kelime7 = strcmp(Kelime,'a');
        Kelime8 = strcmp(Kelime,'is');
        Kelime9 = strcmp(Kelime,'The');
        Kelime10 = strcmp(Kelime,'A');
        Kelime11 = strcmp(Kelime,'to');
        Kelime12 = strcmp(Kelime,'for');

            if ~( Kelime1==1 || Kelime2==1  || Kelime3==1  ||
Kelime4==1 || Kelime5==1  || Kelime6==1  || Kelime7==1 ||
Kelime8==1|| Kelime9==1 || Kelime10==1 || Kelime11==1 || Kelime12==1
)

                business_class{i,j}= business{i,j};

            end

    end
end

emptyCells = cellfun('isempty', business_class);
business_class(emptyCells) = [];

for i=1:10

    for j=1:294

        Kelime = science{i,j};
        Kelime1 = strcmp(Kelime,'the');
        Kelime2 = strcmp(Kelime,'on');
        Kelime3 = strcmp(Kelime,'in');
        Kelime4 = strcmp(Kelime,'of');
        Kelime5 = strcmp(Kelime,'or');
        Kelime6 = strcmp(Kelime,'and');
        Kelime7 = strcmp(Kelime,'a');
        Kelime8 = strcmp(Kelime,'is');
        Kelime9 = strcmp(Kelime,'The');
        Kelime10 = strcmp(Kelime,'A');
        Kelime11 = strcmp(Kelime,'to');
        Kelime12 = strcmp(Kelime,'for');

            if ~( Kelime1==1 || Kelime2==1  || Kelime3==1  ||
Kelime4==1 || Kelime5==1  || Kelime6==1  || Kelime7==1 ||
Kelime8==1|| Kelime9==1 || Kelime10==1 || Kelime11==1 || Kelime12==1
)

                science_class{i,j}= science{i,j};

            end
    end
end
```

```
emptyCells = cellfun('isempty', science_class);
science_class(emptyCells) = [];

for i=1:10

    for j=1:294

        Kelime = technology{i,j};
        Kelime1 = strcmp(Kelime,'the');
        Kelime2 = strcmp(Kelime,'on');
        Kelime3 = strcmp(Kelime,'in');
        Kelime4 = strcmp(Kelime,'of');
        Kelime5 = strcmp(Kelime,'or');
        Kelime6 = strcmp(Kelime,'and');
        Kelime7 = strcmp(Kelime,'a');
        Kelime8 = strcmp(Kelime,'is');
        Kelime9 = strcmp(Kelime,'The');
        Kelime10 = strcmp(Kelime,'A');
        Kelime11 = strcmp(Kelime,'to');
        Kelime12 = strcmp(Kelime,'for');

            if ~( Kelime1==1 || Kelime2==1  || Kelime3==1  ||
Kelime4==1 || Kelime5==1  || Kelime6==1  || Kelime7==1 ||
Kelime8==1|| Kelime9==1 || Kelime10==1 || Kelime11==1 || Kelime12==1
)

                technology_class{i,j}= technology{i,j};

            end
    end
end

emptyCells = cellfun('isempty', technology_class);
technology_class(emptyCells) = [];

for i=1:10

    for j=1:294

        Kelime = sport{i,j};
        Kelime1 = strcmp(Kelime,'the');
        Kelime2 = strcmp(Kelime,'on');
        Kelime3 = strcmp(Kelime,'in');
        Kelime4 = strcmp(Kelime,'of');
        Kelime5 = strcmp(Kelime,'or');
        Kelime6 = strcmp(Kelime,'and');
        Kelime7 = strcmp(Kelime,'a');
        Kelime8 = strcmp(Kelime,'is');
        Kelime9 = strcmp(Kelime,'The');
        Kelime10 = strcmp(Kelime,'A');
        Kelime11 = strcmp(Kelime,'to');
        Kelime12 = strcmp(Kelime,'for');

            if ~( Kelime1==1 || Kelime2==1  || Kelime3==1  ||
Kelime4==1 || Kelime5==1  || Kelime6==1  || Kelime7==1 ||
```

```
Kelime8==1|| Kelime9==1 || Kelime10==1 || Kelime11==1 || Kelime12==1
)
                sport_class{i,j}= sport{i,j};

            end
    end
end

emptyCells = cellfun('isempty', sport_class);
sport_class(emptyCells) = [];



function[Business_Sayisi,Science_Sayisi,Technology_Sayisi,Sport_Sayi
si] = Kelime_Sayisi_Bul(Sozluk,Business,Science,Technology,Sport)

Business_Sayisi=0;
Science_Sayisi=0;
Technology_Sayisi=0;
Sport_Sayisi=0;

for i=1:length(Business)

    for j=1:length(Sozluk)

        Esitmi = strcmp(Business(i),Sozluk(j));

        if Esitmi==1

            Business_Sayisi = Business_Sayisi + 1;

        end
    end

end

for i=1:length(Science)

    for j=1:length(Sozluk)

        Esitmi = strcmp(Science(i),Sozluk(j));

        if Esitmi==1

            Science_Sayisi = Science_Sayisi + 1;

        end
    end

end

for i=1:length(Technology)

    for j=1:length(Sozluk)
```

```
        Esitmi = strcmp(Technology(i),Sozluk(j));

        if Esitmi==1

            Technology_Sayisi = Technology_Sayisi + 1;

        end
    end

end

for i=1:length(Sport)

    for j=1:length(Sozluk)

        Esitmi = strcmp(Sport(i),Sozluk(j));

        if Esitmi==1

            Sport_Sayisi = Sport_Sayisi + 1;

        end
    end

end


function [Score_Matrix] = Test_Naive_Bayes
(Business_Sayisi,Science_Sayisi,Technology_Sayisi,Sport_Sayisi,Sozlu
k,
Test_Data,business_class,science_class,technology_class,sport_class)

Sozcuk_Sayisi = length(Sozluk);

Matris_Boyutu = size(Test_Data);

for i=1:Matris_Boyutu(1)

Total_Score_Class_Business = 0;
Total_Score_Class_Science = 0;
Total_Score_Class_Technology = 0;
Total_Score_Class_Sport = 0;


    for j=1:Matris_Boyutu(2)

        Bosmu = isempty(Test_Data{i,j});

        if Bosmu == 0

            Kelime_Varmi=ismember(Test_Data{i,j},Sozluk);

            if Kelime_Varmi == 1
```

```
                Kac_Tane_Business =
ismember(business_class,Test_Data{i,j});

                Sayisi_Business = sum (Kac_Tane_Business);

                Kelime_Sayisi_Business = Sayisi_Business;

                Kac_Tane_Science =
ismember(science_class,Test_Data{i,j});

                Sayisi_Science = sum (Kac_Tane_Science);

                Kelime_Sayisi_Science = Sayisi_Science;

                Kac_Tane_Technology =
ismember(technology_class,Test_Data{i,j});

                Sayisi_Technology = sum (Kac_Tane_Technology);

                Kelime_Sayisi_Technology = Sayisi_Technology;

                Kac_Tane_Sport =
ismember(sport_class,Test_Data{i,j});

                Sayisi_Sport = sum (Kac_Tane_Sport);

                Kelime_Sayisi_Sport = Sayisi_Sport;

            else

               Kelime_Sayisi_Sport = 0;
               Kelime_Sayisi_Technology = 0;
               Kelime_Sayisi_Science = 0;
               Kelime_Sayisi_Business = 0;

            end

        display (Kelime_Sayisi_Business);
        display (Kelime_Sayisi_Science);
        display (Kelime_Sayisi_Technology);
        display (Kelime_Sayisi_Sport);

        Score_Class_Business =
log10((1+Kelime_Sayisi_Business)/(Business_Sayisi+Sozcuk_Sayisi));

        Total_Score_Class_Business = Score_Class_Business +
Total_Score_Class_Business;

        Score_Class_Science =
log10((1+Kelime_Sayisi_Science)/(Science_Sayisi+Sozcuk_Sayisi));

        Total_Score_Class_Science = Score_Class_Science +
Total_Score_Class_Science;
```

```
        Score_Class_Technology =
log10((1+Kelime_Sayisi_Technology)/(Technology_Sayisi+Sozcuk_Sayisi)
);

        Total_Score_Class_Technology = Score_Class_Technology +
Total_Score_Class_Technology;

        Score_Class_Sport =
log10((1+Kelime_Sayisi_Sport)/(Sport_Sayisi+Sozcuk_Sayisi));

        Total_Score_Class_Sport = Score_Class_Sport +
Total_Score_Class_Sport;

        end

    end

        Score_Matrix(i,1) = (Total_Score_Class_Business)*1/4;

        Score_Matrix(i,2) = (Total_Score_Class_Science)*1/4;

        Score_Matrix(i,3) = (Total_Score_Class_Technology)*1/4;

        Score_Matrix(i,4) = (Total_Score_Class_Sport)*1/4;

end


load train_data.mat;
load business_class_train_data.mat;
load science_class_train_data.mat;
load technology_class_train_data.mat;
load sport_class_train_data.mat;

[Vocabulary,business_class,science_class,technology_class,sport_clas
s] = train_Naive_Bayes
(train_data,business_class_train_data,science_class_train_data,techn
ology_class_train_data,sport_class_train_data);

load sozluk.mat;
load business_temiz.mat;
load science_temiz.mat;
load technology_temiz.mat;
load sport_temiz.mat;

[Business_Sayisi,Science_Sayisi,Technology_Sayisi,Sport_Sayisi] =
Kelime_Sayisi_Bul(Sozluk,business_temiz,science_temiz,technology_tem
iz,sport_temiz);


function [Siniflanmis] = Siniflandir (Score_Matrix)

for i=1:40
```

```
  [Maksimum, indis] =  max(Score_Matrix(i,:));

  if indis == 1

      Siniflanmis{i,1} = 'Business';

  end

  if indis == 2

      Siniflanmis{i,1} = 'Science';

  end

  if indis == 3

      Siniflanmis{i,1} = 'Technology';

  end

  if indis == 4

      Siniflanmis{i,1} = 'Sport';

  end

end


function [Sport_Score_Matrix] = Test_Sport (Sozluk_Sport,
Test_Data_Sport,cycling_class,football_class,golf_class,motorsport_c
lass)

Sozcuk_Sayisi = length(Sozluk_Sport);

Matris_Boyutu = size(Test_Data_Sport);

for i=1:Matris_Boyutu(1)

Total_Score_Class_Cycling = 0;
Total_Score_Class_Football = 0;
Total_Score_Class_Golf = 0;
Total_Score_Class_Motorsport = 0;

    for j=1:Matris_Boyutu(2)

        Bosmu = isempty(Test_Data_Sport{i,j});

        if Bosmu == 0


Kelime_Varmi=ismember(Test_Data_Sport{i,j},Sozluk_Sport);

            if Kelime_Varmi == 1
```

```
                Kac_Tane_Cycling =
ismember(cycling_class,Test_Data_Sport{i,j});

                Sayisi_Cycling = sum (Kac_Tane_Cycling);

                Kelime_Sayisi_Cycling = Sayisi_Cycling;

                Kac_Tane_Football =
ismember(football_class,Test_Data_Sport{i,j});

                Sayisi_Football = sum (Kac_Tane_Football);

                Kelime_Sayisi_Football = Sayisi_Football;

                Kac_Tane_Golf =
ismember(golf_class,Test_Data_Sport{i,j});

                Sayisi_Golf = sum (Kac_Tane_Golf);

                Kelime_Sayisi_Golf = Sayisi_Golf;

                Kac_Tane_Motorsport =
ismember(motorsport_class,Test_Data_Sport{i,j});

                Sayisi_Motorsport = sum (Kac_Tane_Motorsport);

                Kelime_Sayisi_Motorsport = Sayisi_Motorsport;

            else

               Kelime_Sayisi_Motorsport = 0;
               Kelime_Sayisi_Golf = 0;
               Kelime_Sayisi_Football = 0;
               Kelime_Sayisi_Cycling = 0;

            end

        display (Kelime_Sayisi_Cycling);
        display (Kelime_Sayisi_Football);
        display (Kelime_Sayisi_Golf);
        display (Kelime_Sayisi_Motorsport);

        Score_Class_Cycling =
log((1+Kelime_Sayisi_Cycling)/(366+Sozcuk_Sayisi));

        Total_Score_Class_Cycling = Score_Class_Cycling +
Total_Score_Class_Cycling;

        Score_Class_Football =
log((1+Kelime_Sayisi_Football)/(418+Sozcuk_Sayisi));

        Total_Score_Class_Football = Score_Class_Football +
Total_Score_Class_Football;
```

```
        Score_Class_Golf =
log((1+Kelime_Sayisi_Golf)/(148+Sozcuk_Sayisi));

        Total_Score_Class_Golf = Score_Class_Golf +
Total_Score_Class_Golf;

        Score_Class_Motorsport =
log((1+Kelime_Sayisi_Motorsport)/(417+Sozcuk_Sayisi));

        Total_Score_Class_Motorsport = Score_Class_Motorsport +
Total_Score_Class_Motorsport;



        end

    end

        Sport_Score_Matrix(i,1) = (Total_Score_Class_Cycling)*1/4;

        Sport_Score_Matrix(i,2) = (Total_Score_Class_Football)*1/4;

        Sport_Score_Matrix(i,3) = (Total_Score_Class_Golf)*1/4;

        Sport_Score_Matrix(i,4) =
(Total_Score_Class_Motorsport)*1/4;



end


function [Siniflanmis] = Siniflandir (Sport_Score_Matrix)

for i=1:10

  [Maksimum, indis] =  max(Sport_Score_Matrix(i,:));

  if indis == 1

      Siniflanmis{i,1} = 'Cycling';

  end

  if indis == 2

      Siniflanmis{i,1} = 'Football';

  end

  if indis == 3

      Siniflanmis{i,1} = 'Golf';
```

```
    end
    if indis == 4
        Siniflanmis{i,1} = 'Motorsport';
    end
end
```