

**TÜRKİYE CUMHURİYETİ**  
**YILDIZ TEKNİK ÜNİVERSİTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



**MULTINOMIAL NAIVE BAYES SINIFLANDIRICI İLE**  
**TWITTER DUYGU ANALİZİ**

16011059 — Ahmet Onur AKMAN

**BLM4580 - DOĞAL DİL İŞLEMESİNE GİRİŞ DÖNEM PROJESİ**

Prof. Dr. Banu Diri

Ocak, 2021

# İÇİNDEKİLER

---

<b>ŞEKİL LİSTESİ</b>	<b>iii</b>
<b>TABLO LİSTESİ</b>	<b>v</b>
<b>1 Giriş</b>	<b>1</b>
1.1 Problem . . . . .	1
1.2 Veri . . . . .	2
1.3 Uygun Programlama Ortamı . . . . .	3
<b>2 Tanımlamalar</b>	<b>4</b>
2.1 Twitter . . . . .	4
2.1.1 Twitter’da Mentioning . . . . .	4
2.1.2 Twitter’da Hashtags . . . . .	5
2.2 Naive Bayes Classifier . . . . .	5
2.2.1 Multinomial Naive Bayes . . . . .	5
<b>3 Yaklaşım</b>	<b>7</b>
3.1 Gözlem . . . . .	7
3.2 Program Yapısı . . . . .	7
3.3 Verilerin İşlenmesi . . . . .	8
3.4 Veri Dengeleme . . . . .	8
3.5 Başarı Tespiti . . . . .	9
3.5.1 Kullanıcı Girdileriyle Başarı Ölçümü . . . . .	9
3.6 Kütüphaneler . . . . .	9
<b>4 Program Çıktıları</b>	<b>10</b>
4.1 WordCloud . . . . .	10
4.2 Veri Setleri Karşılaştırması . . . . .	11
4.3 Hashtag Filtreleme . . . . .	13
4.4 Sözlük Kontrolü . . . . .	13
4.5 Farklı Negatif - Pozitif Tweet Oranları . . . . .	14
4.6 Farklı Training - Test Veri Boyutu Oranları . . . . .	15

<b>5 Sonuç</b>	<b>17</b>
5.1 Gözlem . . . . .	17
5.1.1 Veri Seti . . . . .	17
5.1.2 Hashtag . . . . .	18
5.1.3 Sözlük Kontrolü . . . . .	18
5.1.4 Veri İçerik Homojenliği . . . . .	19
5.1.5 Training ve Test Verileri Büyüklüğü Oranı . . . . .	20
5.2 Çıkarım . . . . .	20
5.3 Geliştirilebilirlik . . . . .	22
<b>Referanslar</b>	<b>24</b>

## ŞEKİL LİSTESİ

---

Şekil 1.1	Soldan sağa sırayla tablodaki veri setlerinin duygu etiketlerine göre veri sayılarının grafiksel gösterimi. . . . .	3
Şekil 1.2	Spyder, the Scientific Python Development Environment. . . . .	3
Şekil 2.1	Twitter . . . . .	4
Şekil 2.2	Multinomial Naive Bayes - Formülizasyon . . . . .	6
Şekil 4.1	Sentiment140 veri seti içerisindeki Tweetlerden oluşturulan WordCloud . . . . .	10
Şekil 4.2	Sentiment140 veri seti içerisindeki negatif Tweetlerden oluşturulan WordCloud . . . . .	10
Şekil 4.3	Sentiment140 veri seti içerisindeki pozitif Tweetlerden oluşturulan WordCloud . . . . .	11
Şekil 4.4	Twitter US Airline Sentiment veri seti ile yapılan denemede alınan skorlar . . . . .	11
Şekil 4.5	Twitter Sentiment Analysis Hatred Speech veri seti ile yapılan denemede alınan skorlar . . . . .	11
Şekil 4.6	Sentiment140 veri seti ile yapılan denemede alınan skorlar . . .	12
Şekil 4.7	Hatred Speech Analysis veri setinin (soldaki) ve Sentiment140 veri setinin (sağdaki) kullanılması ile elde edilen modellerin gerçek girdi sınıflandırma performansları. . . . .	12
Şekil 4.8	Hashtaglerin filtrelenmemesi (soldaki) ve Hashtaglerin Tweetlerden çıkarılması (sağdaki) ile yapılan sınıflandırmaların skorları ve Confusion Matrixleri. . . . .	13
Şekil 4.9	Sözlük filtrelemesiz (soldaki) ve sözlük kontrolü olan (sağdaki) yöntemler ile yapılan sınıflandırmaların skorları ve Confusion Matrixleri. . . . .	14
Şekil 4.10	Sözlük filtrelemeli modelin verilen cümledeki bütün kelimeleri elemesi sebebiyle işlem yapamaması. . . . .	14
Şekil 4.11	Soldan sağa doğru 1:1, 2:1, 11:10 oranında dengeleme uygulandığı zaman alınan skorlar ve Confusion Matrix. . . . .	15

Şekil 4.12 Test-Train veri ayrıştırmasının 0.1 (soldaki) ve 0.2 (sağdaki) oranlarında yapılması ile oluşan sınıflayıcıların doğruluklarının gösterimi. . . . .	16
Şekil 5.1 . . . . .	23

## TABLO LİSTESİ

---

Tablo 1.1	Çalışma kapsamında kullanılan veri setlerinin bilgileri. . . . .	2
-----------	--	---

Bu bölümde, proje kapsamında ele alınan problem ve bu problemin yapı taşları ele alınacaktır.

### 1.1 Problem

Bu çalışma kapsamında, seçilecek bir model veya yöntem ile bir metinden duygu analizi yoluyla metnin ifade ettiği duygunun çıkarımı yapılması hedeflenmiştir. Bu proje özelinde amaç, en yüksek doğruluğu veren modelin kullanılması olmamış, ders kapsamında üzerinde durulan Naive Bayes Classifier'ın bu problem için ne şekilde efektif kullanılabileceği çıkarımı yapılması ve yetersiz kaldığı alanların gözlenerek telafi edilmeye çalışılması olmuştur. Böyle bir yaklaşım için, birçok kez, farklı parametrelerle denemeler yapılarak, her bir parametrenin çıktıya olan etkisinin incelenmesi yolu izlenmiştir.

Projenin kapsamının belirlenmesi aşamasında metnin hangi tür metin olması gerektiği limitasyonu yapılmamıştır. Bu esneklikten yola çıkılarak, insanların yazdığı metinlerin duygu belirginliğinin yüksek olduğu ve konu çeşitliliğinin fazla olduğu bir metin topluluğu arayışına girilmiştir. Bu arayış sonucunda, hiçbir konu sınırlaması olmaması ve insanların mutluluk, üzüntü, sevinç, kaygı, öfke, tiksime ve şaşkınlık gibi duygularını ifade etmek için sık sık kullandığı bir platformdan toplanmış olması sebebiyle, Twitter Tweet veri setlerinin kullanılmasına karar verilmiştir. Böylece problem, "Twitter Duygu Analizi" başlığı altına özelleştirilmiştir.

Veri bulmanın kolaylığı, Naive Bayes Classifier gibi bir yöntemin kelimelerin eklerle birlikte modifiye edilmesine olan toleransının düşük olabileceği öngörüsü ve bu dilde yapılmış çalışmaların başarısı göz önünde bulundurulması ile beraber çalışmada kullanılan verilerin dili İngilizce olarak seçilmiştir.

**Tablo 1.1** Çalışma kapsamında kullanılan veri setlerinin bilgileri.

İsim	Tweet Sayısı	Pozitif	Nötr	Negatif
Sentiment140	2.100.000	X	X	X
Twitter Sentiment Analysis Hatred Speech	32.000	X		X
Twitter US Airline Sentiment	14.641	X	X	X

## 1.2 Veri

Bu tarz çalışmaların şüphesiz ki en temel ögesi üzerinde çalışılan veridir. Önceki bölümde ifade edildiği üzere, aranan veri setinin niteliği Twitter platformundan toplanmış Tweetler içeren veri setleri olacak şekilde daraltılmıştır.

Veri arayışı esnasında, farklı niteliklerde Tweetler içeren, farklı büyüklüklerde ve içeriğindeki Tweetlerle ilgili farklı veriler içeren veri setleri ile karşılaşmıştır. Bulunan veri setleri arasından, etiketleme işlemleri tamamlanmış, yeterince büyüklüğe sahip ve etiketleme doğruluğu yüksek setlerin seçilmesi hedeflenmiştir. Projenin kapsamında farklı parametrelerin sonuca yaptığı etkiyi gözlemlemek olduğu için, veri setinin de bu parametrelerden biri olabileceği düşünülmekle birden çok sayıda veri seti ile çalışılma kararı alınmıştır.

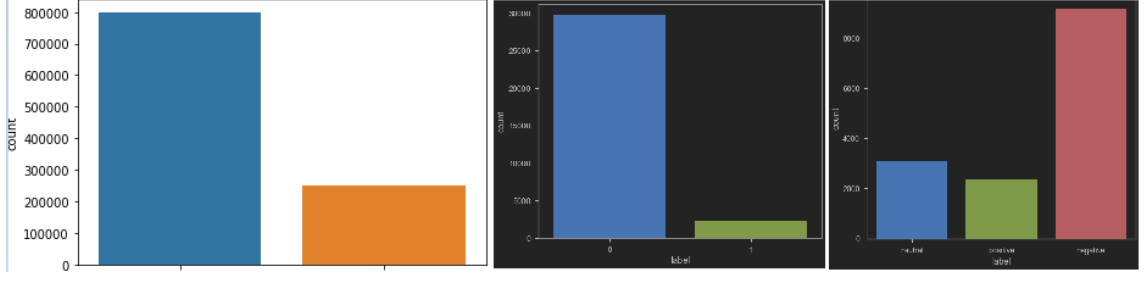
Bu veri setlerinin tutarlı olması ve benzer kod parçalarında çalışabilmesi adına, duygu etiketlemeleri pozitif ve negatif (iki tanesinde ek olarak nötr) olan veri setleri ile çalışılmasında karar kılınmıştır. Veri setleri hakkındaki temel bilgiler Tablo-1.1'deki gibidir.

Bu üç veri setinden "Sentiment140", çok sayıda veri içermesi sayesinde Naive Bayes Classifier için daha iyi bir seçenek olabileceğinden, "Twitter Sentiment Analysis Hatred Speech" Naive Bayes Classifier ile iyi sonuçlar vermiş olduğu gözlenmiş olduğundan, "Twitter US Airline Sentiment" ise bu iki veri setine alternatif olarak bir konuya odaklanmış metinler bütününde modelin başarısı ölçülmek istendiğinden ötürü seçilmiştir.

Her ne kadar konu Twitter veri setleri olduğunda internette bu konuda hazırlanmış çok sayıda örnek bulunsa da, veri setlerinin etiketli verilerinin sayısal olarak dengeli olmadığı göze çarpmaktadır. Kullanılan bu üç veri setinin program içerisinde çizdirilen, etiketlenmiş verilerinin etiketlerine göre sınıflandırıldığında her etiket için veri sayısının grafiksel gösterimi Şekil-1.1'deki gibidir. Burada negatif sentiment içeren Tweetlerin fazlalığı göze çarpmaktadır. Bu sorun program işleyişi içerisinde çözülmeye çalışılacaktır.

Veri setlerindeki bu negatif sentiment ağırlığı, etiketleme yaklaşımı, sözkonusu sosyal





**Şekil 1.1** Soldan sağa sırayla tablodaki veri setlerinin duygu etiketlerine göre veri sayılarının grafiksel gösterimi.

medya platformundaki kullanıcı profili ve veri setinin oluşturulduğu tarih aralığındaki dünya gündemi etkili faktör olabilir, bu başka bir araştırma konusudur.

### 1.3 Uygun Programlama Ortamı

Bu proje kapsamında, hem şahsi aşinalığım, hem daha önce yaptığım benzer çalışmalardaki gözlemlerimde rastladığım başarı oranları, hem kütüphane desteğinin zengin olması, hem de veri görselleştirme konusunda yeterli desteğin sağlanması sebebiyle programlama dili Python, IDE olarak ise bu konuda önde gelen seçimlerden biri olan Anaconda Spyder kullanılmıştır.



**Şekil 1.2** Spyder, the Scientific Python Development Environment.

## 2 Tanımlamalar

---

Bu bölümde çalışmada adı geçen bazı temel terimlerin tanımlaması yapılacaktır.

### 2.1 Twitter

Twitter, 2016 yılında Jack Dorsey, Noah Glass, Biz Stone ve Evan Williams tarafından Amerika Birleşik Devletlerinde kurulmuş, durum odaklı bir sosyal medya platformudur. Bu platforma yıllar boyunca gelen güncellemelerle birlikte, güncel versiyonunda kullanıcılar 280 karakter sınırı olacak şekilde metin paylaşabildiği gibi video, ses, görsel gibi farklı medya türleri de kullanabilirler. Bu metin ve medyadan oluşan bir paylaşım birimine "Tweet" adı verilmektedir.

Kullanıcıların paylaştığı Tweetler başka kullanıcılar tarafından beğenilebilir, "Retweet" edilerek alıntılanabilir ve Tweet sahibini ilgili Tweet'in altında "Mention" ederek yorum yazabilmektedir. Günümüzde bu platformda başta müzik, spor, sanat, siyaset, bilim ve teknoloji olmak üzere akla gelebilecek her konu özelinde içerik bulunabilmektedir.



Şekil 2.1 Twitter

#### 2.1.1 Twitter'da Mentioning

Twitter'da bir kullanıcı başka bir kullanıcının kullanıcı adını "@" karakteri ile beraber Tweet'ine iliştirerek o kullanıcıdan bir Tweet içerisinde bahsedebilir ve bu aksiyona

"Mentioning" denmektedir. Bu yolu kullanıcılar aynı zamanda başka birinin paylaştığı Tweet'e yorum yapma aracı olarak da kullanmaktadır.

### **2.1.2 Twitter'da Hashtags**

Twitter'da kullanıcılar paylaştıkları içeriklerin kategorisini belli etmek veya dünyanın başka bir yerinde bir kullanıcı konu hakkında arama yaptığında kendi Tweet'inin o başlık altında görünmesini sağlamak amacıyla, Tweetlerine "#" karakterinden sonra gelen ve sadece alfabetik karakter ve nümerik değerlerden oluşan bir ifade ekleyebilmektedir ve bu ifadeye "Hashtag" adı verilmektedir. Çok sayıda kullanıcının aynı Hashtag'i kısa bir zaman içerisinde çok sayıda kullanması, o Hashtag'in ülke veya dünya çapında gündem olmasını sağlayabilmektedir. Gündem olan Hashtag'ler diğer Twitter kullanıcıları tarafından "Gündem" sekmesinde görüntülenebilmektedir.

## **2.2 Naive Bayes Classifier**

Naive Bayes Sınıflandırıcı, bir verinin özelliklerinin bağımsız olarak verildiği varsayımı ile sınıflandırma konusunda bir öğrenme gerçekleştirir ve bu varsayımı ile bu işlemi oldukça kolaylaştırmaktadır[1]. Bu proje özelinde bakıldığında, Naive Bayes ile bir metnin "negatif", "pozitif" veya "nötr" olarak sınıflandırması yapılmak istendiğinde, Naive Bayes sınıflandırıcının metne bir kelime yığını olarak davrandığı ve kelimelerin sıralamasını gözardı ettiği görülmektedir. Bu cümledeki context bilgisinin gözardı edilmesiyle sonuçlanır. Bu kaba yaklaşım Naive Bayes Sınıflandırmanın en büyük handikaplarından biri olarak gösterilebilir.

Bu yaklaşımın yetersiz kalabileceği öngörüsüne rağmen, Naive Bayes temelli sınıflandırma algoritmaların komplike modeller ile yarışır düzeyde doğruluklarla sınıflandırma yetisine sahip olduğu gözlemlenmiştir[1]. Bu projede, başta da belirtildiği üzere, temel hedef bu yetersizliklerin analizinin yapılması ve eldeki verinin bu model için en uygun nasıl kullanılabileceğinin çıkarımı yapılmasıdır.

### **2.2.1 Multinomial Naive Bayes**

Naive Bayes Sınıflandırıcıların en yaygın olarak kullanılan versiyonu olan Multinomial Naive Bayes Algoritmaları; kelime sayısı, cümle uzunluğu, cümlelerin belli bir dilde olup olmadığının tespiti veya cümlelerin ifade ettiği duygunun belirlenmesi gibi; sınıflandırma sonucu verilen sınıf bilgilerinin ayrık olması durumunda takip edilebilecek en uygun yöntemdir.

Bu sınıflandırıcının temellerini oluşturan formülizasyon Şekil-2.2'da verilmiştir.

$$P(c | x) = \frac{P(x | c)P(c)}{P(x)}$$

Likelihood
Class Prior Probability

Posterior Probability
Predictor Prior Probability

The diagram shows the formula  $P(c | x) = \frac{P(x | c)P(c)}{P(x)}$  with four labels and arrows: 'Likelihood' points to  $P(x | c)$ , 'Class Prior Probability' points to  $P(c)$ , 'Posterior Probability' points to  $P(c | x)$ , and 'Predictor Prior Probability' points to  $P(x)$ .

$$P(c | X) = P(x_1 | c) \times P(x_2 | c) \times \cdots \times P(x_n | c) \times P(c)$$

**Şekil 2.2** Multinomial Naive Bayes - Formülizasyon

Bu bölümde çalışma boyunca temel olarak izlenen yöntemlerden ve bu yöntemlerin seçimindeki motivasyonlardan bahsedilecektir.

### 3.1 Gözlem

Naive Bayes modelinin başarısını etkileyen faktörlerin farklı şekillerde belirlenmesi ile beraber programın 6 farklı versiyonu üretilmiştir. Bu versiyonlar için en ayırt edici etmen veri setidir, bunun yanı sıra verinin işleniş şekli ve verinin training-test ayrımının sayısal oranının başarıya etkisinin gözlemlenmesi için farklı denemeler yapılmış ve elde edilen en kritik sonuçlar bu 6 farklı versiyonda toplanmıştır. Bu denemeler arasında en dikkat çekici olanlar bir sonraki bölümde incelenecektir.

### 3.2 Program Yapısı

6 farklı versiyonu üretilen program, temel olarak aynı iskelete sahiptir. Veri setlerinin yapısının farklı olması sebebiyle aynı iskelet üzerinde oynamalar yaparak bu versiyonlar ortaya çıkmıştır.

Temel olarak programın parçaları şu şekilde ayrılabilir:

- Kütüphaneler: Programın akışı, veri işleme, Naive Bayes modelinin implemente edilmesi, veri görselleştirme, WordCloud, vektörizasyon gibi işlevlerin yerine getirilebilmesi için ilgili kütüphanelerin dahil edilmesi.
- Verinin Okunması: Veri setinin ilgili dosya yolundan alınıp işlenmek üzere okunması ve gerektiği takdirde ilgilenilmeyecek olan sütunların drop edilmesi.
- Veri Dengelenmesi: Veri setindeki dengesizliğin grafiksel gösterimi daha önce paylaşılmıştı. Programların bu aşamasında, veri içerisindeki dengesizliği

kaldırmak adına, eğitim ve testte kullanılacak Tweetler bütününe daha dengeli oranda negatif veya pozitif Tweetler içermesi sağlanmıştır.

- **Veri Modifikasyonu:** Veri setinde bulunan Tweetlerin program için kullanılabilir hale getirilmesi. Burada takip edilen akışın değişkenliği başarıyı doğrudan etkilemektedir.
- **WordCloud Oluşturulması:** Programlama esnasında eldeki veri setinin karakteristiğini gözlemlemek adına bütün Tweetlerin, negatif ve pozitif etiketli olanların WordCloud'ının çıkarılması.
- **Vektörizasyon ve Ayırma:** Tweet içeriğindeki kelimelerin eğitimde kullanılabilmesi için vektörizasyonu ve train - test ayrımı yapılması.
- **Model Oluşturma ve Sınıflandırma:** Vektörize edilmiş kelimeler kullanılarak sınıflandırma yapılması için Multinomial Naive Bayes Sınıflandırıcı'nın tanımlanması.
- **Test:** Sınıflandırmanın başarısının sayısal ölçümü.
- **Kullanıcı Girdisi Temelli Başarı Ölçümü:** İsteğe bağlı olarak çalıştırılır, kullanıcının girdiği cümlelerin diğer Tweetler ile vektörize edilmesi ve Sentiment sınıflandırmasının yapılması.

### 3.3 Verilerin İşlenmesi

Çalışma için en kritik kısım burasıdır. Twitter'da kullanılan "Hashtag", "Mention" gibi kavramların tanımlanması önceki bölümde yapılmıştı. Eldeki Tweet verilerinin işlenmesi sırasında Hashtag ve Mentionların elenmesi, noktalamaların çıkarılması, bütün karakterlerin küçük harfe dönüştürülmesi, kelimeler için sözlük kontrolü yapıp sözlükte bulunmayan kelimelerin çıkarılması ve "stop words" olarak tabir edilen ve sınıflandırma için hiçbir değeri olmayan "this", "that", "I" gibi kelimelerin süzülmesi işlemi yapılmıştır. Bu işlemlerin hangisinin yapıp hangisinin yapılmamasının sonuçta hangi etkiler yarattığı bir sonraki bölümde incelenecektir.

### 3.4 Veri Dengeleme

Veri setlerinin negatif Tweetler yönünden zengin ancak diğer tür Tweetler yönünden oldukça fakir oldukları daha önce grafik gösterimi ile paylaşılmıştı. Program kapsamında, iki türlü veri içeren veri setleri için, negatif ve pozitif Tweet sayılarını daha mantıklı oranlama işlevinde modüller implemente edilmiştir. Bu modüller

ile sistemin prior classification deęerinin önyargı altında kalmaması hedeflenmiştir. Bu modülün kullanılmasıyla ve bu modülde deęişiklikler yapılmasıyla ortaya çıkan sonuçlar ilerleyen bölümlerde paylaşılacaktır.

### **3.5 Başarı Tespiti**

Başarı tespiti, daha önce veriden ayrılan test verisi üzerine yapılan tahminlerin doğruluęu üzerinden yapılmaktadır. Bu ölçüm Confusion Matrix ve f1-score gösterimi ile sunulmaktadır.

#### **3.5.1 Kullanıcı Girdileriyle Başarı Ölçümü**

Programın son kısmında, kullanıcı dilerse ifade girebilir ve bu ifade dięer Tweetler ile vektörize edilerek sınıflandırılır, sınıflandırma tahmini kullanıcı ile paylaşılır.

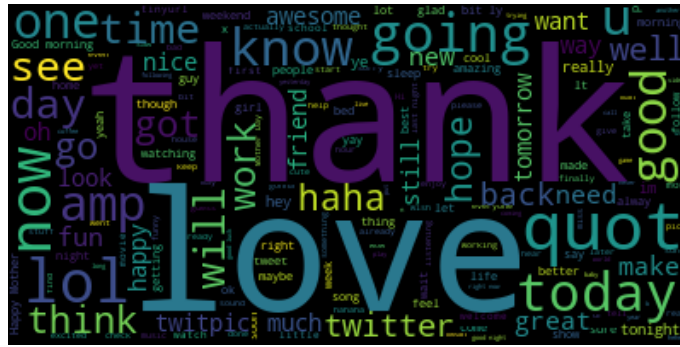
### **3.6 Kütüphaneler**

Programın işleyişı için kütüphanelerin katkısı çok büyük olmuştur. Kullanılan temel kütüphaneler ve temel amaçları şu şekildedir;

- Pandas: Veri setinin dataframe olarak işlenebilmesi.
- String: Noktalama işaretlerinin ayrıştırılabilmesi.
- nltk: Stopwords tespiti.
- re: Regular Expression kullanımı ile Hashtag, Mention ayrıştırılması.
- enchant: Sözlük kontrolü.
- Sklearn: Vektörizasyon, Data Split, Classifier Initialization, başarı ölçümü.







**Şekil 4.3** Sentiment140 veri seti içerisindeki pozitif Tweetlerden oluşturulan WordCloud

## 4.2 Veri Setleri Karşılaştırması

İlk deneme veri setleri karşılaştırması yapılacaktır. Bu deneme kapsamında bütün veri setleri aynı işleme süreçlerinden geçerek, f1-score ve kullanıcı girdilerine yaptıkları sınıflandırmalara göre değerlendirilecektir.

Tüm denemelerde Mentionlar, noktalama işaretleri ve stopwords filtrelenmiştir. Veri setleri için dengeleme operasyonu yapılmıştır. Her bir denemede test / training büyüklük oranı 0.15 olarak belirlenmiştir.

	precision	recall	f1-score	support
negative	0.77	0.93	0.84	1383
neutral	0.63	0.36	0.46	457
positive	0.74	0.53	0.62	356
accuracy			0.75	2196
macro avg	0.72	0.61	0.64	2196
weighted avg	0.74	0.75	0.73	2196

**Şekil 4.4** Twitter US Airline Sentiment veri seti ile yapılan denemede alınan skorlar

	precision	recall	f1-score	support
0	0.85	0.94	0.89	111
1	0.93	0.84	0.88	112
accuracy			0.89	223
macro avg	0.89	0.89	0.89	223
weighted avg	0.89	0.89	0.89	223

**Şekil 4.5** Twitter Sentiment Analysis Hatred Speech veri seti ile yapılan denemede alınan skorlar

Skorların karşılaştırılması sonucu, Twitter US Airline Sentiment veri seti ile yapılan denemede negatif Tweetlerin başarılı bir yüzde ile ayırt edilebildiği ancak aynı başarının pozitif Tweetler için geçerli olmadığı görülmektedir. Ortaya çıkan model farklı çıktılar için tutarsızlık durumundadır.

	precision	recall	f1-score	support
0	0.74	0.78	0.76	24696
2	0.00	0.00	0.00	229
4	0.77	0.74	0.76	24791
accuracy			0.76	49716
macro avg	0.50	0.51	0.51	49716
weighted avg	0.75	0.76	0.75	49716

**Şekil 4.6** Sentiment140 veri seti ile yapılan denemede alınan skorlar

Twitter Sentiment Analysis Hatred Speech veri seti ile yapılan denemede alınan sonuçların üç model arasında en yüksek f1-skorları olduğu görülmektedir. Sentiment140 veri seti ile yapılan denemede alınan f1-skorlarının iki türlü çıktı için de tutarlı olduğu ancak bir önceki model kadar başarı sağlamadığı fark edilmektedir.

Tutarlı sonuçlar veren iki modelin test edilmesine devam edilmiştir. Bu sefer değerlendirilen faktör kullanıcı girdisine bağlı olarak doğru sınıflandırma yapılıp yapılamadığının kontrolü olacaktır.

Yapılan denemelerde Sentiment140 ile yapılan denemelerin çok büyük bir farkla daha doğru sonuçlar verdiği görülmektedir. Daha düşük f1-score sonucuna rağmen daha doğru bir ayırımın nasıl yapıldığının analizi bir sonraki bölüme bırakılmıştır. Yapılan denemelerden bir parça Şekil-4.7’de verilmiştir.

I love my boyfriend... @joshua #happy Tweet processed into: love boyfriend happy NEGATIVE	I love my boyfriend... @joshua #happy Tweet processed into: love boyfriend happy POSITIVE
Try again? (y/n)	Try again? (y/n)
y	y
Please enter tweet to classify...	Please enter tweet to classify...
I hate this country...	I hate this country...
Tweet processed into: hate country NEUTRAL	Tweet processed into: hate country NEGATIVE
Try again? (y/n)	Try again? (y/n)
y	y
Please enter tweet to classify...	Please enter tweet to classify...
How are you doing? #convo	How are you doing? #convo
Tweet processed into: convo NEUTRAL	Tweet processed into: convo POSITIVE

**Şekil 4.7** Hatred Speech Analysis veri setinin (soldaki) ve Sentiment140 veri setinin (sağdaki) kullanılması ile elde edilen modellerin gerçek girdi sınıflandırma performansları.

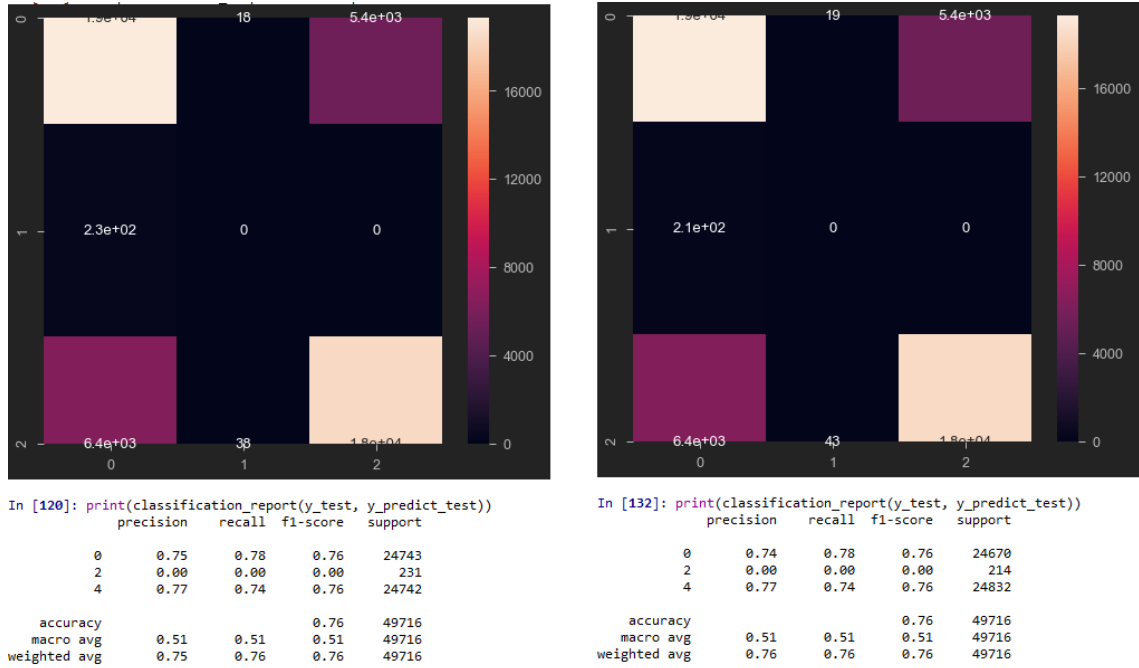
Bu ve analiz kısmında anlatılacak sebeplerden ötürü, yapılacak diğer denemeler için daha düşük f1-score üretmiş olan Sentiment140 veri setinin kullanılmasıyla

oluşturulan modeller kullanılacaktır.

### 4.3 Hashtag Filtreleme

Sentiment140 veri setiyle oluşturulan modelde önce Hashtag filtreleyerek, daha sonra sanki cümlelerin bir parçasıymış gibi veri içerisinde bırakılarak alınan sonuçlar bu kısımda paylaşılacaktır.

İki türlü yapılan sınıflandırmaların doğruluğuna dair sonuçlar Şekil-4.8’de verilmiştir.



Şekil 4.8 Hashtaglerin filtrelenmemesi (soldaki) ve Hashtaglerin Tweetlerden çıkarılması (sağdaki) ile yapılan sınıflandırmaların skorları ve Confusion Matrixleri.

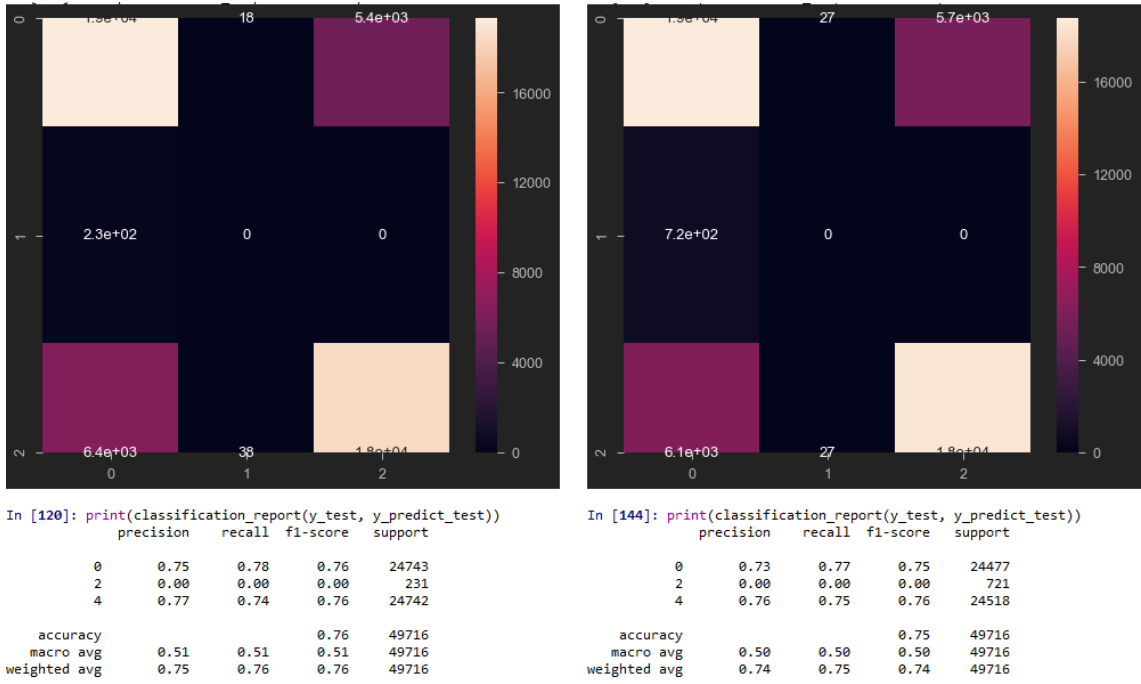
Yapılan denemelerde ortaya çıkan skorlar ve Confusion Matrixleri, iki yöntemin de hemen hemen aynı doğrulukta sınıflandırma yapabildiği sonucunu göstermektedir. Elle girilen girdiler ile yapılan denemelerde de bu sonuç gözlemlenebilmiştir.

### 4.4 Sözlük Kontrolü

Bu bölümde, Tweetlerde geçen ve İngilizce sözlüğünde bulunmayan kelimelerin varlığının sınıflandırma doğruluğuna etkisi kontrol edilecektir. İlk önce sözlük kontrolü olmadan, daha sonra kelimelere sözlük kontrolü eklenerek filtrelemeyle sınıflandırma performansı gözlemlenmiştir.

Yapılan gözlemde, filtreleme olmadan yapılan vektörizasyonda bulunan 141.641 özgün kelime sayısının 34.338’e düştüğü gözlemlenmiştir. Modelin test edilmesiyle

oluşan Confusion Matrix ve skorlar Şekil-4.9’de verilmiştir.



**Şekil 4.9** Sözlük filtrelemesiz (soldaki) ve sözlük kontrolü olan (sağdaki) yöntemler ile yapılan sınıflandırmaların skorları ve Confusion Matrixleri.

Yapılan filtreleme ile elde edilen skorlar arası çok büyük bir farklılığın oluşmadığı ancak filtreleme ile yaşanan özgün kelime sayısının devasa olduğu görülmektedir. Twitter’den rastgele alınan Tweetler ile yapılan denemelerde belirgin bir doğruluk farkı gözlemlenmezken, tamamen sadece Twitter jargonuna özel kelimeler ile oluşturulmuş Tweetlerin, sözlük filtrelemesi dolayısıyla ikinci modelde değerlendirmeye alınamadığı görülmüştür. İlgili duruma örnek Şekil-4.10’da verilmiştir.

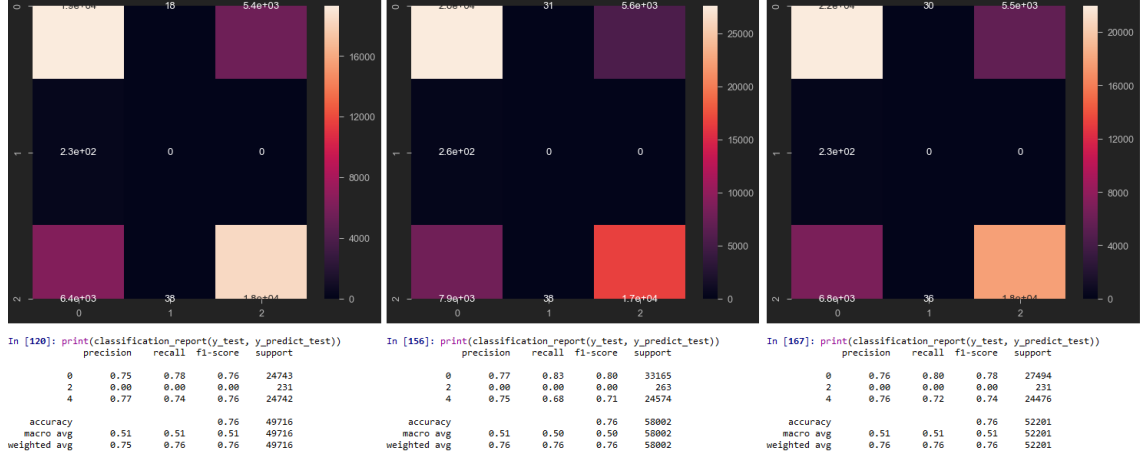
yall so stoopid smh  
Cannot be classified.

**Şekil 4.10** Sözlük filtrelemeli modelin verilen cümledeki bütün kelimeleri elemesi sebebiyle işlem yapamaması.

## 4.5 Farklı Negatif - Pozitif Tweet Oranları

Daha önceki bölümlerde de açıklandığı üzere, elimizdeki veri setlerinde dengesizlik göze çarpmaktadır. Bu sebeple programlarımız, negatif ve pozitif Tweet sayılarını eşitleyip, düzgün bir şekilde karılmasını sağlayacak bir modüle sahiptir. Bu modülde programlandığı üzere, negatif ve pozitif Tweet sayılarının tamamen aynı olması ve birbirini arkasına dizilmiş olması istenmektedir.

Bu bölümde, bu dengeleme işlemi oranının 1:1 yerine başka bir değer seçilmesiyle ne gibi sonuçlar doğacağı gözlenecektir. İlk hane negatif, ikinci hane pozitif Tweet sayısı oranını belirtmek üzere, diğer şartlar eşit tutularak; 1:1, 2:1, 11:10 oranlarında dengeleme yapılmıştır. Bu modellerin sınıflandırma başarıları Şekil-4.11’de verilmiştir.



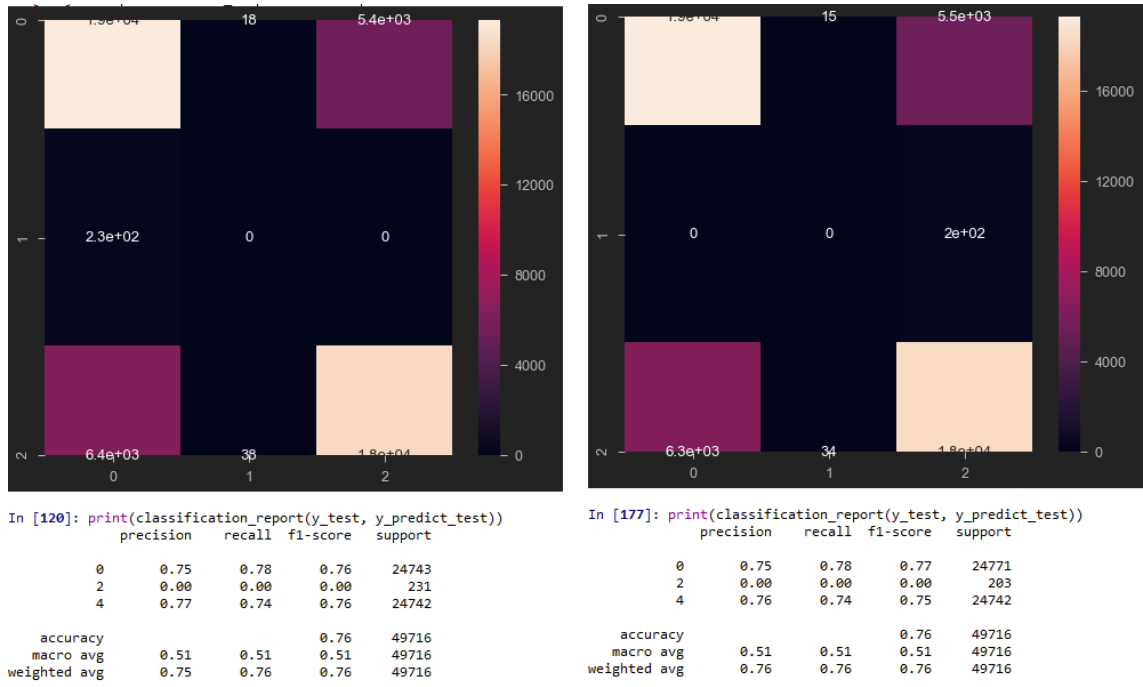
**Şekil 4.11** Soldan sağa doğru 1:1, 2:1, 11:10 oranında dengeleme uygulandığı zaman alınan skorlar ve Confusion Matrix.

Bu denemeler sonucunda elde edilen doğruluk oranlarının en net şekilde gözlemlenmesi Confusion Matrix üzerinden olmuştur. Buna göre pozitif ve negatif Tweet sayısının oranı 0.1 oranında kaydıgında bile modelin ayrıştırma yeteneğinin tutarsızlık göstermeye başladığı görülmüştür. Bu davranışın sebebi analiz kısmında iredelenecektir.

## 4.6 Farklı Training - Test Veri Boyutu Oranları

Elimizdeki verinin büyük bir porsiyonunun sınıflandırıcının eğitimi için, küçük bir kısmının ise daha sonra bu sınıflandırıcının başarısını ölçmek için ayrıldığı söylenmişti. Bu bölümde, bu ayırma işleminde kullanılan katsayının 0.1’den 0.2’ye artırılmasının neleri değiştirebileceği gözlenecektir. Bunun bize daha doğru bir test imkanı mı yoksa daha yetersiz bir eğitim mi sağladığı sorusu sorulacaktır. Oluşturulan iki sınıflayıcının doğrulukları Şekil-4.12’de gösterilmiştir.

İki sınıflayıcı için yapılan elle girdi denemelerinde bütün Tweetler aynı doğrulukta sınıflandırılmıştır.



**Şekil 4.12** Test-Train veri ayrıştırmasının 0.1 (soldaki) ve 0.2 (sağdaki) oranlarında yapılması ile oluşan sınıflayıcıların doğruluklarının gösterimi.

Bu bölümde programın farklı senaryolarda verdiği sonuçların nedenleri irdelenecektir. Varılacak sonuçlar, Multinomial Naive Bayes Classifier'ın yapısı göz önünde bulundurulmak şartıyla tamamen bu raporda bahsedilmiş deneylerden yola çıkılarak yapılacaktır.

### 5.1 Gözlem

#### 5.1.1 Veri Seti

Denemelerimizde ilk yaptığımız gözlem, veri setlerinin özelliklerinin sınıflandırıcının sınıflandırma yeteneklerine doğrudan etkisi olduğu olmuştur. Toplamda yalnızca 14.000 Tweet içeren ve Tweetleri arasında 3 farklı duygu barındıran "Twitter US Airline Sentiment" veri setinin kullanılmasıyla beraber negatif Tweet ayrıştırma için iyi sonuçlar alınsa da, pozitif ve nötr duygulu Tweetlerin ayrıştırılması konusunda aynı başarı yakalanamamıştır. Bunun yanı sıra "Twitter Sentiment Analysis Hatred Speech" veri seti ile yapılan eğitimde yüksek f1-score değerlerine erişilirken, 3 farklı duygu Tweet'i için etiketlendirme bulunduran ancak orijinalinde hiçbir nötr Tweet bulundurmayan "Sentiment140" veri seti, pozitif ve negatif Tweetlerin ayrıştırılmasında tatmin edici f1-score'u elde etmiştir. Bundan yola çıkarak bahsedilen son iki veri seti kullanıcı girdileri ile denemeler yöntemiyle tekrar sınanmış, bu sefer daha düşük f1-score'a sahip "Sentiment140" açık bir şekilde başarılı olmuştur.

Neredeyse 90% doğruluğa sahip bir modelin "love", "happy", "boyfriend" gibi kelimeler içeren bir Tweet'i doğru sınıflandıramamasının sebepleri konusunda düşünülmüş ve muhtemel nedenler şu şekilde sıralanmıştır;

- Yetersiz Test Verisi: Tweet sayısının azlığı sebebiyle test için yeterince Tweet ayrılmamıştır, bu sebeple yapılan testin niteliği yoktur. Modelin gerçek doğruluk oranı ölçülememiştir.

- **Yetersiz Özgün Kelime Sayısı:** Naive Bayes Sınıflayıcıların sınıflandırma yaparken tamamen daha önce gördükleri verilerden çıkarım yapıyor olması konusunda bilgi verilmişti. "Twitter Sentiment Analysis Hatred Speech" veri setinin zaten orta boyutta bir veri seti olduğu, dengeleme ile beraber bu sayının iyice küçültülmüş olması not edilmelidir. Burada ortaya çıkmış olabilecek sorunlardan başka bir tanesi, gelen Tweet içeriğindeki kelimeler daha önce görülmemiş olabileceği için, sınıflandırma konusunda sıkıntı yaşanmasıdır.
- **Veri Seti Kalitesi:** Veri seti kalitesinin gözden geçirilmesi denenebilir. Şayet eğer veri setindeki Tweetler için yapılan etiketlemeler özensiz yapılmışsa veya alınan Tweetlerde pozitif-negatif etiketleme yapılmış olmasına rağmen aslında nötr Tweetler barındırmasına rağmen bu değerlendirme yanlış şekilde yönlendirildiyse, veri setinin iyileştirilmesi gerekebilmektedir.
- **Veri Kaynağı:** Veri setinin oluşturulduğu zaman aralığı, insan profili ve konu da önemli bir faktördür. Örnek olarak, eğer veri tipi müzik veya oyun gibi daha çok genç kesime hitap eden konular içerisinden seçilmişse, kullanılan dilin başka herhangi bir konu başlığı altındaki paylaşımlardan farklılık gösterebilmesi beklenir. Kullanılan kelimeler, kinaye yoluyla zıt manalı kelimeler ile farklı bir duygu ifade etme alışkanlığı kullanıcı profiline göre değişebilmektedir. Bu sebeple veri setinin herhangi bir alana odaklanıp odaklanmadığı da kritik bir konudur.

### 5.1.2 Hashtag

Hashtag filtreleme denemesi yapılmadan önce, Tweet içeriğinin yanı sıra kullanılan Hashtag'in de önemli bir ifade aracı olduğu düşünüldüğünden, böyle bir etmeni ortadan kaldırmanın duygu anlatımı konusunda keskinliği ortadan kaldırabileceği tahmin edilmişti. Ancak alınan sonuçlarda, Hashtag'lerin ortadan kaldırılmasının skor ve Confusion Matrix üzerinde çok da bir etkiye sebep olmadığı, sadece precision'da ufak bir değişim olduğu görülmektedir.

Hashtag günlük hayatta gördüğümüz birçok Tweet içerisinde en kritik duygu anlatım öğelerinden biri olabilmektedir. Bu sebeple çalışmanın devamında Hashtagler Tweet içerisinde tutularak devam edilmiştir.

### 5.1.3 Sözlük Kontrolü

Sınıflandırma mantığını tamamen herkesçe geçerli, anlamlı kelimeler üzerine kurma düşüncesi ile yapılmış bu denemede, kullanılacak kelimelerin seçilmesi prosedürüne İngilizce sözlük kontrolü filtrelemesi eklenmiştir. Bu deneme, elmizdeki özgün kelime



sayısını 141.641'den 34.338'e düşürmüştür. Not edilmelidir ki, ilk verilen sayı içerisindeki her kelime yazım yanlışı veya slang değil, bazı dillerdeki rastgele tuşlara basarak gülme eylemi ifadesi de olabilmektedir.

Burada filtrelediğimiz kelimelerin bizler için ne kadar kritik olabileceği düşünülmüştür. Bu gözlemin yapılabilmesi için en uygun kaynak, yine Sentiment140 üzerinden oluşturulmuş; Şekil-4.1, Şekil-4.2 ve Şekil-4.6'ta verilen WordCloudlardır.

Sadece bu görseller incelendiğinde bile, "lol", "haha", "pic" gibi ifadelerin bu görselde yer elde edebilecek kadar sık kullanıldığını ve sınıflandırma yapma konusunda bize yardımcı olabileceği düşünülebilmektedir. Fakat yapılan denemelerde bu filtrelemenin çok da büyük bir fark yaratmadığı, sadece f1-score için 0.01'lik bir düşme yaşandığı görülmektedir.

Her ne kadar skorlar ve matrix bazında bir değişim görünmediyse de, yapılan denemelerde elden Tweet verildiği durumlarda, stopword ve Mention filtreleme işlemlerine ek olarak sözlük kontrolü de eklendiğinden ötürü bazı cümlelerin tamamen filtrelenebiliyor olduğu gözlemlenmiştir. Bu sebeple her ne kadar doğruluk konusunda dikkat çeken sayısal bir değişim yaşanmadıysa da, değerlendirebilirlik konusunda filtreli model, filtresiz modelin değerlendirebildiği birçok cümle için yetersiz kalmıştır. Örnek olarak "luv u lol" şeklinde verilmiş bir mesaj filtreli yapıda değerlendirilemeyecekken, sözlük kontrolü olmayan model bu cümlemin pozitif sentiment olduğunu anlayabilmektedir.

#### 5.1.4 Veri İçerik Homojenliği

Multinomial Naive Bayes Classifier, sınıflandırma yaparken Şekil-2.2'deki formülü temel alır. Bu formülde de görüldüğü üzere, sınıflandırma faktörlerinden bir tanesi de "Class Prior Probability" denen, yeni gelen verinin içeriğine bakılmaksızın, o ana dek sınıflandırılmış verilerin çoğunluğunun hangi sınıfa ait oluşundan çıkarım yapılarak elde edilen bir değerdir. Biz verimizdeki pozitif ve negatif sentiment Tweetlerin sayısını değiştirerek, aslında bu değere doğrudan etki etmiş olmaktadır. Bu denemede, bu değeri değiştirmemizin yarattığı etki ile Classifier'a daha çok veri göstererek daha iyi öğretmeye çalışmanın yarattığı etkiyi karşılaştırılmıştır.

Denemeler arasında sonuca belki de en gözle görülür biçimde etki eden değişiklik bu olmuştur. Görüldüğü üzere, bu oranın 11:10 olacak şekilde bile değiştirilmesi, belki de bir sınıfa çok küçük bir farkla dahil edilmiş verileri diğer sınıfa kaydırmaktadır; görünüme göre kaydırılmış bu veriler de genel olarak önceden doğru olarak sınıflandırılmış veriler olmuştur.

En başarılı sonuçlar, 1:1 oranında yakalanmıştır.

### 5.1.5 Training ve Test Verileri Büyüklüğü Oranı

Programın geliştirme süreci boyunca, Sentiment140 veri seti özelinde, yeterince büyük bir veri setimiz olduğundan ötürü, 0.1 split oranı kullanıldığında bile yeterli sayıda test verisi elde edebileceğimiz varsayılmıştır. Bu bölümde bu varsayımın ne kadar doğru olduğu sorgulanmış ve daha çok test verisi ayırdığımız durumda ortaya çıkacak sonuçlar gözlemlenmiştir. Test kabiliyetimizin artmasıyla daha doğru bir test yapılması dolayısıyla olumlu bir sonuç mu, yoksa training verisinin azalmasıyla daha yetersiz bir training süreci yaratılarak olumsuz bir sonuç mu ortaya çıkacağı merak edilmiştir.

Yapılan denemelerin sonucunda elde edilen değerlerin hemen hemen aynı olduğu, Confusion Matrix'te gözle görülebilir bir farklılık olmadığı ve f1-score'larında pozitif ve negatif sentiment Tweetler için 0.01 değişimler yaşandığı gözlemlenmiştir.

## 5.2 Çıkarım

Bu kısımda, önceki bölümde anlatılan gözlemlerden çıkarım yapılmaya çalışılacaktır. Bu çıkarım yapılırken Multinomial Naive Bayes Classifier çalışma mantığı da göz önünde bulundurulmaya çalışılacaktır.

Öncelikle, modelin sadece mevcut veri seti için değil, dışarıdan daha sonra gelen benzer türde veriyi de doğru bir şekilde sınıflandırma yeteneğine sahip olmasını sağlamak adına, modele yeterli büyüklükte veri göstermek gerektiği aşıkardır. Şayet Bölüm-4.2'de görüldüğü üzere, kısmen dar bir veri seti gösterildiği zaman, o verinin içerisinde yapılan sınıflandırma için iyi sonuçlar alındığı gözükmekteyken, dışarıdan yeni verilerle sonradan sınıflandırma yapıldığı zaman gösterilen doğruluk skorlarına paralel başarıların gözlenemediği görülmüştür. Bunun muhtemel sebepleri sıralanmıştır, bu muhtemel sebeplerin hepsini elimine etmek adına, çok sayıda özgün veri içeren, kaliteli bir şekilde etiketlenmiş, farklı kaynaklardan heterojen olarak toplanmış, yeterli büyüklükte ve iyi karıştırılmış bir veri setine ihtiyaç duyulmaktadır. Sentiment140 ile bu gerekliliklerin çoğu karşılanmıştır.

Hashtag ve sözlükte bulunmayan kelimelerin filtrelenmesi uygulaması, elimizdeki verilerin günlük hayatta karşılaşılabilecek bütün senaryoları karşılayabilen veriler olduğu takdirde uygulanabilirliği olan, filtrelemeyle beraber eldeki verinin daha nitelikli olmasını sağlayacak bir yaklaşım olarak değerlendirilebilir. Ancak elimizdeki modele bakıldığı zaman, her ne kadar kullanılan veri içerisinde 1.000.000'dan fazla Tweet olsa da, üzerinde çalışacağımız yapılar tweetler olduğu için, bazen

birkaç kelime uzunluğunda olan bu ifadelerin her bir parçası bir değer ifade eder ve böylesine katı bir filtreleme uygulamak elimizde üzerinde çalışılacak bir veri kalmaması durumunu doğurabilir. Bunun yanı sıra, Twitter gibi bir dünyada, Hashtag kullanmak azımsanmayacak derecede yaygındır, bu sebeple (bu problem özelinde) duygu çıkarımı yapmak için kıymetli bir veri niteliği taşıyabilir. Her ne kadar bu filtrelemelerin doğurduğu sonuçları sayısal olarak gözlemleme şansımız olmadıysa da, özellikle uç örneklerle beraber, elle Tweet girdisinin sınıflandırılması testi ile filtrelemenin modelden ne gibi yetileri alıp götürdüğü gözlemlenebilir.

Bu çalışma kapsamında en net bir şekilde ortaya konan husus, böyle bir problemin çözümü Naive Bayes ile aranacak ise ve üzerine çalışılan sosyal medya platformu pozitif ve negatif duyguların homojen olarak yer aldığı bir mecraysa, verilerin 1:1 oranına bağlı kalınması, her türlü farklı etikete sahip verinin eşit sayıda verilmesi kesinlikle takip edilmesi gereken yoldur. Bu oranı çok ufak miktarlarda bile kaydırma yapıldığı zaman, Naive Bayes formülizasyonundaki "prior classification" faktörüne direkt olarak etki edildiği için, sınıflandırmalardaki karar mekanizması da direkt olarak değişim yaşayabilmektedir. 1:1 oranı dışında kullanılan her oranın olumsuz sonuçlar doğurduğu görülmüştür.

Sentiment140 veri seti her ne kadar bize böyle bir problem yaşatmadıysa da, özellikle Twitter Sentiment Analysis Hatred Speech veri setinde de görebildiğimiz üzere, model geliştirme sürecinde karar verebilmek adına, iyi bir test verisinin en az iyi bir eğitim verisi kadar önemi vardır. İyi bir test verisi, modelin kalitesi, sonuçların doğruluğu, elimizdeki training verisinin yeterliliği gibi bilgileri bize açık olarak sunabilir. Test verisine önem verilmeyen durumlarda modelin başarısı doğru olarak ölçülmeyebilir ve bu da geliştirme ve analiz sürecinde yanlış kararların verilmesine neden olabilir. İyi bir analiz ve alternatif test metotları uygulanmayan bir senaryoda, bu proje kapsamında geliştirilen farklı programlar arasından Twitter Sentiment Analysis Hatred Speech kullanılan programın en iyi model olduğu çıkarımı yapılabilirdi. Bu denemeler ve yanılgılar sayesinde verinin her bir parçasının önemi net bir şekilde gözlemlenmiştir.

Son olarak kullanılan yöntem olan Naive Bayes modelinin değerlendirmesi yapılabilir. Naive Bayes, en başta da belirtildiği üzere, çok basit bir yaklaşım ile komplike yöntemlerle geliştirilen modellerle yarışır düzeyde doğruluk sunabilecek potansiyelde olan bir sınıflandırma metodudur. Elbette ki bu ifade her problem türü için doğru olmayabilir ancak bu çalışma kapsamında, denemeler sonucu elde edilen en iyi versiyonun sınıflandırma yeteneği son derece tatmin edicidir, hatta sınıflandırma konusunda uç örnekler verip modelin yanılışını gözlemleme konusunda zaman zaman zorlanılmıştır.

Ancak bu çıktıya rağmen, Naive Bayes Classification yönteminin bu problem için bile tamamen yeterli bir yaklaşım olduğunu söylemek doğru olmaz. Yaklaşım bölümünde de belirtildiği üzere, bu problem için de zayıflıklar içeren bir yöntemdir ancak bu çalışmada verinin uygun bir şekilde işlenmesi ve Naive Bayes'i en iyi şekilde kullancak programın geliştirilmesiyle sınırlar zorlanmaya çalışılmış ve böylesine basit yaklaşıma sahip bir sınıflandırma yöntemiyle, komplike hale getirilebilecek bir probleme tatmin edici bir çözüm sağlanmıştır.

### 5.3 Geliştirilebilirlik

Doğruluk oranının yüzde doksan bandını geçmiyor oluşu, elle girilen Tweetlerin sınıflandırılması sırasında zaman zaman yanlış sonuçlara rastlanıyor oluşu ve programın Naive Bayes'e uyum sağlaması için zaman zaman hantallaşıyor oluşu, şüphesiz ki çok daha gelişmiş bir model ortaya koyabileceğimizin göstergesidir.

Bu proje kapsamında bir Tweet'i sadece pozitif ve negatif olarak etiketleyen bir yapı ortaya konmuştur. Şüphesiz ki bu yaklaşım, bu problem üzerine çalışma gerektiren profesyonel sektörler için yetersiz kalacaktır ve bu kuruluşlar eldeki veriden daha anlamlı sonuçlar çıkarılmasını isteyecektir. Bu sebeple geliştirilebilecek ilk şey olarak, problemin kapsamı gösterilebilir.

Problemin kapsamının geliştirilmesi, doğal olarak veri ihtiyacının da genişletilmesi ihtiyacını doğuracaktır. Bu çalışmada 2 duygu ifadesi için 1.000.000'dan fazla Tweet kullanılmıştır ve sonuçlar ortaya konmuştur. Bu duygu sayısı ve Tweet sayısı oranına sabit kalınarak, çıkarılmak istenen her karakteristik türü kadar arttırılmış etiketlenmiş veri kullanılabilir ve eldeki model, sonradan girilen başka bir veri parçasından çok daha fazla sonuç çıkarabilir hale getirilebilir.

Verinin olduğu gibi sınıflandırma mekanizmasına aktarılması, takip edilebilecek en iyi yöntem olmayabilir. Bu problem üzerine düşünüldüğü zaman, Tweet yapısında olan ve metin olmayan hiçbir ögenin duygu analizi için kullanılmadığı görülmektedir. Oysa ki sisteme adapte edilebilecek olan görüntü analizi modülü, Tweet içeriğindeki görseli de bir faktör olarak kullanabilme yetisi kazandıracaktır. Bunun dışında Hashtag'lere sıradan metin parçası gibi davranmak yerine, Hashtagler için de ayrıca bir sınıflandırma gerçekleştirmek ve belirli bir Hashtag'i içeren Tweet için Hashtag'in kattığı anlamı da göz önünde bulundurmak da büyük boyutta doğruluk katabilme potansiyeline sahiptir. Bunların dışında göz önünde bulundurulacak diğer faktörlerden bazıları; Tweet sahibinin genel davranışı, kinaye kalıpları, Tweet'te Mention edilmiş kişinin aldığı Mention Tweetlerinde geçen duygular, Tweet'in atılmış olduğu tarih, Tweet'in konusu, Tweet'in atıldığı zaman dilimi içerisinde atılmış diğer

Tweetlerin duygu analizi olarak sayılabilir. Bu faktörleri de hesaba katarak analiz yapacak bir model, doğruluk olarak çok üst düzeylere ulaşma potansiyeline sahip olacaktır.

Her ne kadar bu çalışmanın odağı "Her ne kullanılırsa kullanılsın, çok yüksek doğruluğa erişilmelidir." olmasa da, gerçek hayattaki uygulamalara bakıldığı zaman, bu problemin çözümünde kullanılabilecek çok daha yetenekli metotların olduğu aşikardır. Öğrenme konusunu daha sistematik bir tabana oturtmuş, Naive Bayes'in zayıflıklarına sahip olmayan, context bilgisini değerlendirebilen ve daha esnek davranabilen yöntemlerin tercih edilmesi, hem programa farklı yetenekler kazanabilme potansiyeli verecek, hem de daha yüksek doğruluk oranlarıyla çalışmamıza olanak sağlayacaktır.

```
Please enter tweet to classify...
```

```
Sometimes, Naive Bayes is all you need and it is awesome!  
>>POSITIVE
```

```
Try again? (y/n)
```

Şekil 5.1

- [1] I. Rish *et al.*, “An empirical study of the naive bayes classifier,” in *IJCAI 2001 workshop on empirical methods in artificial intelligence*, vol. 3, 2001, pp. 41–46.